Statistical Modeling and Inference for Populations of Networks and Longitudinal Data

Clément Mantoux

INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2022IPPAX072

Thèse de doctorat

ÉCOLE POLYTECHNIQUE

IP PARIS

Modélisation statistique et inférence pour les populations de réseaux de connectivité cérébrale et les données longitudinales

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à l'École polytechnique

École doctorale n°574 École doctorale de mathématiques Hadamard (EDMH)
Spécialité de doctorat : Mathématiques appliquées

Thèse présentée et soutenue à Palaiseau, le 28 septembre 2022, par

CLÉMENT MANTOUX

Composition du Jury :

Marc Lelarge Directeur de recherche, Inria (DYOGENE)	Président
Xavier Pennec Directeur de recherche, Inria (Epione)	Rapporteur
Peter Hoff Professeur, Duke University	Rapporteur (absent)
Maud Delattre Chargée de recherche, Inrae (MaIAGE)	Examinateur
Eric Moulines Professeur, École polytechnique	Examinateur
Stéphanie Allassonnière Professeure, Université Paris-Cité	Directrice de thèse
Stanley Durrleman Directeur de recherche, Inria (ARAMIS)	Co-directeur de thèse


ICM
Institut du Cerveau
et de la Moelle épinière

Résumé

Le développement et la massification des bases de données d'imagerie médicale et de suivi clinique ouvrent de nouvelles perspectives pour la compréhension de phénomènes complexes comme le vieillissement ou les maladies neurodégénératives. En particulier, la connectivité cérébrale, c'est-à-dire l'étude des connexions et des interactions entre les régions du cerveau, peut maintenant être étudiée à l'échelle d'une population et non plus d'un individu isolé. Ce cadre offre la possibilité d'une meilleure prise en compte des spécificités individuelles dans le développement d'outils de suivi.

Dans cette thèse, nous proposons dans un premier temps de nouvelles approches pour modéliser et comprendre la variabilité de la connectivité cérébrale au sein d'un groupe de sujets. Plus généralement, nous nous intéressons aux collections de réseaux où chaque réseau décrit des interactions entre les mêmes entités. Nous nous appuyons sur la propriété empirique de rang faible des matrices d'adjacence de ces réseaux pour rendre compte de leur distribution. Nous proposons deux approches, l'une variationnelle et l'autre statistique, pour rendre compte de l'hétérogénéité de ces matrices. En particulier, dans le second cas, nous montrons qu'un nombre restreint de paramètres suffit à donner une description fidèle et interprétable de la variabilité de la connectivité cérébrale. Nous montrons également la consistence et l'identifiabilité de notre approche sur le plan théorique.

Dans un second temps, nous étudions un modèle longitudinal pour le suivi de la progression de la maladie de Parkinson. Dans ce modèle, la trajectoire de chaque patient est divisée en plusieurs morceaux pouvant correspondre aux différentes phases de la maladie ou d'un traitement. Nous nous montrons qu'il est possible d'estimer les trajectoires constituées de plusieurs morceaux, et de sélectionner le nombre de ruptures le mieux adapté pour décrire l'évolution moyenne de la population.

Abstract

The development and massification of medical imaging and clinical followup databases open up new perspectives for understanding complex phenomena such as ageing or neurodegenerative diseases. In particular, brain connectivity, i.e., the study of connections and interactions between brain regions, can now be studied on the scale of a population scale rather than on an individual basis. This framework offers the possibility of better taking into account individual specificities in the development of monitoring tools.

In this thesis, we first propose new approaches to model and understand the variability of brain connectivity within a group of subjects. More generally, we are interested in collections of networks where each network describes interactions between the same entities. We rely on the empirical low rank property of the adjacency matrices of these networks to account for their distribution. We propose two approaches, one variational and the other statistical, to account for the heterogeneity of these matrices. In particular, in the second case, we show that a limited number of parameters is sufficient to give a faithful and interpretable description of the variability of brain connectivity. We also show the theoretical consistency and identifiability of our approach.

In a second part, we study a longitudinal model for the progression monitoring of Parkinson's disease. In this model, the trajectory of each patient is divided into several pieces that may correspond to the different phases of the disease or of a treatment. We show that it is possible to estimate trajectories consisting of several pieces, and to select the number of breaks best suited to describe the average evolution of the population.

Remerciements

Au terme de trois ans de thèse, c'est avec plaisir et reconnaissance que j'emploie ces premières pages à remercier celles et ceux qui m'ont accompagné et soutenu dans mon parcours.

Mes remerciements vont tout d'abord à mon équipe d'encadrement. Stéphanie, un grand merci pour tes conseils, ton énergie et ta patience ! Stanley, nous avons eu moins d'occasions de nous voir ces dernières années, mais j'ai toujours beaucoup apprécié nos discussions, tes conseils et ton esprit critique. Merci ! Je réalise la chance que j'ai de vous avoir eus tous deux comme directeurs de thèse, et je vous suis sincèrement reconnaissant de m'avoir donné la possibilité d'étudier, travailler et apprendre à vos côtés au cours de ces trois dernières années.

I am grateful to Peter Hoff and Xavier Pennec, for accepting to review my thesis manuscript. Je remercie également Maud Delattre, Marc Lelarge et Éric Moulines pour avoir accepté de participer au jury de ma soutenance.

Je suis heureux de pouvoir remercier les chercheurs du CMAP avec lesquels j'ai eu l'occasion discuter et travailler, que ce soit dans le cadre du monitorat, d'un séminaire ou d'une pause café. Mon passage au CMAP m'a permis d'entrevoir l'incroyable diversité et la pluridisciplinarité des thématiques de recherche, des applications et des domaines mathématiques abordés au laboratoire ; cette richesse, que j'ai pu découvrir au travers de nombreux séminaires, a contribué à faire de ma thèse une expérience passionnante. Merci en particulier à l'équipe SIMPAS, et à Aymeric pour l'organisation des SGMs. Merci également à l'équipe administrative du CMAP, et notamment Nasséra et Alexandra, pour leur réactivité et leur disponibilité, et à Pierre Straebler pour son assistance tout au long de ma thèse. Merci enfin à Emmanuel Fullenwarth pour son accompagnement dans la préparation de ma soutenance.

Je remercie l'équipe Aramis, pour son accueil toujours chaleureux et nos échanges, et en particulier Baptiste Couvy-Duchesnes et Ninon Burgos pour leur assistance et leurs conseils, ainsi que Federica Cacciamani et Stéphane Epelbaum pour leur expertise en neuroscience et leur contribution à cette thèse. Merci à Hélène Milome d'avoir facilité mon intégration entre l'Inria, Polytechnique et l'ICM.

Merci aux doctorants du CMAP, et en particulier à mes (rares, mais d'autant plus appréciés) collègues et ex-collègues du bureau 2009, Vianney, Madeleine, Benoît, Alice, Kevish et les autres. Merci à l'équipe du séminaire des doctorants, Leila, Kevin, Yoann, Louis, Clément et Baptiste, pour l'organisation de séminaires toujours plus ponctuels, agiles et connectés. Merci également à l'équipe du CJC, dans le désordre : Solange, Apolline, Constantin, Claire, Pierre, Louis, Baptiste (encore), Guillaume B., Dominik, Corentin, Josué et Thomas. Merci à tous les autres, Eugénie, Guillaume C., Benjamin, Arthur, Margaux, Ignacio, et désolé à ceux que j'oublie ! Merci aussi aux doctorants de l'équipe Heka, et notamment à Clément, Fleur, Pierre, Louis et Linus, pour nos discussions toujours passionnantes (et bon courage à Pierre et Louis pour la relève de certains TPs). Plus généralement, merci à tous les doctorants qui ont eu la patience de (tenter de) m'expliquer leurs sujets de thèse et leurs travaux, avec un succès parfois mitigé, mais toujours dans la bonne humeur.

Je remercie chaleureusement le mathématicien et ami Godalle Marmanthier, pour nos discussions, ses conseils et son infatigable soutien, et auquel on prédit sans peine un avenir prometteur. Je remercie bien sûr Alexis, Induja, Florentin, Anne, Maxime, Guillaume, Éloïse, Pierre et Marine pour les multiples jeux, restaurants, randonnées, pique-niques et (re)lectures qui ont émaillé ces années de thèse. Merci à Arthur, Guillaume, Paul et Benoît pour les diverses péripéties qui ont amené les débris d'une batte à s'échouer sur les côtes d'une île lointaine. Merci également à Hugues, qui a échappé auxdites péripéties, sans pour autant démériter en matière de naufrages, de

haches et d'îles désertes. Je remercie les French Terrors, pour ce qu'elles apportent au quotidien au rayonnement de notre culture en France et ailleurs (et, accessoirement, pour les divers restaurants et cinémas sauvages). Je tiens par ailleurs à remercier l'ensemble des Clément qui ont, chacun à leur façon, apporté leur contribution au présent édifice, à commencer par votre serviteur, mais sans pour autant oublier le Petit Clément, le Grand Clément, et tant d'autres homonymes qu'un rigoureux effort de concision me contraint d'omettre. On dit parfois que l'union fait la force ; chez les Clément, nous l'avons bien compris, et profitons de notre multitude pour pallier la qualité par le nombre.

Mes remerciements vont enfin et bien sûr à ma famille, pour tout le soutien qu'elle m'a apporté au cours de ces années d'études, et qui sait bien qu'

À tant bon oint s'en faut, tout n'y vient que tréfaille.

Contents

Chapter 1

Résumé en Français

1.1 Modélisation pour les populations de réseaux

Le sujet central de cette thèse est la modélisation statistique pour les populations de réseaux, et plus particulièrement les réseaux de connectivité cérébrale. Dans cette section, nous motivons cette question et présentons brièvement les résultats obtenus dans les contributions de la thèse.

1.1.1 Analyse statistique pour les réseaux

L'analyse des réseaux est un domaine de recherche vaste et actif, qui vise au sens large à comprendre leur structure à l'aide de mesures et de modèles quantitatifs [Newman, 2012]. Nous définissons un réseau comme un ensemble d'entités, appelées *nœuds*, connectées entre elles par un ensemble d'*arêtes*. Un réseau comportant n entités est ainsi résumé par une *matrice d'adjacence* A de taille $n \times n$, où le coefficient A_{ij} donne l'intensité de la connexion reliant les nœuds i et j. De nombreux phénomènes peuvent être modélisés via ce formalisme: réseaux sociaux, graphes d'interactions entre espèces vivantes, réseaux informatiques, etc.

Dans cette thèse, nous nous intéressons plus particulièrement à l'application de l'analyse des réseaux pour l'étude du cerveau humain [Fornito et al., 2016]. Le développement des méthodes d'imagerie cérébrale (électroencéphalogrammes, imagerie par résonance magnétique nucléaire, etc.) a permis de spectaculaires avancées dans la compréhension du cerveau, notamment en mesurant les interactions entre les différentes régions du cerveau. Deux types de connexions sont considérés: d'un côté, la connectivité *structurelle* compte les fibres de matière blanche qui relient les régions du cortex entre elles, et mesure ainsi le lien physique entre les régions. De l'autre, la connectivité *fonctionnelle* mesure la corrélation entre les activités de chaque région ; autrement dit, on considère que deux régions sont très connectées si leurs activations sont synchronisées. Cette deuxième mesure de connectivité permet de comprendre comment les régions du cerveau interagissent les unes avec les autres, et décrit ainsi le fonctionnement du cerveau – d'où le nom de connectivité fonctionnelle.

L'étude de la connectivité cérébrale consiste à appliquer les outils de l'analyse des réseaux aux données de connectivité issues de l'imagerie cérébrale. Cette étude apporte un nouvel éclairage sur le vieillissement du cerveau et le développement des maladies neurodégénératives [Damoiseaux, 2017]. Dans ce contexte, on n'analyse non plus un unique réseau, comme c'est souvent le cas en analyse de réseaux traditionnelle, mais une population de réseaux, dont chaque élément correspond au réseau de connectivité cérébrale d'un individu. Ce cadre de travail est complexe : d'un point de vue statistique, il s'agit de modéliser des données de grande dimension (un réseau à n nœuds est décrit par une matrice A de n^2 coefficients) à partir d'un nombre restreint d'observations, la plupart des études en neurosciences disposant de quelques centaines de sujets.

L'approche la plus courante pour contourner cette difficulté et modéliser une population de réseaux consiste à résumer chaque réseau par un petit nombre de caractéristiques: par exemple, on peut résumer chaque matrice d'adjacence A par le nombre moyen de voisins de chaque sommet, ou encore par la longueur moyenne du plus court chemin d'un nœud à l'autre [Rubinov and Sporns, 2010, Harris, 2014, Ghosh et al., 2018]. Cette approche est simple et interprétable, et elle est

couramment utilisée en pratique dans les études de neurosciences. Cependant, cette simplicité a un coût : en résumant chaque réseau par ses caractéristiques, on se prive de l'arsenal des méthodes statistiques pour le traitement des données en grande dimension.

Dans cette thèse, nous travaillons directement sur les matrices d'adjacence des réseaux de connectivité, sans les résumer en une liste de caractéristiques. Nous proposons d'utiliser les propriétés naturelles de ces matrices pour construire des modèles simples et interprétables pour les populations de réseaux.

1.1.2 Une formulation variationnelle rang faible / parcimonie

Dans la première contribution de cette thèse (Chapitre 4), nous proposons une approche simple pour modéliser une collection de matrices d'adjacence $A_1, \ldots, A_N$. Nous nous appuyons sur les propriétés empiriques de ces matrices : pour chaque matrice, de très nombreux coefficients sont nuls et le rang de la matrice est faible (au sens où les valeurs propres décroissent rapidement). La première propriété reflète le fait que deux nœuds, pris au hasard, ont peu de chances d'interagir entre eux. La seconde traduit l'organisation modulaire des nœuds : par exemple, plus les nœuds tendent à se regrouper en modules clairement délimités, plus leur matrice d'adjacence se rapproche d'une matrice constante par blocs, dont le rang est en général faible.

En nous appuyant sur ces propriétés, nous postulons que chaque matrice de connectivité A_i est une déviation simple d'un terme commun de structure simple. Autrement dit, on suppose que l'on peut écrire $A_i = T + V_i + \varepsilon_i$, avec T un terme central commun à chaque matrice, V_i une déviation propre à chaque individu et ε_i un terme résiduel de faible amplitude. Selon les propriétés formulées au paragraphe précédent, dire que T et les V_i sont *simples* au sens des réseaux revient à supposer que ces matrices ont un rang faible et sont parcimonieuses (c'est-à-dire qu'elles ont de nombreux coefficients nuls). Le terme central T n'est pas nécessairement égal à la moyenne des matrices A_i : cette moyenne n'a pas de raison d'être parcimonieuse ou de rang faible. Il se peut également que les V_i soient à coefficients positifs, auquel cas leur moyenne empirique est également positive.

Nous proposons d'estimer les valeurs pour T et les V_i par une approche variationnelle, c'est-à-dire en résolvant un problème d'optimisation. Notre approche d'appuie sur les travaux de Richard et al. [2012, 2013], qui proposent précisément une méthode d'estimation d'une matrice parcimonieuse de rang faible. Nous étendons leur approche au modèle de décomposition détaillé au paragraphe précédent. Nous résolvons les problèmes d'optimisation issus de cette formulation via des méthodes d'optimisation convexe non-lisse [Parikh and Boyd, 2014], et nous validons les algorithmes obtenus sur des données simulées.

Nous utilisons enfin ces algorithmes sur des données réelles. Appliquée à un réseau de transport aérien et à des réseaux de connectivité fonctionnelle cérébrale, notre approche produit des résultats cohérents avec la structure de rang faible des matrices d'adjacence réelles ; les termes résiduels ε_i sont systématiquement très faibles comparés aux déviations simples V_i, suggérant que ce modèle de décomposition est adapté pour décrire des collections de matrices d'adjacence. Les valeurs des hyperparamètres du modèle laissent une grande flexibilité dans sa formulation : selon le choix, il est possible d'accorder plus ou moins d'importance à la parcimonie ou au rang faible.

1.1.3 Un modèle statistique pour la décomposition propre

Dans le Chapitre 5, nous étendons le raisonnement du chapitre précédent dans un cadre statistique: l'approche développée dans le Chapitre 4, bien qu'intéressante de par sa simplicité, ne définit pas un modèle statistique. Il est plus souhaitable de disposer d'un modèle génératif dont les réseaux observés seraient des échantillons : un tel modèle sert d'une part à mesurer et comprendre la variabilité au sein de la population, et d'autre part à réaliser différentes formes d'inférence (classification, clustering, imputation de données manquantes, régression, etc.). On s'intéresse ainsi aux modèles statistiques susceptibles de produire des matrices ayant naturellement une structure de rang faible, éventuellement parcimonieuse.

Plusieurs modèles co-existent dans la littérature scientifique pour aborder ce problème ; citons par exemple les auto-encodeurs de graphes [Kipf and Welling, 2016] et les modèles de dictionnaire [D'Souza et al., 2018]. Dans le Chapitre 5, nous nous sommes intéressés plus spécifiquement à une famille d'approches qui proposent de définir des distributions de probabilités sur l'ensemble

des matrices de rang faible. Cet espace (courbe) n'est pas explicite, et il est en pratique bien plus simple de définir la distribution d'une matrice aléatoire de rang faible A par celle de ses valeurs propres λ et de ses vecteurs propres X, puis de calculer $A = X^\top \mathrm{Diag}(\lambda) X + \varepsilon$, avec ε un bruit résiduel. Nous suivons cette approche, et définissons ainsi un modèle statistique sur les matrices de rang faible. Notre approche offre une grande flexibilité : changer la distribution des valeurs propres ou celle du bruit permet de modéliser simplement des matrices binaires ou à coefficients positifs, ou encore des matrices aux valeurs propres positives (dans ce cas particulier, d'autres modèles, plus adaptés, tirent parti de la géométrie de l'espace des matrices positives).

La définition de notre modèle statistique fait appel aux outils de la géométrie Riemannienne : la variable aléatoire des vecteurs propres X satisfait en effet une contrainte non-linéaire (les colonnes de la matrice sont unitaires, orthogonales deux à deux). Il est donc impossible de modéliser X par une distribution Gaussienne classique, qui produirait des valeurs en dehors de l'espace contraint. Cependant, des alternatives spécifiques à cet espace ont été développées, et remplissent en première approximation le rôle des distributions Gaussiennes – au sens où elles se définissent par un paramètre de « moyenne » et un paramètre de « variance ». Ces distributions ont fait l'objet de nombreuses études, et des méthodes efficaces permettent d'en estimer les paramètres [Khatri and Mardia, 1977, Kume et al., 2013]. Dans le Chapitre 5, nous utilisons ces résultats, ainsi qu'une nouvelle méthode d'échantillonnage dans l'espace contraint, pour estimer les paramètres de notre modèle à l'aide d'un algorithme d'Espérance-Maximisation stochastique [Kuhn and Lavielle, 2004, Allassonnière et al., 2010]. Nous validons l'implémentation de notre algorithme et fournissons des exemples simples permettant de comprendre son fonctionnement. Toujours sur données simulées, nous montrons que le modèle proposé permet de prédire efficacement la valeurs de coefficients manquants, et d'effectuer des tâches de clustering sur des collections de matrices.

Nous appliquons ensuite notre méthode à une collection de réseaux de connectivité fonctionnelle cérébrale issus de l'UK Biobank [Littlejohns et al., 2020]. Nous montrons que le modèle que nous proposons produit une représentation des réseaux à la fois fidèle et interprétable. Les termes moyens de la décomposition de rang faible font intervenir des groupes de régions bien identifiés, qui interagissent de façon simple. Les termes de variance permettent de quantifier la variabilité de chaque type d'interaction d'un individu à l'autre. Un des avantages du modèle réside dans sa parcimonie : il utilise un nombre restreint de paramètres pour décrire la distribution des matrices, ce qui assure que ces paramètres peuvent être estimés précisément, même à partir de peu de données. La fidélité du modèle est mesurée par plusieurs indicateurs, notamment la portion de variance capturée et la qualité de prédiction de données manquantes.

Bien que simple dans son interprétation, le modèle du Chapitre 5 fait intervenir des objets complexes (modèles hiérarchiques non linéaires et variétés Riemanniennes), qui rendent difficile une appréhension intuitive de la qualité de l'estimation obtenue par l'algorithme. Dans le Chapitre 6, nous donnons des éléments de réponse à cette question. Nous montrons que le modèle est identifiable, c'est-à-dire que deux jeux de paramètres différents produisent toujours deux modèles différents. Nous nous intéressons ensuite aux propriétés de l'estimation du modèle lorsque le nombre de réseaux observés devient grand, en nous appuyant sur la théorie des statistiques asymptotiques [van der Vaart, 1998]. En mettant de côté l'erreur d'optimisation de l'algorithme, nous prouvons que les paramètres du modèle convergent presque sûrement vers leur valeur optimale. Pour une version restreinte du modèle, nous retrouvons le résultat classique de normalité asymptotique de l'estimateur. La difficulté technique provient de la structure hiérarchique de notre modèle, qui complique l'expression de la vraisemblance des données et de l'information de Fisher.

Ces travaux ouvrent plusieurs perspectives d'améliorations et d'extensions. D'une part, la précision de l'estimation pourrait être améliorée en utilisant des méthodes Bayésiennes récentes adaptées à notre cadre de modélisation. D'autre part, le modèle que nous proposons pourrait être étendu à de nombreuses configurations plus complexes, dont par exemple la régression et la multimodalité. En particulier, une extension intéressante est la modélisation de l'évolution temporelle des réseaux de connectivité cérébrale.

1.2 Modèles longitudinaux segmentés

Dans la dernière contribution de cette thèse (Chapitre 7), nous changeons de perspective et étudions un modèle de population plus mature pour des données longitudinales – c'est-à-dire des données où chaque sujet est observé plusieurs fois au fil du temps. Bien que ce contexte diffère du contexte central de l'analyse des réseaux de connectivité cérébrale, les contributions présentées dans cette thèse sur les populations de réseaux pourraient être utilisées comme base pour une modélisation longitudinale de la connectivité cérébrale. Dans cette mesure, le développement d'une solide compréhension du fonctionnement et des capacités des modèles longitudinaux est une étape cruciale.

1.2.1 Modèles à effet mixtes pour les données longitudinales

L'approche standard en statistiques pour les données longitudinales est le cadre des modèles à effets mixtes. Il décrit l'évolution de chaque patient au fil du temps comme une fonction de ses caractéristiques connues (sexe, facteurs génétiques, etc.) – les effets fixes – et de facteurs inconnus – les effets aléatoires. Estimer les paramètres d'un modèle à effets mixtes permet ainsi de comprendre le rôle des effets fixes et de quantifier l'impact des effets aléatoires [Lavielle, 2014]. De nombreux travaux considèrent par exemple des modèles à effets mixtes linéaires, où les calculs sont simples et l'estimation se trouve facilitée. Au cours des deux dernières décennies, plusieurs travaux ont montré que les modèles non-linéaires peuvent également être estimés, et offrent des outils plus riches pour modéliser les évolutions longitudinales.

En particulier, nous nous plaçons dans cette thèse dans le cadre de modélisation longitudinale proposé par Schiratti et al. [2015]. Cette approche consiste à décrire une population de sujets en deux composantes. D'une part, une trajectoire moyenne est paramétrée, pour décrire l'évolution représentative d'un sujet moyen de la population au fil du temps. D'autre part, des facteurs individuels quantifient la déviation entre la trajectoire de chaque sujet et la trajectoire moyenne – la trajectoire individuelle peut se situer à un autre point de l'espace et du temps, se dérouler à une vitesse différente de la trajectoire moyenne, etc. L'estimation du modèle consiste alors à déterminer simultanément la trajectoire moyenne de population et les déviations individuelles. Ce découplage population-individu a l'avantage d'une grande interprétabilité : la trajectoire de population peut être interprétée en elle-même, et les déviations de chaque individu informent sur la variabilité autour de la moyenne.

Cette approche a permis en particulier de modéliser la progression de maladies neurodégénératives comme la maladie d'Alzheimer ou la maladie de Parkinson [Koval, 2020, Couronné, 2021]. Les initiatives de suivi de population à grande échelle ont permis de collecter des grandes quantités de données de diverses natures (scores cliniques, imagerie cérébrale, etc.) en parallèle du suivi des patients. L'objectif des modèles statistiques est d'utiliser ces données pour anticiper le développement de la maladie, afin d'adapter les traitements aux patients en fonction de leurs caractéristiques individuelles et des observations déjà enregistrées.

1.2.2 Estimation et sélection de modèle pour les trajectoires segmentées

Dans le Chapitre 7 de cette thèse, nous prolongeons les travaux de Chevallier et al. [2021] sur un modèle longitudinal segmenté, autrement dit un modèle longitudinal où la trajectoire de population est définie en plusieurs morceaux – par exemple, la trajectoire peut être une fonction affine par morceaux. Ce modèle permet d'étudier les cas où la dynamique de chaque sujet est susceptible de connaître une ou plusieurs ruptures. La principale application est le suivi de patients suivant un traitement, par exemple une chimiothérapie. Les ruptures peuvent correspondre à l'apparition des symptômes, leur atténuation suite à la prise du traitement, ou leur éventuel retour en cas de une rechute.

Nous nous intéressons au suivi de patients de la cohorte PPMI (Parkinson's Progression Markers Initiative), dont l'état est mesuré par un ensemble de scores cliniques. Nous modélisons l'évolution de ces scores par des trajectoires affines par morceaux, et étendons l'estimation proposée par Chevallier et al. [2021] au cas de plusieurs ruptures. En particulier, nous nous intéressons à la sélection du nombre de ruptures dans la trajectoire moyenne. Nous montrons que les critères de

sélections de modèles classique (et plus précisément leur adaptation au cas des modèles à effets mixtes) permettent, sur des données synthétiques, de sélectionner le nombre correct de ruptures jusqu'à trois ruptures, même en présence d'un bruit élevé et d'une forte proportion de données manquantes. Nous montrons que ces critères peuvent être calculés en pratique à l'aide de méthodes Monte-Carlo basées sur le principe d'échantillonnage préférentiel [Gronau et al., 2017].

Nous appliquons notre méthode à la cohorte PPMI, et étudions les résultat obtenus selon les groupes de sujets. Le modèle décrit fidèlement l'évolution des patients dans la phase prodromiques de la maladie de Parkinson, c'est-à-dire la phase d'apparition de symptômes avant-coureurs de la phase principale de la maladie. Les résultats au niveau de la moyenne de population sont plus difficiles à interpréter pour les patients qui suivent un traitement : chaque individu commence son traitement à un stade différent de progression de la maladie, ce qui complique l'obtention d'une représentation moyenne interprétable. Néanmoins, les trajectoires individuelles demeurent relativement bien décrites par leur estimation du modèle statistique.

Chapter 2

Introduction

Contents

2.1 From network analysis to population models

The central question that motivated this thesis is the design of statistical models for populations of networks, with a focus on brain connectivity. In this first section, we motivate this question by introducing first notions in network analysis and its application to brain networks. We delineate the core topic of this thesis, by showing how existing network models are used to understand network structures and build interpretable representations, and how these representations can be understood from a population modeling perspective.

First, we give a brief introduction to the concepts of network analysis used in this thesis. Next, we provide an overview of how network analysis can be leveraged to better understand brain networks. Finally, we move to the central topic, i.e., statistical analysis of populations of networks. We refer the reader to Newman [2012] for an introduction to network analysis, and to Fornito et al. [2016] for an introduction to brain network analysis.

2.1.1 Network analysis

From an abstract perspective, a network – or graph – is the summary of interactions among a set of distinct entities. Entities are called *nodes*; if two nodes interact together they are connected by an *edge*. For binary networks, where each interaction is either one or zero, the complete structure is thus characterized as a set of nodes V and a list of edges $E \subset V \times V$. In this thesis, we will be studying undirected networks, i.e., networks such that $(i, j) \in E \iff (j, i) \in E$. The edges are often also given a weight in real world applications – e.g., the distance between cities, or the number of interactions between persons. In this case, the network is described by a square *adjacency matrix* of weights A, where the coefficient A_{ij} gives the interaction between node i and node j. More recently, the structure of interaction networks tends to be enriched with additional information: each node and each edge may have a label or a set of features

The field of network analysis seeks to understand the properties of networks arising in very diverse applications. For instance, the interactions between computers in a network, species in an ecological system, users in a social network or cortical regions in a human brain all share this common graph structure. The development of general tools for network analysis thus has a very large impact on very diverse fields of application. Essentially, the goal is to analyze the structure

of the interactions, i.e., to what extent they can be described with simple patterns rather than pure randomness.

Remark. The term *graph* often refers rather to a theoretical object, whereas the term *network* rather designates real entities and their connections. In this thesis, we will be focusing more on the latter concept, but both terms are employed, depending on the context, essentially designating the same objects.

Graph parameters. From this perspective, a fundamental approach is to compute summary indicators which help analyze the network structure. For instance, in a social network, determining the maximal distance between two individuals, or the average distance, provides a description of the global structure. It may contribute to the understanding of fake news diffusion, and how to prevent it. Other parameters, like the numbers of cliques or the clustering coefficient, measure the extent to which the nodes tend to gather in groups. More generally, the frequency of patterns like cycles or other subgraphs provide useful information on the network structure. The topology of the networks can also be studied: some indicators measure the number of connected components (or groups of strongly connected nodes), or the similarity with multidimensional lattices. Other measures focus on the properties of individual nodes, e.g., in order to determine which nodes act as "hubs", or which nodes are the most important to preserve connectedness. We refer the reader to Rubinov and Sporns [2010] for an exhaustive review on network measures, with a focus on brain network analysis.

These measures are used for two purposes. On the one hand, they summarize very complex interactions into simple, interpretable indicators which give precise information on the structure of the network. This information can be used without specific mathematical knowledge, either to analyze a given network or to understand the differences between several networks, e.g., a young brain and an old brain. On the other hand, from a statistical and probabilistic perspective, these measures are used as features to design models relying on them and to theoretically compare different network models. We will discuss this point further in Section 2.1.3 on populations of networks.

Stochastic Blockmodel. One of the most important goals of network analysis is to help identify communities in a population given the interactions between its members. This problem has countless applications in very diverse fields, and many approaches have been developed to tackle it – see Khan and Niazi [2017] for a meta-review. From a statistical modeling perspective, the Stochastic Blockmodel (SBM) and its variants are the main approach to model a set of communities. The SBM is a simple interpretable generative model, which assumes that the nodes are independent samples from the same distribution. Each node i first randomly chooses a community $c(i) \in \{1, \dots, K\}$. Then, for each pair of nodes (i, j), the interaction between nodes i and j is determined by a Bernoulli random variable with probability $P_{c(i),c(j)}$. In other words, the probability of nodes i and j being connected depends exclusively on their respective communities. Alternatively, the SBM may generate weighted networks by drawing the interaction coefficient between i and j from a distribution with continuous support, parameterized by a coefficient $W_{c(i),c(j)}$.

The main purpose of the SBM lies in the estimation of its parameters given an input matrix. The labels of the nodes are the unknown variables of interest; they are estimated by maximum likelihood, along with the sizes of the communities and their interaction coefficients. The estimation of the SBM is a well-studied problem, and has been addressed in numerous setups like weighted networks, multiscale communities, time-varying networks or Bayesian SBMs [Aicher et al., 2015, Ho et al., 2012, Zhang et al., 2020a, Peixoto, 2020]. It provides a sound theoretical baseline for community detection methods. The core assumption is that the nodes are exchangeable in distribution, i.e., no node plays a unique, specific role determined in advance.

Modeling very large networks. When networks exhibit no clear community structure, the SBM fails to accurately describe the interactions because of its simplicity. From a theoretical perspective, this limitation can be lifted by considering an SBM with infinitely many communities, e.g., one community per real number $x \in [0, 1]$. The nodes label $c(i)$ is then drawn from a uniform distribution on $[0, 1]$. The interaction – weighted or binary – between nodes i and j is determined

by a coefficient $w(c(i), c(j))$, with $w : [0,1] \times [0,1] \to \mathbb{R}_+$ a symmetric function. The function w is called a *graphon*; it can be seen as a limit of the SBM, in the sense that the random adjacency matrices it produces can be seen as a limit in distribution of SBMs with an increasing number of communities [Lovász, 2012]. Graphons also strictly generalize SBMs, which can be seen as piecewise constant graphons. Estimating a graphon amounts, given one or several adjacency matrices, to retrieving the function w as well as the node labels, i.e., their coordinates in $[0, 1]$. As with the SBM, the graphon estimation is a well studied problem. Several efficient algorithms can be used with theoretical convergence guarantees, depending on the context [Wolfe and Olhede, 2013, Gao et al., 2015, Cai et al., 2015, Sischka and Kauermann, 2022]. Because of its great flexibility, the graphon model has received increased attention in the literature over the last decade, with theoretical results on the topology of graphon space and connections with the practical estimation of network measures or the graph isomorphism problem [Latouche and Robin, 2016, Zhang, 2018]. Figure 2.1 shows three examples of graphons representing different types of networks. The first column shows an SBM which represents three communities and their interactions. The second column shows a graphon with a smooth "core-periphery" structure, i.e., a network structure with a subset of highly connected nodes (the core) interacting with the other nodes (the periphery), with the periphery nodes sparsely interacting with each other. The third column accounts for a network structure with a line topology, where nodes tend to interact only with their topological neighbors on their left and right.

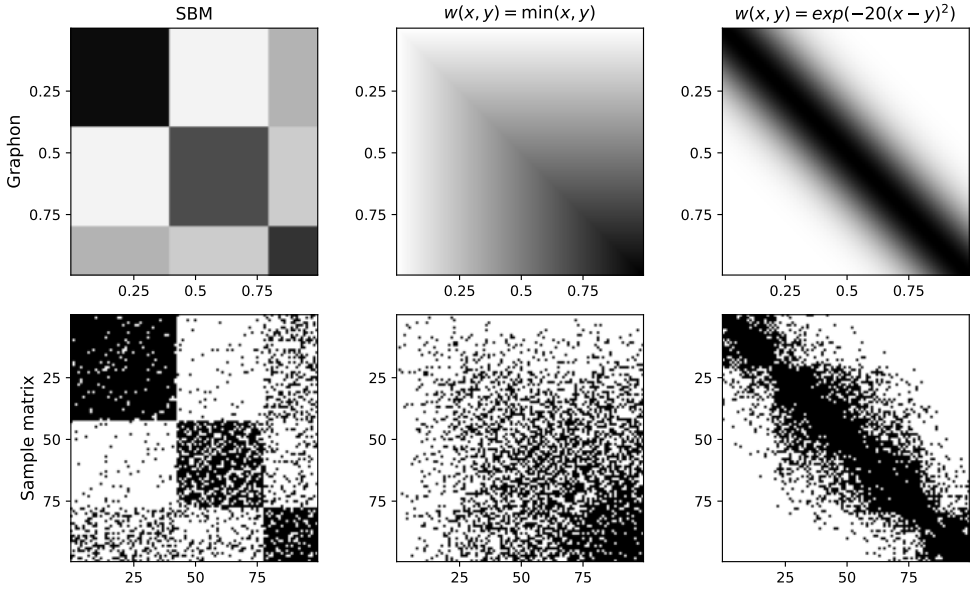

Figure 2.1: Example of three graphons (top row) and sample graphs with 100 nodes (bottom row). Darker shades represent a connection (or a strong interaction probability).

Low rank structure of network matrices. In most real-world networks, the adjacency matrices have relatively low rank, in the sense that a few eigenvalues dominate the remaining others. This phenomenon is well understood, and has been used extensively to build robust estimation procedures, e.g., to predict missing links [Martínez et al., 2016]. When analyzing single networks, the low rank property also allows clustering the network nodes with spectral clustering. In this thesis, we will be relying on this property to design models for populations of networks.

The low rank structure of network adjacency matrices reflects the fact that they often represent interactions with a strong structure. For instance, in the Stochastic Blockmodel shown in Figure 2.1, the nodes of the first community all interact with each other, and have almost no interaction with the nodes in the second community. If, in first approximation, the adjacency matrix was replaced by the graphon value at each pair of nodes, we would obtain a matrix with rank only three; the Bernoulli sampling step only perturbs this overall structure, so that the observed adjacency

matrix has rapidly decaying eigenvalues. More generally, a low rank is observed in networks with a highly modular structure, i.e., in networks where the nodes tend to group in clusters with simple interactions.

The low rank property may also arise for graphons with an overall smooth structure. Consider a smooth symmetric function $w : [0,1] \times [0,1] \to \mathbb{R}$. The function w defines a compact Hilbert-Schmidt operator $T_w(f)(x) = \int w(x,y)f(y)\mathrm{d}y$. Considering the eigenvectors $(\varphi_k)_{k\geq 0}$ of T_w and its eigenvalues $(\lambda_k)_{k\geq 0}$ (sorted by decreasing absolute magnitude), we have the decomposition [Lax, 2002]:

$$w(x,y) = \sum_{k=0}^{+\infty} \lambda_k \varphi_k(x)\varphi_k(y)\,.$$

In general, for smooth functions w, the eigenvectors φ_k are oscillating functions, with the frequency of the oscillations increasing with k. The smoothness of w directly relates to the rate of decay of the eigenvalues λ_k. In particular, a smooth graphon can be well approximated by keeping only the first p terms of the decomposition:

$$w(x,y) \simeq w_p(x,y) = \sum_{k=0}^{p} \lambda_k \varphi_k(x)\varphi_k(y)\,.$$

As a consequence, when sampling uniform nodes $x_1, \ldots, x_n$ on $[0,1]$, the obtained weights matrix $W = (w(x_i,x_j))_{i,j}$ can be well approximated by $W_p = \Phi_p^\top \Lambda_p \Phi_p$, with $\Phi_p = (\varphi_k(x_i))_{k\leq p, i\leq n}$ and $\Lambda_p = \mathrm{Diag}(\lambda_1, \ldots, \lambda_p)$. The final adjacency matrix A is obtained, in the case of binary networks, by $A_{ij} \sim \mathcal{B}(W_{ij})$ (and, for instance, by an exponential distribution $A_{ij} \sim \mathrm{Exp}(W_{ij})$ for weighted networks). The eigenvalues decay can of A can be related to that of W, which is itself close to W_p. As a consequence, since W_p has rank $p \ll n$, the final sampled adjacency matrix A also has a relatively low rank, with a restricted number of eigenvalues that are significantly larger than the remainder.

The low rank property also allows interpreting the interactions between the nodes from a simple latent space perspective. Consider an $n \times n$ symmetric adjacency matrix A with rank $p \ll n$. The spectral theorem gives that A can be written as $U^\top \mathrm{Diag}(\lambda) U$, with $U \in \mathbb{R}^{n\times p}$ and $\lambda \in \mathbb{R}^p$. This relation can be rewritten, for all nodes i and j, as

$$A_{ij} = \sum_{k=1}^{p} \lambda_k U_{ik} U_{jk}\,.$$

In other words, the dependency between nodes i and j is obtained as a (linear) function of low-dimensional vectors U_i and U_j, with U_i denoting the i-th row of U. As we will see in Section 2.1.3, this property suggests designing models representing network nodes by the vectors U_i. This representation is robust, in the sense that small perturbations of A mostly affect the direction of the eigenvectors related to small eigenvalues.

2.1.2 Brain network analysis

The techniques of network analysis have been successfully applied to understanding the interactions between regions of the brain. This was made possible thanks to the progress of medical imaging, especially electroencephalography and Magnetic Resonance Imaging (MRI). Brain network analysis is a rapidly evolving field; it greatly contributes to the understanding of the brain functions, and of the development of neurodegenerative diseases like Alzheimer's disease.

Measuring the connectivity. The brain is essentially composed of two parts. The cerebral cortex, also known as gray matter, is the outer layer of the brain where the neurons are located. The white matter is the inner part of the brain, and consists of axons connecting the different regions of the cerebral cortex. Neuroimaging devices allow visualizing these components, as well as other structures of the brain. The types of imaging modalities divide in two main categories [Bigler, 2013]. On the one hand, structural (or anatomical) imaging retrieves static images of the brain. On the other hand, functional imaging records the brain activity over a given period of time, and

provides a dynamic view on the role of each brain region. In each of these two categories, several techniques are used, with their own advantages and drawbacks in terms of spatial and temporal resolution, but also device cost and ease of use. In this thesis, for the brain networks applications we will mostly be relying on MRI data, which provides the highest spatial resolution. Combining all available information on the brain regions, their connections and their interactions results in a global structure called the human *connectome*. For this reason, the analysis of brain networks is also called connectomics.

Structural connectivity. Structural MRI – in particular T1 and T2 MRI – provides 3D images of the brain, and enables to retrieve the shape and the size of each gray matter region. It is also a powerful tool to analyze the *structural connectivity* of the brain, i.e., how the regions of the brain are connected through fibers of white matter. Using the result of the diffusion weighted MRI, which gives the diffusion direction of water molecules in the brain, it is possible to reconstruct the individual fibers of white matter through which the water molecules move. In turn, counting the number of white matter fibers provides information on how the brain regions are physically wired to each other.

Functional connectivity. In contrast, Functional MRI (fMRI) provides a more dynamic view of the brain function. It records, at each point (voxel) of the brain, the evolution of the blood oxygenation level across time. In turn, this level is directly connected to neuronal activation, which consumes oxygen. Averaged at the level of each brain region, fMRI time series thus provide details on when each brain region activates. This information is used by neuroscientists to study the temporal interaction between the activation of the brain regions. By computing, for each pair of brain regions, the correlation between their activation time series, a correlation matrix is obtained. It summarizes the interaction of the brain regions into simple, interpretable coefficients: if the correlation between two regions is positive, then the regions tend to activate together. On the contrary, if it is negative the regions tend to activate in distinct periods of time. This matrix provides information on how the brain regions function together; it is thus called a functional connectivity matrix.

The analysis of the brain functional connectivity reveals that consistent associations within groups of brain regions are present across subjects and studies [Damoiseaux et al., 2006]. These subsets of regions form so-called "functional brain networks". Understanding the role and the interactions between functional brain networks is a major focus of the field of brain connectivity analysis. Depending on the task and the subject, different functional networks activate with varying intensities. As an example, the default mode network was the first functional brain network to be identified from fMRI data analysis. It gathers regions active when the subject is at rest with their mind wandering, and it is thus particularly well identified from resting-state fMRI. The functional connectivity is affected by the underlying structural connections between the brain regions; in that regard, the study of the relations between structural and functional connectivity is a complex question, which remains an active research topic to this day [Hermundstad et al., 2013, Tewarie et al., 2020, D'Souza et al., 2021a].

Questions arising from brain network analysis. The study of the brain connectivity, and its connection to individual characteristics – age, sex, activity, neurodegenerative diseases, etc. – has received increasing attention in the literature over the last decades. This interest has followed the development of large databases and the steady improvements in the image quality and resolution. Network analysis gives a rich perspective on brain networks, with many network measures providing complementary information on the interactions between the regions of the brain [Bullmore and Sporns, 2009]. It highlights the core-periphery structure of the brain, with a few regions playing a central role in signal propagation. Further analysis emphasizes the trade-off between segregation (regions clustering into groups with sparse interactions) and integration (cross-cluster region interactions which allow integrating information across different subnetworks). This trade-off results in a small-world architecture, where the regions are connected to each other by very short paths through "core" areas.

The brain architecture evolves with time. As the subjects get older, the connectivity and the local efficiency – a local measure of information propagation efficiency – decrease, increasingly

favoring integration over segregation. It is theorized that the increased integration allows mitigating the impact of degradation to preserve the overall efficiency. In general, except for the visual network, the connectivity within functional networks decreases with age, especially the default mode network [Zonneveld et al., 2019]. This overall decrease of functional connectivity is correlated to cognitive decline. In contrast, the connectivity between the functional networks increases with age, e.g., between the default mode network and the fronto-parietal network or the dorsal attention network [Damoiseaux, 2017].

The mechanisms of brain aging are complex, with an important heterogeneity at the individual level. They also interact with the development of neurodegenerative diseases: for instance, Parkinson's disease and Alzheimer's disease affect the structural and functional connectivity. For this reason, the development of disease progression models – and more generally population models – for brain networks is a question of immediate practical interest. In the case of aging, as noted by Damoiseaux [2017], no large longitudinal study has yet been carried out. To our knowledge, the collection of such data is still an ongoing process, and it is likely that further studies on this matter can be conducted in a few years.

2.1.3 Statistical modeling for populations of networks

The studies on brain network analysis roughly form two categories: those focusing on the network structure represented by the adjacency matrix, and those considering adjacency matrices as vectors. The first approach allows for a fine understanding of the network properties, whereas the second approach naturally lends itself to involved statistical modeling and inference.

In both cases, the goal is to extract information from a data set of adjacency matrices representing brain networks. This objective is twofold: on the one hand, we wish to identify average structures and trends. On the other hand, we also need to characterize the individual heterogeneity. This double perspective is at the core of *population models*, which define a form of average behavior and explain how each individual deviates from this average. This modeling aims at capturing the variability in the data, i.e., explaining how and why each individual observation differs from the arithmetic average of all samples.

In the context of network analysis, population models need to be tailored: the observations often have a large dimension, comparable to the number of subjects (typically, a brain divided into $n = 20$ regions produces a symmetric connectivity matrix with $n(n+1)/2 = 210$ distinct coefficients; many neuroimaging studies have less than a hundred subjects). In this context, most simple generative models have either a simplistic form or too many parameters to be estimated in practice from such low numbers of samples (e.g., a Gaussian distribution would require $210 + 210^2 = 44310$ coefficients). This section reviews the various categories of models that have been proposed and used in the literature over the last two decades to design population models accounting for the specific structure of network adjacency matrices. This review does not claim to be exhaustive: the field of network population analysis is very heterogeneous, in particular for brain network analysis, with numerous approaches coexisting in the literature, sometimes only slightly differing in methodology. Rather, we attempt to give an overview of the general lines of research that have been proposed, and to paint a big picture of the evolving state of the art.

Single-network exchangeable models

We start by emphasizing the limits of the models primarily designed to handle a single – possibly very large – graph. Considering e.g., the Stochastic Blockmodel, the first idea to model a population of adjacency matrices would consist in considering each matrix as an i.i.d. sample of a single SBM. The same remark can be made for graphons. This idea is especially relevant to handle a collection of networks where each network has its own set of nodes, different from the other networks; in particular, in such collections the number of nodes may differ from one network to another. For instance, graphons could be used to model a collection of computer networks; SBMs could be used to describe the interactions between random samples of a large population organized in clusters.

From a similar perspective, probabilistic models like the Barabási-Albert model [Albert and Barabási, 2002], the non-linear preferential attachment model [Kunegis et al., 2013] or the Erdős-Rényi model [Erdős and Rényi, 1959] – which is a special case of SBM – provide very simple

data generating mechanisms such that the nodes play exchangeable roles. As with the SBM, the generation procedure depends on a very restricted number of interpretable parameters.

These models were designed to provide a basic understanding of phenomena arising in real-world networks like communication networks, transportation networks or social networks. Their simplicity grants both a transparent data generation mechanism and a strong theoretical understanding. This simplicity arises partially from the exchangeability of the nodes, which limits the amount of information that can be added to the data generation mechanism. For these reasons, these models are also not well suited to analyze populations of networks where the nodes remain the same from one network to another, with changes only in the edges or their weights – this is typically the case for brain networks. Applied to a data set of networks, they would provide a simplistic view of the population heterogeneity. As an example, an SBM would by default assume that each network is generated with different nodes. But if we impose that the node labels remain the same from one network to another, the only difference between sampled networks is the random independent Bernoulli sampling step (or weighted edge sampling for weighted networks). In other words, if the nodes labels ($c(i)$) are considered fixed variables, the covariance between two coefficients A_{ij} and A_{kl} is zero. This is unrealistic from a practical perspective: for instance, the coefficients in a row or column of an adjacency matrix should not be considered independent, as they all depend on the state of the same node.

For these reasons, new different models are required to account not only for the structure of the networks in the population, but also for the variability in the population. Very diverse approaches have been proposed to this problem. In this introduction, we roughly divide them into three categories: models relying on graph features, on distances between graphs and on the low rank structure of the adjacency matrices.

Remark. Recently, Chandna and Maugis [2020] proposed an extension of the graphon model to better handle populations of networks. They define a so-called multi-graphon $w(x, y, z)$, where the variable z is a latent state which controls the network structure. Each network is generated by sampling a value of z, and sampling nodes and edges from the graphon $w(\cdot, \cdot, z)$.

Feature-based population models

Several network population models proceed by summarizing each network by a set of interpretable features. These features are then used as input in statistical models to perform inference.

ERGM. First, Exponential Random Graph Models propose to define the probability to observe an adjacency matrix A by a linear combination of features: up to a normalizing constant, they write as $p(A) \propto \exp(\theta_1 f_1(A) + \cdots + \theta_L f_L(A))$. Each coefficient θ_k represents the weight of the feature f_k in the probability $p(A)$: if θ_k is high, then the feature f_k plays a major role to discriminate which matrices are likely to occur, and vice-versa [Harris, 2014]. The features f_k are defined by the user, and often represent graph features like the average degree, the average shortest path length, the clustering coefficient or the frequency of patterns (cliques, cycles, etc.). The simple expression of the probability results in a highly interpretable model, and it is widely used in practice, in particular for brain network analysis [Fornito et al., 2016, Obando and Fallani, 2017].

Kernel methods. A more inference-oriented approach relies on graph kernels. Kernel methods allow extending classical vector-based statistics and machine learning algorithms to handle more complex objects, and in particular graphs. By defining a positive semi-definite similarity measure between networks, they enable to use methods like ridge regression on data sets of networks. Many cost-efficient procedures can be used to define a graph kernel [Ghosh et al., 2018]. Most of them can be seen as representing the network by a – possibly infinite – collection of interpretable features. Kernel methods are mostly suited for regression or classification purposes. They are widely used in brain network analysis, especially for the second goal, e.g., in order to detect mild cognitive impairment Takerkart et al. [2014], Jie et al. [2016], Kong et al. [2022].

Topology-based methods. A recent line of work suggests that topological features can be of interest to analyze populations of brain networks [Chung et al., 2015, 2017b, 2019, Li et al., 2020a].

In these papers, the authors propose to represent the topology of brain networks with persistent homology. In practice, this consists in counting the number of independent cycles and connected components in the network for a given threshold on the weights. The evolution of these numbers as the threshold grows gives curves which characterize the network topology. The authors show that the distribution of the obtained topological features across the population helps understand changes in the brain organization, e.g., for child maltreatment or the resemblance between the brains of twins.

Remark. Feature-based models are overall interpretable methods to handle populations of networks. They go further than single-network models, as they provide tools to analyze and leverage the variability in the population. However, they can only provide representations based on the computed features, which inherently restricts their capacities. For instance, drawing samples from an ERGM calibrated on brain networks is unlikely to produce matrices that look like brain networks: the small amount of features used in ERGMs is insufficient to describe the shape of a matrix-valued distribution. Graph kernels do not allow to compute the average of several graphs (or at least the average is not a graph in itself), which prevents from building statistical models relying on them. Summarizing the networks as a set of features provides advantages in terms of interpretability and robustness, but it also limits the possibilities of generative modeling.

Distance-based models

Although the Gaussian distribution, which endows the space of adjacency matrices with an elliptic geometry, is a poor choice to model data sets of networks, alternative models have been proposed, relying on non-Euclidean distances to compare networks and define the mean of graphs. In particular, specific approaches have been developed to define distances between networks invariant to node permutations.

From a geometric perspective, defining a distance d on a space of networks implicitly allows for generalized Gaussian distributions $p(G) \propto \exp(-d(G, \bar{G})^2/2\sigma^2)$, with a mean $\bar{G}$ and a variance σ^2. In particular, if a Riemannian manifold structure can be defined on an abstract space of networks, it enables to consider Fréchet means and covariances using the induced Riemannian distance [Pennec, 2006]. Such an approach is being developed by Calissano et al. [2020, 2022]. The authors define a graph space by taking the quotient of labeled adjacency matrices with a given size under node permutations. The resulting space has its own distance function, and it can be used to perform principal geodesic components analysis or regression. In a similar line of work, Lunagómez et al. [2021] and Zhou and Müller [2021] propose computing Fréchet means and regressions for adjacency matrices and graph Laplacians; they use custom distances between matrices with the same number of nodes.

Chung et al. [2017a] use persistent homology on networks as described in the previous section to define topological distances between networks. These distances, like the homology, are permutation-invariant and measure structure similarities. They provide higher-order characterizations, in the sense that they do not require that the networks have the same number of nodes.

In graph theory, the *cut distance* is used to compare adjacency matrices with different numbers of nodes [Lovász, 2012, Janson, 2013a]. It is intrinsically defined up to permutation, in the sense that it writes as an infimum over node permutations. It is particularly well suited to theoretical analysis, as it can be extended to define a distance between graphons, which are defined up to "node permutation", i.e., up to measure-preserving bijections from $[0, 1]$ to $[0, 1]$ [Khetan and Mj, 2018]. However, to the best of our knowledge, it has not yet been used as a tool to model populations of networks. In particular, computing the Fréchet mean of a set of networks or graphons remains an open problem. Although out of the scope of this thesis, we believe that this question is interesting, and that it could provide a theoretically-grounded reference point for other permutation-invariant averaging methods.

Low-rank network models and embedding methods

Finally, low-rank and embedding methods have recently drawn significant attention in machine learning and in network analysis applications. They provide a more applied perspective on the network modeling problem, as they naturally connect to regression, clustering and classification

problems. We divide them into three methodological families (very uneven in sizes): node embeddings, dictionary models and low-rank population models. Most of the approaches we present were designed for brain networks or have been used to analyze them.

Node embedding methods. A first category of models computes a low-dimensional vector for each node of the network, representing its features and its interactions with the other nodes. Some node embedding methods are designed to analyze single networks, but they often tightly relate to the analysis of populations of networks: once a method is proposed to embed the nodes of a network in a low-dimensional space, statistical modeling can be done on these embeddings to analyze populations of networks.

As we saw in Section 2.1.1, a first representation of the nodes is given by the leading eigenvectors of the adjacency matrix. In practice, the graph Laplacian matrix is often preferred to extract low rank node embedding, in which case the result is called the spectral embedding [Richiardi et al., 2013]. This approach was recently extended to handle populations of networks, by averaging the eigendecompositions of the Laplacian matrices [Chen et al., 2020].

Ma et al. [2017] propose a method to find node embeddings combining several "views" of the same network, e.g., structural and functional connectivity for brain networks. Another approach adapts the `word2vec` formalism to network nodes. It relies on deep learning to find low-dimensional representations of the nodes accounting for their role in random walks on the network. In particular, it was applied on functional and structural brain networks by Rosenthal et al. [2018], who showed that the resulting embedding of the brain regions produces interpretable representations of the connectome. Similarly, using a deep learning approach, Xu et al. [2019] obtained latent brain node embeddings with a Gaussian uncertainty. This uncertainty was then used to quantify the effect of an intervention on amnestic mild cognitive impairment patients, by comparing the node embeddings before and after the intervention.

From a different perspective, Krioukov et al. [2010] found a natural geometric method to represent networks with a core-periphery structure. Nodes in a network can be embedded in a hyperbolic space, e.g., the Poincaré disk, which resembles the Euclidean plane at the origin, and grows faster as the points get closer to the disk's edge. The authors showed that placing the core nodes at the center of the disk and the periphery nodes close to the boundary provides a natural embedding of the network structure in dimension two. This method was successfully applied to the structural and functional connectivity of the brain, providing alternative interpretations of its architecture [Cacciola et al., 2017, Tadić et al., 2019, Zheng et al., 2019a, Gao et al., 2020]. The recent works of Chami et al. [2019], Mathieu et al. [2019] show that deep learning tools can also be leveraged for hyperbolic network analysis, and this field should keep developing in the years to come.

A large proportion of the current research on node embedding relies on graph neural networks and auto-encoders. This subject is addressed in a dedicated paragraph below.

Graph Auto-Encoders. Shortly after the Variational Auto-Encoder (VAE) paradigm was proposed by Kingma and Welling [2014], applications to network analysis followed. Generally speaking, auto-encoders learn a low-dimensional representation z of very high-dimensional structured data x like images. The maps from z to x and back are parameterized with neural networks and optimized with stochastic gradient descent. The strength of VAEs comes from the neural network architecture, which can be tailored to the data, e.g., convolutions for images. Variational Graph Auto-Encoders (VGAEs, Kipf and Welling [2016]) extend the VAE idea to networks: given an $n \times n$ symmetric adjacency matrix A and a list F of features for each node (e.g., the shape and surface of each brain region), the VGAE produces n low-dimensional latent vectors $z_1, \dots, z_n$, assuming for instance that $A_{ij} \simeq \sigma(z_i^\top z_j)$, with σ a sigmoid function. The map from (A, F) to z is parameterized by a graph neural network [Henaff et al., 2015, Wu et al., 2020]. Graph neural networks can be considered an equivalent of convolutional neural networks for graphs, in the sense that they leverage the network structure to aggregate information from one layer to the other. As a great advantage, graph neural networks can use the features of each node along with the adjacency matrix, which helps build a better data representation.

Many variants of VGAEs coexist (essentially depending on the graph neural network used in the architecture), but the general idea often fundamentally relies on the low rank hypothesis of the adjacency matrix. VGAEs are particularly well suited to analyze a single very large network

(e.g., academic citation networks), where the effective rank is very low compared to the number of nodes. They are also widely used for populations of networks, especially brain networks. They allow incorporating multimodal connectivity information, as in Liu et al. [2019]. The obtained embeddings can then be used for predictive purposes, e.g., to distinguish patients with Alzheimer's disease from patients with late mild cognitive impairment [Banka and Rekik, 2019]

The VGAE framework (and more generally deep learning for graphs – also sometimes called geometric deep learning) is popular in the current research literature for its flexibility, its representation power and its performance on prediction tasks. However, from a statistical modeling perspective, it leaves aside the generative procedure: as an example, the basic VGAE assumes that the latent node embeddings z_i are drawn from a standard Gaussian distribution $\mathcal{N}(0, I)$, which is simplistic in the case of brain networks. They are thus better suited for regression than population modeling.

Dictionary models. A complementary perspective was proposed in a series of papers by D'Souza et al. [2018, 2019a,b], D'Souza et al. [2021b]. It proposes to model each brain connectivity matrix in a population as a weighted sum of rank one patterns: $A = \lambda_1 x_1 x_1^\top + \cdots + \lambda_n x_n x_n^\top$. The patterns x_i are defined across the population, and the weights λ_i are defined separately for each individual. This formulation thus represents the connectivity by a dictionary of patterns with simple structure. The x_i can be learned orthogonal to each other, in order to maximize the model's expressiveness – in the sense that the dictionary components $x_i x_i^\top$ are orthogonal to each other if the x_i's are. The weights λ_i can then be used as regression variables, e.g., to predict clinical scores for patients diagnosed with autism spectrum disorder. The advantage of this model is its interpretability: each pattern $x_i x_i^\top$ can be interpreted as a heatmap on the brain, giving the heat tone x_{ik} to the brain region k. It also directly models the eigendecomposition of the adjacency matrices, by imposing that the eigenvectors are fixed across the population. In this sense, dictionary models are dual to graph auto-encoders, which learn latent node embeddings (playing the role of eigenvectors) for each matrix and consider the weights of the inner product (playing the role of eigenvalues) as fixed across the population.

Low rank population models. Finally, a line of research proposes tractable statistical models to simultaneously capture both aspects of the population variability (eigenvalues and eigenvectors). These models design probability distributions on the eigenvectors and the eigenvalues, and consider them as latent variables. This approach was first proposed by Hoff [2009a], who showed that data sets of covariance matrices can be modeled with their eigendecomposition. Later, Durante et al. [2017] proposed a low-rank non-parametric mixture model for binary networks. As a notable difference with Hoff [2009a], this work proposes to model the eigendecomposition of the deviations of the observations from the cluster mean, rather than the decomposition of the adjacency matrices themselves. Recently, Duan et al. [2020] proposed an eigendecomposition mixture model to handle spiked graph Laplacian matrices; they showed that this approach allows performing community detection. The same year, Signorelli and Wit [2020] proposed an alternative low-rank model to perform clustering on network populations. More generally, the authors of the papers mentioned here proved that simple hierarchical models are well suited to handle populations of networks. Their structure allows easily including cofactors, class labels and regression variables [Hoff and Ward, 2004, Hoff, 2007b, Aliverti and Durante, 2019].

In Chapters 4 and 5 of this thesis, we propose two contributions in line with the low rank population modeling approach. In particular, Chapter 5 leverages the flexibility of inference algorithms in exponential models and proposes a general low rank model for populations of networks, which can handle weighted network matrices as well as binary matrices or positive matrices within the same framework, with only minor algorithmic differences. Low rank models simultaneously provide a strong representation power using a restricted number of parameters and very good interpretability. Both of these properties are crucial to the objective of integrating statistical methods in research and medical applications.

Remark. In this thesis, we chose to focus on the population modeling problem, rather than on network regression and classification tasks. Numerous methods have been proposed to achieve these goals, in particular using graph neural networks. They have found great success for brain

network analysis, and are still improving every year (the interested reader may refer to the yearly conference MICCAI, and especially the GRAIL workshop – exceptionally called GLMI in 2019 – on graphs and machine learning in medical applications). Regression and classification could be considered as extensions of the contributions of this thesis. In particular, the hierarchical structure of the model proposed in Chapter 5 naturally lends itself to the inclusion of cofactors. This point is further discussed in the conclusion of the thesis in Chapter 8. The population modeling problem differs from regression in general, in the sense that it aims at understanding the variability in a population.

2.2 From population models to disease progression analysis

In the last contribution of this thesis, we change our perspective and study a more mature population model for longitudinal data. Although this setting differs from the central context of brain network analysis, the contributions in this thesis on network data sets could be used as a basis for longitudinal models for brain networks. To this extent, building a solid understanding on the capacities of longitudinal models is a crucial step toward the design of longitudinal brain network models.

Longitudinal data analysis. Longitudinal studies consist in the repeated observations of a given population of subjects across time. They differ from cross-sectional studies, which only observe each individual once. Cross-sectional studies are more frequent, as they require no long-term monitoring and resources. However, longitudinal studies are often considered more robust: following the time evolution of the same subjects allows isolating the impact of external cofactors. In practice, longitudinal data analysis serves two main purposes. On the one hand, it allows monitoring physiological evolutions with "normal" aging, i.e., aging unaffected by specific diseases. On the other hand, it is used to analyze and monitor disease progression. We are especially interested in the latter, which finds important applications for neurodegenerative diseases like Parkinson's disease or Alzheimer's disease [Koval, 2020, Couronné, 2021].

Mixed-effects models. From a statistical modeling perspective, longitudinal data models are a category of population models, where each individual may have several measurements. Formally, each individual i has n_i observations $y_{i1}, \dots, y_{in_i}$ at times $t_{i1}, \dots, t_{in_i}$. These observations can be a set of clinical scores, as in this thesis, but more complicated types of data like images and shapes can be considered. The observations may have a portion of missing data, i.e., each observation is not necessarily complete. Classical population models assume that the observations are independent, which is not the case within the measurements of a single individual. In other words, two consecutive observations of the same individual could not be considered independent even if the known characteristics of the individual (age, sex, ...) were considered as part of the observations: the individual is not entirely characterized by its features, and an inherent unknown latent variability remains to differentiate one individual from another. In terms of modeling, this statement is formalized by introducing, for each individual i, a latent variable z_i which participates in the data generating mechanism. Such models are called mixed-effect models: they combine both fixed effects (where the same parameters are used for each individual) and random effects (where the parameters – the latent variable z_i – vary for each individual). In their simplest form, mixed-effects models assume a linear dependency between the observations and the latent and observed features. This approach is widely used in the literature, as it allows for explicit computations and simple interpretations [Lavielle, 2014]. Over the last two decades, non-linear mixed-effects models have gained increased popularity, with new algorithms allowing for efficient parameter estimation [Kuhn and Lavielle, 2005]. We refer the reader to [Chevallier, 2019, Ch. 2, Sec. 1] for a more detailed introduction on linear and non-linear mixed-effects models for longitudinal data.

Segmented progression models. In this thesis, we will be considering the general framework proposed by Schiratti [2017] for working with longitudinal data. This framework defines a non-linear mixed-effects model that characterizes the trajectory of each individual as a random modification of an average population trajectory. It brings together a large flexibility in model

specification, simple interpretability and efficient estimation algorithms. In particular, it was used to analyze the progression of Alzheimer's disease and Parkinson's disease from clinical scores and anatomical shapes. More recently, an extension was proposed in Chevallier et al. [2021] to account for the impact of a treatment in the trajectory of subjects affected by a disease. This extension proposes to consider a trajectory segmented into two or more pieces, with each piece corresponding to a progression stage of the disease, or to the impact of the treatment. It was successfully applied to chemotherapy monitoring and, more recently, it was included in a more complex model on mixtures of trajectories with different numbers of pieces [Debavelaere et al., 2020].

With the increasing complexity of disease progression models (latent factors, missing data, trajectories in several pieces, mixture models, etc.), reliability becomes a crucial issue. In particular, selecting the best model among a set of competitors becomes an increasingly complex task, which gets intractable as the problem dimension grows large. Some papers have considered the problem of selecting models in a longitudinal data context [Azari et al., 2006, Jones, 2011], working in the setting of linear mixed-effects models. Considering the specific case of selecting the number of breaks in the trajectory, we will be attempting to quantify the extent to which classical model selection methods can be adapted to discriminate between several non-linear mixed-effects models in the framework of Schiratti [2017].

2.3 Outline of this thesis

The remainder of this thesis is organized in six chapters. Chapter 3 introduces the necessary tools and notions from statistics, optimization and geometry. Chapter 4, 5 and 6 propose modeling approaches for populations of networks and theoretical consistency results. Chapter 7 studies a specific longitudinal population model for the progression analysis of Parkinson's disease. Finally, Chapter 8 concludes this thesis and presents several perspectives opened by the various topics explored throughout the other chapters.

- **Chapter 3** *Tools for Optimization and Inference in Population Models*

  This chapter gathers five distinct sections, introducing the key algorithmic and theoretical notions used in the next chapters. We first introduce some concepts on non-smooth convex optimization, used in Chapter 4. Next, we introduce tools on inference in hierarchical models. We introduce the Stiefel manifold and its interest for statistical modeling, and in particular the Cayley transform. We propose two new methods for the Cayley transform. These notions are used in Chapter 5, where they allow performing MCMC with Metropolis-Hastings on the Stiefel manifold. Finally, we introduce two classical model selection techniques used in Chapter 7.

- **Chapter 4** *Sparse Low Rank Decomposition for Graph Data Sets*

  This chapter proposes a first decomposition model for populations of networks. It relies on the empirical low rank and sparsity of real-world network matrices and proposes a new variational objective. This formulation decomposes the variability of the population into a fixed sparse and low-rank template, and individual-dependent sparse low-rank deviations. We use classical proximal methods to optimize the variational objective. We show that this formulation is suited to describe real world populations of networks like transportation networks and functional brain networks.

- **Chapter 5** *A Spectral Model for Populations of Networks*

  This chapter can be considered a hierarchical modeling counterpart of the previous chapter. We go further in the understanding of the population variability, by introducing a random effect model to account for individual-level deviations from the population average. This model relies on a low-rank truncation of the network matrices eigendecomposition. We provide an algorithm to estimate the model parameters and show its performance on simulated data. This algorithm is applied to a data set of functional brain networks from the UK BioBank. We show that our model provides a compact, interpretable and precise representation of the population variability.

  This chapter was published in the journal Entropy [Mantoux et al., 2021].

- **Chapter 6** *Asymptotic Analysis of the Spectral Model*

  In this chapter, we provide asymptotic guarantees for the statistical model studied in the previous chapter. We first show that the model is identifiable. Then, we prove the strong consistency of the Maximum A Posteriori estimator of the model parameters. Finally, we prove that this estimator is normally asymptotic for a restricted, simplified version of the model.

  This chapter was published in the journal ESAIM-PS [Mantoux et al., 2022].

- **Chapter 7** *Estimation and Model Selection for Segmented Longitudinal Trajectories*

  This chapter studies a longitudinal model for disease progression analysis. It takes stock on previous work modeling the trajectory of subjects affected by Parkinson's disease and Alzheimer's disease; in this chapter, the trajectory of each individual is modeled as a piecewise affine trajectory. We show that the population average trajectory can be robustly estimated for trajectories with more than one piece, and that the number of pieces can be selected robustly even with a very strong noise and a large portion of missing data. We apply our methodology to the cohort of the Parkinson's Progression Markers Initiative. We show how our model can be used to describe the disease evolution, and how it could be extended to better account for the impact of treatments.

Chapter 3

Tools for Optimization and Inference in Population Models

This chapter introduces tools on proximal algorithms, Markov Chain Monte Carlo methods, inference on the Stiefel manifold and model selection. In particular, Section 3.4 introduces new methods for sampling and averaging on Stiefel manifold. The next chapters will be relying on these tools and only briefly recall their definitions.

Contents

3.1 Non-smooth convex optimization for regularization

In this section, we briefly review the notions from convex optimization used in Chapter 4. We will be interested in a variational approach to model populations of networks relying on a non-smooth convex optimization problem. This formulation allows retrieving real-world network properties like sparsity and low rank. For detailed introductions to non-smooth convex optimization and proximal methods, we refer the reader to Nesterov [2018] and Parikh and Boyd [2014]. For a general introduction to convex optimization, see Boyd and Vandenberghe [2009]. Note that, in this section, we only consider functions defined on the entire space $\mathbb{R}^n$, taking finite values.

3.1.1 Non-smooth convex optimization

Subdifferentials. When optimizing a non-smooth convex function $f : \mathbb{R}^n \to \mathbb{R}$, the gradient of f is not defined everywhere, so that classical gradient-based methods cannot be used. In this setting, the notion of gradient is generalized to the notion of subdifferential. Taking stock on the fact that, for a differentiable convex function g, we have for all $x, y \in \mathbb{R}^n$:

$$g(y) \geq g(x) + \langle \nabla g(x), y - x \rangle \,,$$

the subdifferential of f at a point x is defined as the set $\partial f(x)$ of vectors v that verify

$$\forall y \in \mathbb{R}^n, f(y) \geq f(x) + \langle v, y - x \rangle \,.$$

In particular, for a differentiable convex function, the subdifferential corresponds exactly to the singleton containing the gradient. The elements of $\partial f(x)$ are called subgradients of f at x. If they are known, subgradients can be used as a surrogate for gradients to minimize f. In particular, x^* is a global minimizer of f if and only if x^* verifies the optimality condition $0 \in \partial f(x^*)$. This condition replaces the classical condition $\nabla g(x^*) = 0$ in the differentiable convex case.

Proximal splitting methods. Although several algorithms are available to deal with general non-smooth convex optimization problems, in this thesis we are interested more specifically in objective functions with a simple structure – more precisely sums of norms on vector and matrix spaces. In this specific case, proximal methods provide a simple and powerful framework to derive optimization algorithms. For a constant $\lambda > 0$, given a – possibly non-smooth – convex function f, the proximal operator of f at x is defined by:

$$\mathrm{prox}_{\lambda f}(x) = \mathrm{argmin}_y \, \lambda f(y) + \frac{1}{2} \|x - y\|^2 \,.$$

The proximal operator can be understood as a gradient descent step minimizing a function that has the same minima as f – see [Parikh and Boyd, 2014, Sec. 3.3]. In particular, the fixed points of the proximal operator correspond to the global minimizers of f:

$$x^* = \mathrm{prox}_{\lambda f}(x^*) \iff 0 \in \partial f(x^*) \,.$$

This equation gives a fixed-point scheme to obtain a global minimizer: if the proximal operator of f is available, f can be minimized simply by iteratively applying its proximal operator.

The same idea can be used to minimize more complex functions. Generally, we have the relation $\mathrm{prox}_{\lambda f}(x) = (I + \lambda \partial f)^{-1}(x)$ (Eq. 3.4 in Parikh and Boyd [2014]). This formula is to be understood in the sense that $y = \mathrm{prox}_{\lambda f}(x)$ is the unique element such that $x \in y + \lambda \partial f(y)$. It can be used to derive other optimization algorithms, like the proximal gradient method.

Douglas-Rachford algorithm. In Chapter 4, we will be interested in minimizing a function f which writes as the sum $g + h$ of two non-smooth convex functions whose proximal operators are known. In this specific case, another fixed-point equation can be derived, relying on the reflexive proximal operator, defined by:

$$\mathrm{Rprox}_{\lambda h}(x) = 2\,\mathrm{prox}_{\lambda h}(x) - x \,.$$

It can be proved that minimizers x^* of $g + h$ satisfy the fixed-point equation:

$$\begin{cases} z^* = \mathrm{Rprox}_{\lambda h} \circ \mathrm{Rprox}_{\lambda g}(z^*) \\ x^* = \mathrm{prox}_{\lambda h}(z^*) \,. \end{cases}$$

This equation gives rise to the Douglas-Rachford fixed-point algorithm [Boyd et al., 2011]. For a step size $\alpha \in]0, 2[$, it writes as:

$$\begin{cases} z_{t+1} = \left(1 - \frac{\alpha}{2}\right) z_t + \frac{\alpha}{2} \mathrm{Rprox}_{\lambda h} \circ \mathrm{Rprox}_{\lambda g}(z_t) \\ x_{t+1} = \mathrm{prox}_{\lambda h}(z_t) \,, \end{cases}$$

and converges to global minimizers of $g + h$.

In a similar vein, we will also minimize the sum of a differentiable function and functions with known proximal operator. This case is addressed by the Generalized Forward-Backward Splitting method, which combines ideas from the Douglas-Rachford algorithm and the forward-backward algorithm. We refer the reader to the paper proposing the algorithm, which clearly exposes its structure [Raguet et al., 2013].

3.1.2 Sparse and low-rank regularizations

In the context of this thesis, the main interest of proximal methods is their application to regularization in optimization problems. It is well known that penalizing an optimization problem by the ℓ_1-norm of the variable induces sparsity in the optimal value [Bunea et al., 2007]. In other words, if instead of minimizing the convex function $x \mapsto f(x)$, one minimizes $x \mapsto f(x) + \lambda \|x\|_1$ with $\lambda > 0$, the solution will provide a trade-off between the optimality of f and the sparsity of x.

The sparsity induced by the ℓ_1-norm naturally arises from the associated soft thresholding proximal operator: we have that

$$\text{prox}_{\lambda\|\cdot\|_1}(x) = \text{sgn}(x) \odot \max(\text{abs}(x) - \lambda, 0)\,,$$

with sgn the sign and abs the absolute value taken for each coefficient. In particular, the proximal operator of the ℓ_1-norm truncates the coordinates of x below λ. Sparsity inducing penalties are widely used in compressed sensing to recover structured information from noisy data [Crespo Marques et al., 2019]. They provide a theoretically sound framework with recovery guarantees on the support of the optimization variables.

In Chapter 4, we will be using sparsity inducing penalties to account for the fact that real world network matrices often have a very sparse structure. Additionally, we will be using the low rank of these matrices to define a network decomposition model.

Consider an optimization problem where the variable of interest X is a square matrix. Like sparsity, the rank can be regularized using an optimization penalty, given by the nuclear norm:

$$\|X\|_* = \sum_{i=1}^{n} \sigma_i(X)\,,$$

with $\sigma_i(X)$ the i-th – possibly null – singular value of X. While the sparsity-inducing penalty sums the absolute values of the coefficients, the low-rank inducing penalty sums the singular values of the optimization variable. The same regularization mechanism then induces sparsity among the singular values of the matrix, and thus reduces the matrix rank of the optimization result.

Low rank regularization is commonly used for matrix completion and network analysis [Hu et al., 2018]. Like the ℓ_1 penalization, it provides a simple and robust tool to design rank-constraining algorithms with a solid underlying theoretical understanding. The interactions between sparse and low rank penalties also produce interesting combinations: as an example, in the context of network analysis, both penalties can be combined to separate an input matrix X into a sum of a low rank matrix and a sparse matrix [Kanada et al., 2018]. In Chapter 4, we will also be considering combined penalties relying on both the ℓ_1-norm and the nuclear norm.

3.2 Inference in hierarchical models

In this section, we introduce the key tools and concepts used in Chapters 5 and 7, where we will be working on latent variable models for brain network analysis and longitudinal analysis of disease progression. For a much more complete introduction on inference in complex models and the concepts exposed in this section, we refer the reader to Murphy [2012].

3.2.1 Bayesian Inference and Markov Chain Monte-Carlo

In the framework of Bayesian statistics, complex situations can be accounted for by using a hierarchical structure in the dependency between random variables. This dependency expresses the fact

the observed data y is given as a function of hidden latent variables z, whose distribution is determined by the model parameters θ. Formally, the model consists in a prior distribution $p(\theta)$ on the parameters, a distribution of the latent variables given the parameters $p(z \mid \theta)$, and a distribution of the observed data given the latent variables and the parameters $p(y \mid z, \theta)$.

Inference methods. When given a set of observations y, two options are available to estimate the model parameters. The first option is standard Bayesian inference: it consists in determining the posterior distribution of the parameters $p(\theta \mid y) = p(y, \theta)/p(y)$. The second option is to estimate the Maximum A Posteriori, i.e., the maximum value of the posterior distribution: $\hat{\theta} \in \operatorname{argmax}_\theta p(\theta \mid y, z)$. This second option is cheaper, as it only requires obtaining a single value of θ rather than an entire distribution. However, it is also less precise, since it gives no information on the concentration of the distribution around θ, i.e., on the uncertainty of the maximum value. It also prevents from considering cases where the distribution $p(\theta \mid y)$ has several modes. We will discuss the problem of computing the Maximum A Posteriori in the next section on the EM algorithm, and focus here on the first option, i.e., obtaining information on the distribution $p(\theta \mid y)$.

Posterior distribution inference. In most complex models (and in particular in the models considered in this thesis), the posterior distributions of interest do not correspond to usual distributions, and are often known only up to the normalizing constant $p(y)$. Obtaining information on these distributions is a difficult problem in general. Two wide classes of methods have been developed to find approximate solutions. The first option, called Variational Inference, consists in finding a distribution among a simple family (e.g., the set of Gaussian distributions) that is closest to the posterior distribution, in the sense of the Kullback-Leibler divergence. Variational inference often produces fast algorithms, but may fail to accurately describe the posterior distribution. The second option, called Markov Chain Monte-Carlo (MCMC), consists in designing a simple Markov Chain such that the distribution of its samples asymptotically converges to the posterior distribution. For instance, if the distribution of interest is $p(z \mid y, \theta)$, an MCMC method will provide a sequence $z_1, z_2, \dots$ such that, as $t \to +\infty$, the sample z_t can be considered a true sample of the target distribution $p(z \mid y, \theta)$. In this thesis, we will be mostly relying on MCMC methods, which in general are preferable to Variational Inference when they are computationally tractable, in the sense that they are asymptotically exact.

Metropolis-Hastings algorithm. Many MCMC methods have been developed over the years (Gibbs sampling, MALA, HMC, NUTS, to name only a few). In this thesis, we will be relying on the Metropolis-Hastings (MH) algorithm, arguably one of the most simple MCMCs, popular for its flexibility and its versatility. Although maybe not the most performant in general in very high dimensional settings, its combination with the Gibbs sampling allows for a fast convergence without computing the gradient of the density. We briefly present the algorithm for a general target distribution π. Given a sample x_t, the MH algorithm builds a new sample x_{t+1} based on any transition kernel $q(x_t, \cdot)$ (for instance, a Gaussian distribution centered at x_t). The algorithm first draws a sample y_t from the distribution $q(x_t, \cdot)$. Then, it defines an acceptance probability:

$$\alpha(x, y) = \min\left(1, \frac{\pi(y)q(y, x)}{\pi(x)q(x, y)}\right).$$

The new sample x_{t+1} is chosen equal to y_t with probability $\alpha(x_t, y_t)$, or equal to x_t with probability $1 - \alpha(x_t, y_t)$. The expression of α ensures that the Markov chain thus obtained is reversible with respect to π. Along with additional assumptions on π and q, this property allows proving the convergence of the Markov Chain to the distribution π. Intuitively, the acceptance step "corrects" the base transition kernel q, by encouraging transitions that increase $\pi(x)$.

In most cases, the transition kernel is a symmetric random walk: for instance, it may be defined by a Gaussian density: $q(x, \cdot) = \mathcal{N}(x, \Sigma)$. Choosing a symmetric transition kernel simplifies the expression of the acceptance probability into $\alpha(x, y) = \pi(y)/\pi(x)$, which speeds up computations in practice.

Adaptive MCMC. The choice of the matrix Σ in the Gaussian random walk has an important impact on the performance of the algorithm: on the one hand, if Σ is too big, the transitions rarely fall into regions of high probability, and most transitions are thus rejected. On the other hand, if Σ is too small, the ratio α gets close to one and most transitions are accepted: the MCMC produces highly correlated samples and the convergence is very slow. In practice, the jump amplitude is often adjusted along the chain, in order to reach a given target rate of accepted transitions: as long as the acceptance rate is too high, Σ is increased, and as long as it is too low, Σ is decreased. In dimension greater than one, the shape of the covariance matrix – i.e., the direction and relative magnitude of its eigenvectors – also has an important impact: performing jumps with optimal size but in the wrong direction also results in high rejection rates. In cases where the identity matrix produces poor results, the covariance Σ can be updated along the MCMC to match the empirical covariance matrix of the samples. Both of these adaptations – acceptance rate and covariance matrix – are crucial to obtain a fast convergence and a fast decreasing correlation between consecutive samples.

Metropolis within Gibbs. Finally, a last modification is required to obtain a satisfactory convergence of the symmetric random walk Metropolis-Hastings algorithm for high-dimensional distributions. Given the algorithm description we provided above, if a transition from x to y is such that $\pi(y) > \pi(x)$, it will always be accepted ($\alpha(x, y) = 1$). However, in high dimension, assuming the jump size is small enough, it is sufficient that only a subset of the coordinates of y contribute to a significant increase of π for the jump to be accepted. As an example, if x has 1000 components, and if variations of the 999 first components have a strong impact on $\pi(x)$, then the value of the last coordinate x_{1000} will rarely play a role in the acceptance or the rejection of the transitions. In other words, when a transition occurs, it will impact the value of the last coordinate x_{1000} in a direction often independent of the marginal distribution $\pi(x_{1000})$. As a consequence, the Metropolis-Hastings algorithm often performs poorly on certain coordinates, in the sense that the marginal distributions converge very slowly to the target distribution; this problem, in turn, may prevent the remaining coordinates to find the regions of high probability.

A widely used solution consists in applying the Metropolis-Hastings procedure within a Gibbs sampler. The Gibbs sampling is another very simple MCMC method, which iteratively samples each coordinate of the random variable given the other coordinates. For instance, if the random variable of interest has three components $X = (x, y, z)$, then the Gibbs sampling algorithm iteratively samples $x_{t+1} \sim \pi(x \mid y = y_t, z = z_t)$, then $y_{t+1} \sim \pi(y \mid x = x_{t+1}, z = z_t)$, and finally $z_{t+1} \sim \pi(z \mid x = x_{t+1}, y = y_{t+1})$. The resulting Markov chain converges in distribution to the distribution $\pi(x, y, z)$ under certain regularity hypotheses on π. The strong advantage of Gibbs sampling is the decoupling of sampling procedures for each coordinate. It addresses the issue of the Metropolis-Hastings algorithm in high dimension, in the sense that each component is treated equally. However, unlike Metropolis-Hastings, it requires that the conditional distributions $\pi(x \mid y, z)$, $\pi(y \mid x, z)$, ... can be sampled from explicitly, which is rarely the case when working with non-linear hierarchical models.

The Metropolis within Gibbs algorithm overcomes this problem: for each coordinate, instead of sampling from the conditional density, it performs a Metropolis-Hastings step targeting the conditional density. The sampling procedure is described in Algorithm 3.2.1. As with Gibbs sampling and Metropolis-Hastings, the Metropolis within Gibbs algorithm can be shown to converge to the target distribution as the length of the chain grows large. It allows sampling from very high dimensional densities without the problem faced by Metropolis-Hastings which we described above. It may be slower than Metropolis-Hastings, in the sense that each step may take more time to perform; however much fewer steps are required to reach convergence. Finally, the Metropolis within Gibbs may also benefit from an adaptive MCMC structure: in practice, each coordinate has its own transition variance, and each variance is tuned to reach the same acceptance rate on every coordinate.

3.2.2 The EM algorithm and its variants

MLE and MAP. In practice, the posterior distribution $p(\theta \mid y)$ of complex non-linear hierarchical models is hard to infer. Sampling from the distribution $p(\theta \mid y)$ is a difficult task, for which MCMC methods may struggle to converge. In these cases, it is often simpler to compute

Algorithm 3.2.1: The Metropolis within Gibbs algorithm

input : Target distribution $\pi(x_1, \dots, x_d)$
Initialize $x_1^{(0)}, \dots, x_d^{(0)}$
for $t = 1$ **to** T **do**
  for $k = 1$ **to** d **do**
    Define the Metropolis kernel q_k targetting $\pi(\cdot \mid x_1^t, \dots, x_{k-1}^t, x_{k+1}^{t-1}, \dots, x_d^{t-1})$
    Sample $x_k^t \sim q_k(\cdot \mid x_k^{t-1})$
  end
end
return $(x^{(t)})_{1 \leq t \leq T}$

the Maximum A Posteriori (MAP) of the distribution, i.e., a point where the maximum value of $p(\theta \mid y) \propto p(y, \theta)$ is reached. The MAP is the Bayesian equivalent of the Maximum Likelihood Estimator (MLE), and it only differs by the presence of the prior term $p(\theta)$, which acts as a regularizer on the parameter space. For instance, a Gaussian prior on θ acts as an ℓ_2 penalty, an exponential prior acts as an ℓ_1 penalty, etc.

EM algorithm. In the case of models with latent variables, the MAP and MLE estimations are not straight-forward, since the model likelihood writes as an integral over the space of latent variables:

$$p(y \mid \theta) = \int p(y \mid z, \theta) p(z \mid \theta) \mathrm{d}z \,.$$

This integral is costly to compute in general: it sums over probabilities that may vary by several orders of magnitude, and naive Monte-Carlo estimation produces very high variance results. For these reasons, classical optimization methods like gradient descent cannot be used. The EM algorithm, introduced by Dempster et al. [1977], overcomes this hurdle by proposing an iterative optimization scheme that does not require evaluating the model's likelihood. For a detailed introduction to the EM algorithm and many of its most classical variants, we refer the reader to McLachlan and Krishnan [2008]. The main insight leading to the EM algorithm is the definition of an auxiliary function Q that relies on a previous guess $\tilde{\theta}$ for the value of θ:

$$Q(\theta \mid \tilde{\theta}) = \mathbb{E}_{p(z \mid y, \tilde{\theta})}[\log p(y, z \mid \theta)] \,.$$

Note that this Q-function is defined for the MLE estimation. The Q-function for the MAP would instead have its expectation on the term $\log p(y, z, \theta)$. The expectation of Q comes in a much more tractable form than $p(y \mid \theta)$: it sums over logarithms of probabilities, which often have simple expressions. The key remark motivating its definition is that increasing Q leads to increasing $p(y \mid \theta)$: it can be shown easily that, for all $\theta, \tilde{\theta}$,

$$\log p(y \mid \theta) - \log p(y \mid \tilde{\theta}) \geq Q(\theta \mid \tilde{\theta}) - Q(\tilde{\theta} \mid \tilde{\theta}) \,.$$

The EM algorithm uses this remark to build a sequence of parameters $(\theta_t)_{t \geq 0}$ such that $p(y \mid \theta_t)$ is non-decreasing, by defining:

$$\theta_{t+1} \in \operatorname{argmax}_\theta Q(\theta \mid \theta_t) \,.$$

Under suitable regularity conditions, it can be shown that the EM algorithm (as well as all the variants we introduce in this chapter) converges to a critical point of the objective function. In practice, the algorithm proceeds in two steps: first, the expectation in the function $Q(\cdot \mid \theta_t)$ is computed (e.g., if it is a polynomial, its coefficients are computed). Next, it is maximized to find θ_{t+1}. These two steps (Expectation, Maximization) give their name to the algorithm. The EM algorithm is a powerful and flexible tool to deal with simple latent variable models. It provides a general procedure that handles many types of data, from mixture models to time series analysis.

SAEM algorithm. The computation of the function Q is often intractable for complex hierarchical models – for instance, when the posterior distribution $p(z \mid y, \theta)$ corresponds to no classical

distribution. The Expectation step must then be approximated with Monte-Carlo. For instance, a first idea could be to maximize the approximation

$$\hat{Q}(\theta \mid \theta_t) = \frac{1}{n_t} \sum_{i=1}^{n_t} \log p(y, z_i) \,,$$

with $z_i \sim p(z \mid y, \theta_t)$. If $n_t \to +\infty$ as $t \to \infty$, then the Monte-Carlo EM algorithm induced by this approximation can be shown to converge. However, this convergence comes at a prohibitive computational cost. The Stochastic Approximation EM overcomes this hurdle by sampling only one value z_t of $p(z \mid y, \theta)$ at each time step [Delyon et al., 1999]. It defines a surrogate $Q_t(\theta)$ to the function $Q(\theta \mid \theta_t)$, by aggregating the functions $\theta \mapsto \log p(y, z_t \mid \theta)$ of the previous time steps. More precisely, the function Q_{t+1} at step $t+1$ writes as a weighted average between Q_t and $\theta \mapsto \log p(y, z_{t+1} \mid \theta)$:

$$Q_{t+1}(\theta) = (1 - \gamma_t) Q_t(\theta) + \gamma_t \log p(y, z_{t+1} \mid \theta) \,,$$

and θ_{t+1} is defined as a maximizer of Q_{t+1}. The weights $(\gamma_t)_{t \geq 0}$ are positive, often decreasing, and must satisfy the conditions $\sum_t \gamma_t = +\infty$ and $\sum_t \gamma_t^2 < +\infty$. The idea of the SAEM algorithm is that, when t grows large and θ_t starts varying slower, Q_t becomes a good approximation of $Q(\theta \mid \theta_t)$, so that the maximization of Q_t behaves like the EM algorithm, and guarantees convergence.

EM and SAEM for exponential models. In general, computing Q_t means storing each function $\log p(y, z_t \mid \theta)$, which rapidly becomes intractable as the iterations count grows. However, in the specific case of statistical models belonging to the (curved) exponential family, the storage problem becomes much simpler. A *curved exponential model* is such that $\log p(y, z \mid \theta)$ can be written under the form

$$\log p(y, z \mid \theta) = \langle S(y, z), \varphi(\theta) \rangle + \psi(\theta) \,.$$

If $\varphi(\theta) = \theta$, the model is simply called an *exponential model*. In other words, the dependency between the observations and the parameters is linear; the useful information on y and z is stored in the so-called sufficient statistics $S(y, z)$. Curved exponential models form a very large family, including for instance Gaussian graphical models, most linear hierarchical models, as well as the model proposed in Chapter 5. In the special case of network analysis, the interest for the Exponential Random Graph Model is largely motivated by its exponential form. The specific structure of exponential models allows for efficient inference, in particular with the EM algorithm. Their expression of Q simplifies to

$$Q(\theta \mid \theta_t) = \langle \mathbb{E}_{p(z \mid y, \theta_t)}[S(y, z)], \varphi(\theta) \rangle + \psi(\theta) \,.$$

In practice, this means that the problem of computing the expectation of a function is brought down to computing its coefficients $\mathbb{E}_{p(z \mid y, \theta_t)}[S(y, z)]$. The SAEM algorithm also benefits from this interesting property: for each t, the function Q_t may be written under the form

$$Q_t(\theta) = \langle S_t, \varphi(\theta) \rangle + \psi(\theta) \,.$$

At $t = 0$, S_0 is given by $S(y, z_0)$. Then, recursively, S_{t+1} is obtained by

$$\begin{aligned}
Q_{t+1}(\theta) &= (1 - \gamma_t) Q_t(\theta) + \gamma_t \log p(y, z_{t+1} \mid \theta) \\
&= (1 - \gamma_t)(\langle S_t, \varphi(\theta) \rangle + \psi(\theta)) + \gamma_t (\langle S(y, z_{t+1}), \varphi(\theta) \rangle + \psi(\theta)) \\
&= \langle (1 - \gamma_t) S_t + \gamma_t S(y, z_{t+1}), \varphi(\theta) \rangle + \psi(\theta) \\
&= \langle S_{t+1}, \varphi(\theta) \rangle + \psi(\theta) \,,
\end{aligned}$$

So that we have $S_{t+1} = (1 - \gamma_t) S_t + \gamma_t S(y, z_{t+1})$. From a practical perspective, this means that the sequence of functions $\log p(y, z_t \mid \theta)$ does not need to be stored, as all the necessary information is stored in S_t, which is computed iteratively. This formulation makes the SAEM algorithm a very powerful tool for inference in exponential models.

MCMC-SAEM algorithm. Finally, in the case of complex non-linear hierarchical models, the density of $p(z \mid y, \theta)$ often does not correspond to any classical family of distributions, so that sampling z_t from $p(z \mid y, \theta_t)$ cannot be achieved directly. A first idea to overcome this issue would be to run a MCMC chain $z_t^1, z_t^2, \dots, z_t^T$ at each time step t, and use z_t^T as an approximate sample of $p(z \mid y, \theta_t)$. This procedure would be very costly, requiring a Markov Chain to converge at every step of the SAEM. A simpler, much faster alternative was proposed by Kuhn and Lavielle [2004], and later studied by Allassonnière et al. [2010]. It consists in performing one single MCMC step per SAEM step, using z_t as a base point to obtain z_{t+1}. The resulting sequence $(z_t)_{t\geq 0}$ does not form a homogeneous Markov Chain: at each time step t, the transition kernel evolves and targets the distribution $p(z \mid y, \theta_t)$. But as t grows large and the value of θ stabilizes, so does the transition kernel. Asymptotically, z_t becomes a true sample of the distribution $p(z \mid y, \theta_t)$, so that the resulting procedure – called the MCMC-SAEM algorithm – behaves like the SAEM. The convergence of the MCMC-SAEM is a threefold process: the distribution of z_t, the stochastic approximation S_t and the estimation θ_t evolve simultaneously. The improvement of each of the three helps refine the two others, until the convergence regime is reached.

Remark. The convergence rate of the EM algorithm is asymptotically linear in general, with model-specific variations. For instance, in the case of Gaussian mixture models, the convergence speed depends on the overlap between the mixture components of the true model [Ma et al., 2000]. Depending on the choice of γ_t, the SAEM algorithm converges at an optimal rate of $t^{-1/2}$.

3.2.3 Inference in non-exponential families

In practice, complex models like non-linear mixed-effects models may not belong to the (curved) exponential family. Similarly, much more complex models relying on neural networks feature a highly non-linear dependency between the variables and the model parameters. This is the case of the model we work with in Chapter 7. Many models used in network analysis, like Graph Auto-Encoders, do not belong to the curved exponential family. Non-exponential models cannot be estimated easily using the EM algorithm, and alternatives have been developed to perform approximate inference.

Making the model exponential. After studying the MCMC-SAEM algorithm, Kuhn and Lavielle [2005] proposed a trick to approximate any non-exponential model by an exponential model. Given a model $p(y, z, \theta)$, the trick consists in considering θ as a latent variable, introducing a new parameter $\bar{\theta}$ such that $\theta \sim \mathcal{N}(\bar{\theta}, \sigma_\theta^2 I)$. The new model $p(y, z, \theta, \bar{\theta})$ is exponential, and the optimal value of $\bar{\theta}$ is close to the optimal value of θ for the non-exponential model [Debavelaere and Allassonnière, 2021]. Exponentializing the model comes at a computational cost: it often slows the convergence, as the MCMC on the latent variable θ must converge along with the rest of the latent variables. However, it benefits from a great simplicity and allows using the MCMC-SAEM with no major hurdle.

Fisher identity, towards variational methods Another option to estimate the parameters of a non-exponential model relies on the so-called Fisher identity, which we will be using in Chapter 6. It states that

$$\nabla_\theta \log p(y \mid \theta) = \mathbb{E}_{p(z|y,\theta)}[\nabla_\theta \log p(y, z \mid \theta)] \,.$$

In other words, the (intractable) gradient of the MLE objective function $\log p(y \mid \theta)$ can be seen as an expectation of the (tractable) gradient of $\log p(y, z \mid \theta)$. This gradient can often be computed easily, either by hand or using automatic differentiation. The Fisher identity directly gives a stochastic gradient algorithm to find a point θ such that $\nabla_\theta \log p(y \mid \theta) = 0$. Drawing at each step a point z_{t+1} from the distribution $p(z \mid y, \theta_t)$, the next value of θ is obtained as $\theta_{t+1} = \theta_t + \gamma_t \nabla_\theta \log p(y, z_{t+1} \mid \theta_t)$. If the step sizes γ_t are suitably chosen, this stochastic approximation converges to a critical point of the model likelihood. For cases where the posterior distribution $p(z \mid y, \theta)$ cannot be directly sampled from, an MCMC can be used as in the MCMC-SAEM, performing a single step of MCMC at each stochastic gradient iteration.

The stochastic gradient descent provided by the Fisher identity is a very general method that can be used for arbitrarily complex models. Yet, in practice it is rarely used under this form,

because the gradient $\nabla_\theta \log p(y, z \mid \theta)$ often has a very high variance, and poor initializations may prevent the algorithm from converging [Hoffman, 2017, Sec. 3]. The recent work of Fang and Li [2021] shows that variance reduction techniques can be used to make the algorithm usable in practice.

Variational Bayes. As we mentioned in Section 3.2.1, complex distributions are classically handled either with MCMC or with Variational Inference – which approximates the complex distribution by a close, simple distribution. The MCMC-SAEM and the Fisher stochastic gradient descent rely on the first approach. Variational Bayes (VB) is a more recent alternative proposed by Kingma and Welling [2014], which relies on the second approach to estimate the MLE (or the MAP). It proposes to approximate the posterior distribution $p(z \mid y, \theta)$ by a simple variational approximation $q(z \mid y, \varphi)$, e.g., a Gaussian distribution. It defines a new objective to be maximized:

$$Q(\theta, \varphi) = \log p(y \mid \theta) - \mathrm{KL}[\, q(z \mid y, \varphi) \;||\; p(z \mid y, \theta)\,] \,.$$

This objective simultaneously tries to maximize the likelihood of the observed data, while enforcing that the variational approximation $q(z \mid y, \varphi)$ is close to its target $p(z \mid y, \theta)$: maximizing Q thus gives an estimate for both θ and φ.

The main interest of this new objective function is that, unlike maximizing $\log p(y \mid \theta)$ alone, Q can be written in a tractable form. Indeed, it can be shown easily that we have

$$Q(\theta, \varphi) = \mathbb{E}_{q(z|y,\theta)} \left[\log p(y, z \mid \theta) - \log q(z \mid y, \theta)\right] \,,$$

which suggests a stochastic gradient algorithm by sampling z_t from $q(z \mid y, \varphi)$. In practice, the so-called reparameterization trick allows differentiating w.r.t. the dependency of z_t on φ. We refer the reader to Kingma and Welling [2014] for more details. The Variational Bayes framework is at the core of Variational Auto-Encoders, which play a central role in large networks analysis Kipf and Welling [2017].

Summary. When designing a new model, the choice between SAEM and VB should be done cautiously, considering the advantages and drawbacks of both frameworks. The SAEM is preferable for exponential models, as it provides an exact and straightforward estimation procedure, which often converges relatively fast. However, its extension to non-exponential models comes at the cost of model exponentialization, which slows the convergence in practice. On the other hand, the VB method provide a fast, approximate inference procedure. It easily scales to very high-dimensional data and models by allowing for *amortization*: for n independent data samples $(z_1, y_1), \ldots, (z_n, y_n)$, instead of learning each distribution $q(z_i \mid y_i, \varphi)$ separately, the model imposes a common structure to each posterior distribution, e.g., $q(z_i \mid y_i, \varphi) = \mathcal{N}(\mu_\varphi(y_i), \sigma^2 I)$. In practice, VB is preferred for large scale applications where approximate inference is acceptable, whereas the SAEM is recommended for models in a curved exponential family, and models with small amounts of parameters. Finally, note that both VB and SAEM are inherently prone to classical inference pitfalls like mode identification in multimodal posterior distributions. In each case, adaptations must often be made to obtain satisfactory results in practice.

3.3 Stiefel manifolds and directional statistics

In this section, we introduce the main tools to model and manipulate data and tangent vectors taking values in the Stiefel manifold. These concepts will be used in Chapter 5 to model the distribution of the eigenvectors of brain functional connectivity matrices.

3.3.1 Brief reminders on Riemannian geometry

The model we consider in Chapter 5 relies on variables which satisfy a non-linear constraint. This constraint defines a curved subspace of the ambient Euclidean space. Riemannian geometry is the appropriate tool to work with this subspace: it consists in the study of smooth manifolds from a geometric perspective, allowing to measure distances and angles on manifolds.

Little knowledge on smooth manifolds is assumed in this thesis, apart from the basic definitions of a manifold and a tangent space. In this section, rather than dwelling on rigorous developments, we will focus on recalling the main notions important to our purpose (in particular, we only consider sub-manifolds of $\mathbb{R}^n$). For detailed introductions to the topic, we refer the interested reader to the excellent books by Lee [2003, 2018].

Riemannian manifold. A d-dimensional *Riemannian manifold* $\mathcal{M} \subset \mathbb{R}^n$ is a smooth manifold endowed with a smooth map which, at every point $x \in \mathcal{M}$, gives an inner product $\langle \cdot, \cdot \rangle_x$ on the tangent space $T_x\mathcal{M}$. In practice, this inner product defines the norm $\|u\|_x = \sqrt{\langle u, u \rangle_x}$ of a tangent vector u. It defines a local, "straight line" distance on the manifold: informally, if $x \in \mathcal{M}$ and $u \in T_x\mathcal{M}$ is very close to zero, the distance between x and $x + u$ is given by $\|u\|_x$. The map $x \mapsto \langle \cdot, \cdot \rangle_x$ is called a *Riemannian metric* on $\mathcal{M}$.

Since $\mathcal{M}$ is embedded in $\mathbb{R}^n$, the tangent spaces $T_x\mathcal{M}$ are naturally embedded in $\mathbb{R}^n$, and a natural inner product on $T_x\mathcal{M}$ can be defined as the restriction of the Euclidean inner product of the ambient space $\mathbb{R}^n$. This method produces a reference Riemannian metric for any smooth submanifold, called the Euclidean metric.

As an example, when measuring the distance between two places close to each other on Earth, we often proceed by assuming that the ground in-between is flat – which, in a geometrical sense, amounts to assuming that each place lives close to the tangent plane of the other place. However, the "straight line" distance on Earth only works locally: for larger distances, e.g., between countries, the Earth curvature must be taken into account to obtain accurate results.

Geodesics. The next point of interest is the definition of a "straight line" in the manifold $\mathcal{M}$. A straight line can be defined, in a Euclidean space, as the shortest curve from one point to another. This definition can be extended to Riemannian manifolds: once the norm of tangent vectors is set, it is possible to define the length of a curve $\gamma : [0, 1] \to \mathcal{M}$. This definition generalizes the Euclidean case, defining the length $L(\gamma)$ as

$$L(\gamma) = \int_0^1 \|\gamma'(t)\|_{\gamma(t)} \, \mathrm{d}t \, .$$

For instance, if the Riemannian norm $\|\gamma'(t)\|_{\gamma(t)}$ is constant equal to v for all t, we have $L(\gamma) = v$. With this definition, a *geodesic* – equivalent of a straight line in a manifold – is defined as a smooth curve γ such that no other curve with the same ends has a shorter length. As an example, the geodesics of the sphere are given by circles with maximal diameter, e.g., the meridians or the equator for planets.

Geodesics bring a notion of distance on the manifold: as in the Euclidean case, the distance between two points is defined as the length of the shortest path (i.e., the geodesic curve) between the two points. The distance between two points x and y is thus defined as the length of the shortest geodesic curve from x to y.

Exponential map. As with straight lines in the Euclidean case, it is sufficient to give a starting point $x \in \mathcal{M}$ and an initial speed vector $v \in T_x\mathcal{M}$ to define a unique geodesic curve $\gamma(t)$, such that $\gamma(0) = x$ and $\gamma'(0) = v$. In the case of Stiefel manifolds, that we will define in the next section, it can be proved that the curve $\gamma(t)$ is well-defined for all $t \in \mathbb{R}$.

The application mapping v to $\gamma(1)$ is called the *exponential map* at x, denoted $\mathrm{Exp}_x(v)$. It can be seen as a way to project the tangent vectors onto the manifold – in the sense that, if v is small, we have $\mathrm{Exp}_x(v) = x + v + o(v)$. As an example, let us consider the unit circle at $x = (1, 0)$. The tangent space at x is given by the vertical vectors, which can be written as $v = (0, \theta)$. It can be proved that, for any $\theta \in \mathbb{R}$, the exponential map at x sends the vertical tangent vector $v = (0, \theta)$ to the point $\mathrm{Exp}_x(v) = (\cos(\theta), \sin(\theta))$. Note that the map $v \mapsto \mathrm{Exp}_x(v)$ is not injective, as two tangent vectors can lead to the same points.

Logarithm map. If two points x and y are sufficiently close on $\mathcal{M}$, it can be proved that there exists only one smallest tangent vector v such that $y = \mathrm{Exp}_x(v)$. The vector v is called the *logarithm* of y at x, denoted $\mathrm{Log}_x(y)$. The logarithm map thus inverts the exponential map: given

a point on the manifold sufficiently close to x, it produces the tangent vector $v \in T_x\mathcal{M}$ such that the geodesic starting at x in direction v lands on y at $t = 1$. However, the logarithm is only defined locally: for instance, on a sphere, there are several ways to go from one point to its antipode.

Riemannian gradient. Finally, an important application of Riemannian geometry is optimization on manifolds. Given a smooth function $f : \mathcal{M} \to \mathbb{R}$, it is possible to compute an equivalent version of the gradient of f on $\mathcal{M}$, which we denote $\nabla_{\mathcal{M}} f$. In the common classical case where f is defined over the entire ambient space $\mathbb{R}^n$, and the manifold $\mathcal{M}$ is endowed with the Euclidean metric, the Riemannian gradient $\nabla_{\mathcal{M}} f(x)$ at a point x simply corresponds to the orthogonal projection of the Euclidean gradient $\nabla f(x)$ onto the tangent space $T_x\mathcal{M}$. The definition of the gradient allows for algorithms like gradient descent or stochastic gradient descent, which make it a central tool to manipulate manifold-valued data.

3.3.2 The Stiefel manifold

In Chapters 5 and 6, we will be focusing more specifically on the Stiefel manifold. It is defined as the set of matrices $X \in \mathbb{R}^{n\times p}$ such that $X^\top X = I_p$. In other words, an element X of the Stiefel manifold is given by a set of p orthogonal vectors $x_1, \ldots, x_p \in \mathbb{R}^n$ with unit norm, which form the columns of X. In this thesis, such matrices will be used to represent the eigenvectors of low-rank network adjacency matrices. The Stiefel manifold, denoted as $\mathcal{V}_{np}$, has been thoroughly studied for many application purposes, for instance in rigid body motion analysis [Oualkacha and Rivest, 2012], classification [Ali and Gao, 2018], graphical models [Adhikary et al., 2019], community detection [Duan et al., 2020] or financial time series analysis [Meng, 2021]. It also plays an important role in applications involving Grassmann manifolds. In practice, Stiefel manifold-valued variables often arise by dropping the last $n - p$ columns of an orthogonal matrix. It allows taking advantage of setups where $p \ll n$: the complexity of basic data manipulation thus scales as $O(np)$ instead of $O(n^2)$.

In particular, many explicit formulas are available for the expression of the tangent spaces and the exponential map, which we briefly recall here. Most of these formulas, as well as many others, are given and explained in great detail in Edelman et al. [1998].

Tangent space. For a matrix $X \in \mathcal{V}_{np}$, it can be shown that any tangent vector $D \in T_X\mathcal{V}_{np}$ can be written under the form

$$D = XA + X_\perp B\,, \tag{3.1}$$

with $A \in \mathbb{R}^{p\times p}$ a skew-symmetric matrix (i.e., $A^\top = -A$), $B \in \mathbb{R}^{n-p\times p}$ and $X_\perp$ a list of $n - p$ vectors completing X into an orthonormal basis of $\mathbb{R}^n$. In particular, for $n = p$, the tangent space at an orthogonal matrix $X \in \mathbb{R}^{n\times n}$ is given by the set $\{XA \mid A^\top = -A\}$. Given X and D, A can be obtained as $X^\top D$, and $X_\perp B$ can be computed as $(I_n - XX^\top)D$. Note that B is not unique, as it depends on the choice of $X_\perp$.

For instance, consider $I_{3,2}$, the 2×2 identity matrix padded with zeros on the third row. A basis of the tangent space at $I_{3,2}$ is given by

$$D_1 = \begin{pmatrix} 0 & 1 \\ -1 & 0 \\ 0 & 0 \end{pmatrix}, \quad D_2 = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{pmatrix}, \quad \text{and} \quad D_3 = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

Figure 3.3.2 represents the vectors D_1, D_2 and D_3 in the three-dimensional space. In this specific case, A is given by the first two rows of the matrices and B by the third row.

Riemannian structure of $\mathcal{V}_{np}$ Defined as a submanifold of $\mathbb{R}^{n\times p}$, the Stiefel manifold naturally inherits the Euclidean metric of the ambient space. A second metric can also be used, which comes from an alternative definition of the Stiefel manifold: $\mathcal{V}_{np}$ can also be seen as the quotient of the orthogonal group $O_n(\mathbb{R})$ by the orthogonal group $O_{n-p}(\mathbb{R})$. This amounts to saying that a matrix $X \in \mathcal{V}_{np}$ can be seen as an orthogonal matrix whose last $n-p$ columns have been quotiented out. A Riemannian metric can be defined from this quotient structure; it is called the *canonical metric* on the Stiefel manifold. Both Euclidean and canonical metrics have a simple expression:

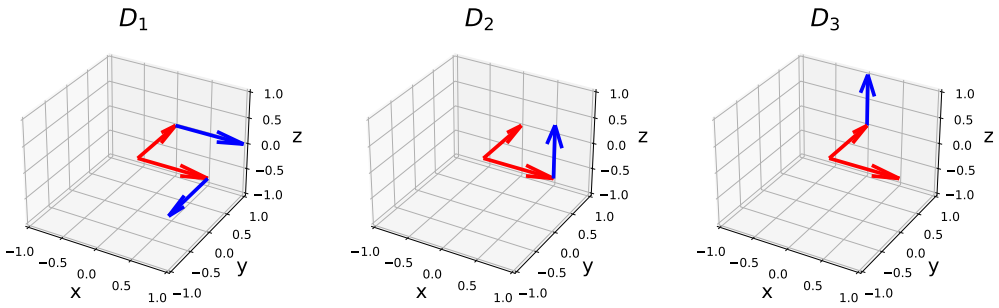

Figure 3.1: Basis vectors for the tangent space of $\mathcal{V}_{3,2}$ at $X = I_{3,2}$. The two columns of $I_{3,2}$ are represented in red, and each D_k is represented in blue.

given the decompositions (A, B) and (A', B') of two tangent vectors D and D' as in Equation (3.1), the two metrics respectively write as

$$\begin{cases} \langle D, D' \rangle_{\text{Euclidean},X} = \langle A, A' \rangle_F + \langle B, B' \rangle_F \\ \langle D, D' \rangle_{\text{Canonical},X} = \frac{1}{2} \langle A, A' \rangle_F + \langle B, B' \rangle_F \,, \end{cases}$$

denoting by $\langle \cdot, \cdot \rangle_F$ the canonical Frobenius inner product of matrices $\langle U, V \rangle_F = \text{Tr}(U^\top V)$. Note that this expression does not depend on the choice of B. The canonical metric gives half its weight to the skew-symmetric component, which allows counting each degree of freedom only once. For this reason, in this thesis we chose to use the canonical metric on the Stiefel manifold.

Exponential map. Given a starting point X and an initial speed vector $D = XA + X_\perp B$ (with $A^\top = -A$), the geodesic curve $t \mapsto \text{Exp}_X(tD)$ for the canonical metric can be computed explicitly using a matrix exponential (Corollary 2.2 in Edelman et al. [1998]). Let (Q, R) be the compact QR decomposition of $X_\perp B$: $QR = X_\perp B$, with $Q \in \mathcal{V}_{np}$ and R a $p \times p$ upper triangular matrix. Defining the two curves:

$$\begin{pmatrix} M(t) \\ N(t) \end{pmatrix} = \exp \left[t \begin{pmatrix} A & -R^\top \\ R & 0 \end{pmatrix} \right] \begin{pmatrix} I_p \\ 0 \end{pmatrix} ,$$

we have that $\text{Exp}_X(tD) = XM(t) + QN(t)$. The exponential map on the Stiefel manifold can thus be computed very simply, by computing the matrix exponential of a $2p \times 2p$ matrix.

Logarithm map. No explicit formula is available for the logarithm map on Stiefel manifolds. Zimmermann [2017] recently proposed an iterative algorithm to approximate the logarithm. Although this algorithm converges in very few iterations when its input is close to the base point, it requires computing a matrix exponential and a matrix logarithm at every step, which makes it difficult to include in large scale algorithms. Furthermore, the convergence becomes more and more unstable as the dimension grows, and only works for points very close to the base point in high dimension. As a last drawback, the definition domain of the logarithm map on $\mathcal{V}_{np}$ has not been fully characterized to this day, with only a lower bound on its size found by [Rentmeesters, 2013, p. 95].

Function gradients. Given a function $f : \mathbb{R}^{n \times n} \to \mathbb{R}$, the Riemannian gradient of f for the canonical metric has a simple expression:

$$\nabla_{\mathcal{V}} f(X) = \nabla f(X) - X \nabla f(X)^\top X \,.$$

It is thus very simple to compute in practice once the Euclidean gradient of f is known. This will be the case in the practical application considered in Chapter 5.

3.3.3 Statistical modeling on the Stiefel manifold

Chapters 5 and 6 consider random variables taking values in $\mathcal{V}_{np}$. This problem is at the core of the field of directional statistics, where the objects of interest are orientations in space (given by unit vectors), or collections of orientations (orthonormal tuples of vectors belonging to a Stiefel manifold). The seminal work of Khatri and Mardia [1977] proposed to adapt the well-studied von Mises distribution on the sphere to the Stiefel manifold. Other distributions were proposed, in particular to account for antipodal distributions giving equal weight to a vector and its opposite [Jupp and Mardia, 1979]. Recent work has focused on tractable statistical inference, with sampling algorithms [Hoff, 2009b, Jauch et al., 2020a], Maximum Likelihood Estimation [Kume et al., 2013] and Bayesian inference [Pal et al., 2020, Meng, 2021]. For a detailed introduction to statistics on Stiefel (and Grassmann) manifolds, we refer the reader to Chikuse [2003c], as well as other papers from the same author [Chikuse, 1990, 2003b, 2006].

Invariant measure. Defining continuous statistical models on the Stiefel manifold requires a base measure to be used as support for parametric probability densities. The invariant measure $[\mathrm{d}X]$ over the Stiefel manifold is well suited for this purpose. It is defined (up to a constant factor) as the only measure over $\mathcal{V}_{np}$ invariant under orthogonal rotations. It can also be shown that the invariant measure corresponds to the Hausdorff measure over $\mathcal{V}_{np}$, which can be considered a generalization of the Lebesgue measure over $\mathcal{V}_{np}$ [Jauch et al., 2020b].

Parametric distributions. Using the invariant measure, parametric models on $\mathcal{V}_{np}$ can be defined with probability distributions under the form $p_\theta(X)[\mathrm{d}X]$, with θ the model parameter. The most widely known example is the von Mises-Fisher distribution, that we will use in Chapter 5, and which defines $p_\theta(X) \propto \exp\langle X, \theta\rangle_F$. Other distributions like the Fisher-Bingham distribution and its generalization rely on the exponential of quadratic expressions. Most of the models proposed in the literature share two properties: on the one hand their likelihood has a simple interpretable density; on the other hand, their normalizing constant is often intractable, and custom methods must be used to perform inference.

Remark. The default idea of a probabilistic model on general Riemannian manifolds is to generalize the Gaussian distribution [Pennec, 2006]. It consists in choosing a mean point $x \in \mathcal{M}$, drawing a random variable ε following a Gaussian distribution in $T_x\mathcal{M}$, and computing $\mathrm{Exp}_x(\varepsilon)$. Although this method works well in general, the logarithm map is needed to retrieve the mean and the covariance of ε. Gaussian distributions are thus not well suited to Stiefel manifolds, where simpler and better tailored alternatives are available.

3.4 New applications of the Cayley transform

This section introduces the Cayley transform on the Stiefel manifold and presents two new contributions. We provide a simple algorithm to compute the inverse of the Cayley transform, and show that the Cayley transform can be used to perform fast Metropolis-Hastings sampling on the Stiefel manifold.

3.4.1 The Cayley transform on Stiefel manifolds

As we saw in the previous section, although the exponential and logarithm maps can be computed with good precision on Stiefel manifolds, their utility is limited in practice by their computational cost or – for the logarithm – the stability domain of the approximation. The Cayley transform alleviates the first difficulty and removes the second. Given $X \in \mathcal{V}_{np}$, it provides an alternative map from $T_X\mathcal{V}_{np}$ to $\mathcal{V}_{np}$ that is fast to compute and approximates the exponential map at the first order.

Informal construction. We give a step-by-step construction of the Cayley transform which informally justifies its formula. We start by defining an initial version C_X^0 for skew-symmetric

matrices. For W an $n \times n$ skew-symmetric matrix, this version is given by:

$$C_X^0(W) = \left(I + \frac{1}{2}W\right)\left(I - \frac{1}{2}W\right)^{-1} X\,.$$

Since the matrix $\left(I + \frac{1}{2}W\right)\left(I - \frac{1}{2}W\right)^{-1}$ is orthogonal, $C_X^0(W)$ also belongs to the Stiefel manifold.

Remark. The matrices of the form $I + W$, with W a skew-symmetric matrix, are always non-singular. This can be seen by applying the spectral theorem to the matrix iW, which is a Hermitian matrix: it gives that W can be diagonalized with imaginary eigenvalues. As a consequence, $I + W$ can be diagonalized with eigenvalues having real part equal to one; hence the matrix is non-singular.

It can be shown easily that, as W goes to zero,

$$C_X^0(W) = X + WX + o(\|W\|_F)\,.$$

This relation helps design a function that is a first-order approximation of the exponential map, which verifies $\mathrm{Exp}_X(D) = X + D + o(\|D\|_F)$. The goal is now to build a matrix W, such that $WX = D$.

Remembering the decomposition $D = XA + X_\perp B$ introduced in the last section ($A^\top = -A$), it can be noticed that taking $W = DX^\top - XD^\top$ gives $WX = 2XA + X_\perp B = D^\sharp$. The map $D \mapsto D^\sharp$ is very simple: mapping D to $D^\sharp$ amounts to multiplying A by two, and conversely going from $D^\sharp$ to D amounts to dividing A by two. As a consequence, if we define $D^* = \frac{1}{2}XA + X_\perp B$, we obtain that taking $W = D^*X^\top - X(D^*)^\top$ leads to $WX = D$.

Finally, note that $D^* = (I + XX^\top)D$ and that $D^\sharp = (I - \frac{1}{2}XX^\top)D$. With this last remark, we are ready to give the main definition of the Cayley transform.

Cayley transform. For $X \in \mathcal{V}_{np}$, let $W_X(D^*) = D^*X^\top - X(D^*)^\top$. The Cayley transform at X is defined by:

$$C_X(D) = C_X^0(W_X(D^*))\,.$$

By construction, the Cayley transform satisfies the property $C_X(D) = X + D + o(\|D\|)$. In practice, it only requires matrix additions and multiplications. The inversion of the $n \times n$ matrix $I - \frac{1}{2}W$ can be avoided by using the Sherman-Morrison-Woodbury formula, which allows decreasing the cost to that of inverting a $2p \times 2p$ matrix instead of an $n \times n$ matrix [Wen and Yin, 2013]. This property makes the Cayley transform a very appealing candidate to replace the exponential map when performing gradient descent, and it is widely used in practice [Li et al., 2020b].

Intuitively, the Cayley transform behaves like the exponential map when D is close to zero. However, unlike the exponential map, when $t \to +\infty$ the point $C_X(tD)$ converges to a limit point, which can be computed explicitly using the spectral theorem. Figure 3.4.1 compares the curves obtained for the exponential map and the Cayley transform, in the specific case $X = I_{3,2}$ and the vectors D_1, D_3, D_3 of Figure 3.3.2.

Remark. The map $D \mapsto D^*$ is nothing but the natural isomorphism from $T_X\mathcal{V}_{np}$ to its dual space $T_X^*\mathcal{V}_{np}$ (i.e., the space of linear forms from $T_X\mathcal{V}_{np}$ to $\mathbb{R}$). The map $D \mapsto D^\sharp$ is the reverse mapping.

3.4.2 A fixed-point algorithm for the inverse Cayley transform

To the best of our knowledge, no simple algorithm exists to compute the inverse of the Cayley transform in a general setting. It has been shown that the Cayley transform is injective [Macías-Virgós et al., 2018] and spans the entire Stiefel manifold apart from a set with measure zero [Jauch et al., 2020b]. A closed-form inversion method is available for the special case where p is even [Kaneko et al., 2013]. Computing the inverse of the Cayley transform is a problem of practical interest. For instance, it enables to compute averages on the Stiefel manifold, as proposed by Kaneko et al. [2013] (care should be taken when reading this paper: the notations n and p are reversed compared to ours). It could enable to work with Gaussian distributions, as defined in the previous section, replacing the exponential map with the Cayley transform.

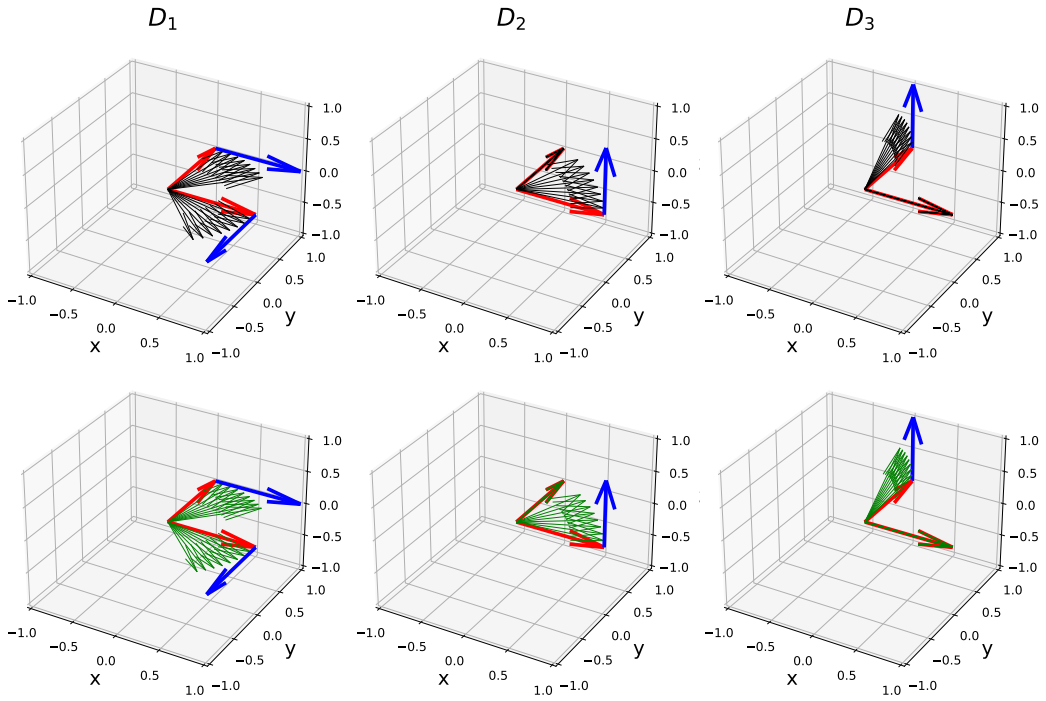

Figure 3.2: For the three vectors D_1, D_2 and D_3 defined in the previous section, we show the curves $\mathrm{Exp}_{I_{32}}(tD_k)$ and $\mathrm{C}_{I_{32}}(tD_k)$ for $t \leq 0.8$. The first row shows the exponential curves (in black), and the bottom row shows the Cayley curves (in green).

We provide a simple algorithm to compute the inverse of the Cayley transform for any value of p. The unique solution D^* of the equation $Y = C_X^0(W_X(D^*))$ – assuming that it exists – satisfies the relation:

$$\begin{aligned}
Y &= \left[I_n - \frac{1}{2}(D^*X^\top - X(D^*)^\top)\right]^{-1}\left[I_n + \frac{1}{2}(D^*X^\top - X(D^*)^\top)\right]X \\
\iff &\left[I_n - \frac{1}{2}(D^*X^\top - X(D^*)^\top)\right]Y = \left[I_n + \frac{1}{2}(D^*X^\top - X(D^*)^\top)\right]X \\
\iff &2(Y-X) - (D^*X^\top - X(D^*)^\top)Y = D^* - X(D^*)^\top X \\
\iff &2(Y-X) - D^*X^\top Y + X(D^*)^\top(Y+X) = D^* \,.
\end{aligned}$$

This finally leads to the fixed point equation:

$$D^* = \left(2(Y-X) + X(D^*)^\top(Y+X)\right)(I_p + X^\top Y)^{-1}\,. \tag{3.2}$$

From this equation, we propose the following fixed-point algorithm on D^*:

$$D^*_{t+1} = (1-\gamma_t)D^*_t + \gamma_t \Pi_X\left[\left(2(Y-X) + X(D^*_t)^\top(Y+X)\right)(I_p + X^\top Y)^{-1}\right] \tag{3.3}$$

Two additions are made to the base fixed point equation (3.2). First, we project the result of Equation (3.2) onto the tangent space at X. This operation, denoted Π_X in Equation (3.3), ensures that $D^*_t \in T^*_X\mathcal{V}_{np}$ in practice (note that Π_X has a simple explicit expression, see Edelman et al. [1998]). Second, we introduce an averaging scheme with a sequence of positive decreasing weights γ_t in order to ensure convergence (we use $\gamma_t = 1/(1+t)$). Once D^* is computed, the resulting value of D is obtained with $D = (I + XX^\top)D^*$.

In practice, the convergence is extremely fast, even for large values of n and p (e.g., $n = 1000$ and $p = 300$). For relatively small values of D, in four steps or fewer, the difference between two consecutive iterates is of the order of machine precision, and the obtained value of D yields a Cayley transform equal to Y up to machine precision. The convergence is slower for very large

values of D, but after three steps the error remains of the order of 10^{-6} in Euclidean norm, which is more than sufficient in many cases.

Remark. To the best of our knowledge, this method has not been used before to compute the inverse Cayley transform. Fiori et al. [2012] however suggested that the iterative method of Huang et al. [2008] could be used for a similar purpose.

3.4.3 Metropolis-Hastings sampling with Cayley proposals

Motivation

As a second application of the Cayley transform, we turn to the problem of sampling from a general probability distribution $\pi(X)$ on the Stiefel manifold. This problem has been addressed by Ouyang [2008], who proposes a Metropolis-Hastings algorithm with a symmetric proposal based on Gaussian distributions and orthogonal projections. More recently, two papers from Jauch et al. [2020a,b] propose alternative methods. The first method builds a distribution on square matrices such that the polar decomposition of the random variable produces samples from the distribution of interest on the Stiefel manifold. The second method proposes to lift the Stiefel manifold in the tangent space at the identity I_{np} via the inverse Cayley transform, and perform sampling in the (flat) tangent space, where traditional MCMC methods can be used.

The second method, although it can be applied to any distribution, could perform poorly in cases where the distribution of interest is centered around (or close to) a point unreachable by the Cayley transform, for instance $-I_{np}$. In this thesis, like Ouyang [2008], we use the Metropolis-Hastings algorithm directly on the Stiefel manifold, except that the proposals are generated with the Cayley transform. This results in a very simple procedure. Starting from a point X_t, we generate X_{t+1} by drawing a centered isotropic Gaussian variable D_t in $T_X\mathcal{V}_{np}$, and considering the proposal update $Y_t = C_{X_t}(D_t)$. The proposal is accepted or refused according to the probability

$$\alpha(X_t, Y_t) = \frac{\pi(Y_t)q(Y_t, X_t)}{\pi(X_t)q(X_t, Y_t)}\,,$$

with $q(X, Y)$ the Markov kernel defined by the Cayley transition. The main difficulty lies in the expression of $q(X, Y)$. It can be derived with computations similar to those carried out in Jauch et al. [2020b], but in practice this computation is not necessary. Indeed, it turns out that the proposal density is very close to being symmetric, i.e., that $q(X_t, Y_t) = q(Y_t, X_t)$. We provide partial theoretical evidence for this property in the next subsection, partly relying on numerical experiments. Although this we could not prove this equality formally, it holds in practice with great accuracy for the typical variances of D_t used in practice. It might be, however, that the approximation fails for large values of D, as the Stiefel manifold is not a symmetric space.

Metropolis-Hastings-type MCMCs are of particular interest in our application of Stiefel manifold MCMC in Chapter 5: they allow for a fast exploration of a distribution centered around a mode. In our application, this configuration is of particular interest, as we will be considering the posterior distribution of the eigenvectors of a matrix given its noisy coefficients, with a prior imposing a choice on the order and sign of the eigenvectors.

Note that distributions with a high degree of symmetry (e.g., w.r.t. column sign or column order) can also be explored using Metropolis-Hastings. Jumping from one mode to another can be achieved by modifying the transition kernel to choose between a Cayley jump and a change of sign or column order. However, for specific cases like Fisher-Bingham distributions, more efficient, tailored algorithms are available [Hoff, 2009b].

Proposal symmetry

The density of the random variable Y_t can be computed explicitly with a change of variable, as in the Euclidean case. The change of variable theorem in the Euclidean setting is generalized to smooth manifolds by Traynor [1994], as pointed out by Jauch et al. [2020b] (Theorem 4.1), to which we refer the reader for more details. We apply this extension to the change of variable from D_t to Y_t. Let us denote $p_{X_t}(D_t)$ the density of the random variable D_t in the vector space $T_{X_t}\mathcal{V}_{np}$. We have that:

$$q(X_t, Y_t) = p_{X_t}(D_t) \times J(X_t, D_t)\,, \tag{3.4}$$

with $J(X, D)$ a generalized Jacobian determinant of C_X at D, which accounts for the fact that the Jacobian matrix $\mathrm{d}C_X(D)$ is rectangular. More specifically $J(X, D)$ is defined as:

$$J(X, D) = |\mathrm{d}C_X(D)^* \mathrm{d}C_X(D)|^{1/2} , \tag{3.5}$$

where the star denotes the adjoint linear operator. In order to show that $q(X_t, Y_t) = q(Y_t, X_t)$, we will prove that the two terms in the right hand side of Equation (3.4) take the same values from X_t to Y_t and conversely.

In order to study $q(Y_t, X_t)$, we need to introduce $E_t = C_{Y_t}^{-1}(X_t)$. The following proposition shows that D_t and E_t are related to each other by an explicit relationship.

Proposition 1. *Let $X \in \mathcal{V}_{np}$, $D \in T_X\mathcal{V}_{np}$, $Y = C_X(D)$ and $E = C_Y^{-1}(X)$. Then we have the relations $W_X(D^*)X = D$, $W_Y(E^*)Y = E$ and $W_X(D^*) = -W_Y(E^*)$.*

Proof. The first point ($W_X(D^*)X = D$ and $W_Y(E^*)Y = E$) was established in the previous sections and will not be discussed further. For the sake of brevity, in this proof we denote $W_X = W_X(D^*)$.

Since D^* (and thus D) can be retrieved from W_X, the mapping $D \mapsto W_X$ is injective. Furthermore, the definition of the Cayley transform gives the equation

$$\left(I - \frac{1}{2}W_X\right) Y = \left(I + \frac{1}{2}W_X\right) X .$$

Hence, if we can find a tangent vector $\tilde{E}$ such that $W_Y(\tilde{E}^*) = \tilde{E}^*Y^\top - Y(\tilde{E}^*)^\top = -W_X$, then $C_Y(\tilde{E}) = X$, and thus necessarily $E = \tilde{E}$ and $W_X = -W_Y(\tilde{E}^*) = -W_Y(E^*)$, by injectivity of the Cayley transform. Knowing the expected result, we consider $\tilde{E} = -W_X Y$: we need to show that $\tilde{E}^*Y^\top - Y(\tilde{E}^*)^\top = -W_X$. We have, by definition of $\tilde{E}^*$:

$$\begin{aligned}
\tilde{E}^*Y^\top - Y(\tilde{E}^*)^\top &= \left(I - \frac{1}{2}YY^\top\right)\tilde{E}Y^\top - Y\tilde{E}^\top\left(I - \frac{1}{2}YY^\top\right) \\
&= -\left(I - \frac{1}{2}YY^\top\right)W_X YY^\top - YY^\top W_X\left(I - \frac{1}{2}YY^\top\right) .
\end{aligned}$$

We reduce this equation by using the decomposition $I = YY^\top + Y_\perp Y_\perp^\top$ (which is the sum of the orthogonal projections onto the column spaces of Y and $Y_\perp$). We obtain:

$$\tilde{E}^*Y^\top - Y(\tilde{E}^*)^\top = -W_X - Y_\perp Y_\perp^\top W_X Y_\perp Y_\perp^\top .$$

We thus need to prove that $Y_\perp Y_\perp^\top W_X Y_\perp Y_\perp^\top = 0$. First, we have that

$$\begin{aligned}
Y_\perp Y_\perp^\top &= I - YY^\top \\
&= I - \left(I - \frac{1}{2}W_X\right)^{-1}\left(I + \frac{1}{2}W_X\right)XX^\top\left(I - \frac{1}{2}W_X\right)\left(I + \frac{1}{2}W_X\right)^{-1} \\
&= I - \left(I - \frac{1}{2}W_X\right)^{-1}\left(I + \frac{1}{2}W_X\right)(I - X_\perp X_\perp^\top)\left(I - \frac{1}{2}W_X\right)\left(I + \frac{1}{2}W_X\right)^{-1} \\
&= \left(I - \frac{1}{2}W_X\right)^{-1}\left(I + \frac{1}{2}W_X\right)X_\perp X_\perp^\top\left(I - \frac{1}{2}W_X\right)\left(I + \frac{1}{2}W_X\right)^{-1} ,
\end{aligned}$$

noticing that the order of matrix products containing W_X's can be interchanged freely. As a consequence:

$$\begin{aligned}
Y_\perp Y_\perp^\top W_X Y_\perp Y_\perp^\top &= \left(I - \frac{1}{2}W_X\right)^{-1}\left(I + \frac{1}{2}W_X\right)X_\perp X_\perp^\top\left(I - \frac{1}{2}W_X\right)\left(I + \frac{1}{2}W_X\right)^{-1} \\
&\quad \times W_X \times \left(I - \frac{1}{2}W_X\right)^{-1}\left(I + \frac{1}{2}W_X\right)X_\perp X_\perp^\top\left(I - \frac{1}{2}W_X\right)\left(I + \frac{1}{2}W_X\right)^{-1} \\
&= \left(I - \frac{1}{2}W_X\right)^{-1}\left(I + \frac{1}{2}W_X\right)X_\perp X_\perp^\top W_X X_\perp X_\perp^\top\left(I - \frac{1}{2}W_X\right)\left(I + \frac{1}{2}W_X\right)^{-1} .
\end{aligned}$$

Finally, by the definition of W_X we have that $X_\perp^\top W_X X_\perp = 0$, so that $Y_\perp Y_\perp^\top W_X Y_\perp Y_\perp^\top = 0$. □

Proposition 1 shows that E_t can be easily computed from D_t, and conversely. With the result $W_{Y_t}(E_t^*) = -W_{X_t}(D_t^*)$, we can now prove the equality between $p_{X_t}(D_t)$ and $p_{Y_t}(E_t)$.

Equality between $p_{X_t}(D_t)$ and $p_{Y_t}(E_t)$ A first simple fact can be noted: for X, Y, D, E as in Proposition 1, we have the decompositions $D = XA + X_\perp B$ and $E = -YA + Y_\perp C$. This is the direct consequence of the following:

$$Y^\top W_X Y = X^\top \left(I + \frac{1}{2}W_X\right)^{-1} \left(I - \frac{1}{2}W_X\right) W_X \left(I - \frac{1}{2}W_X\right)^{-1} \left(I + \frac{1}{2}W_X\right) X = X^\top W_X X\,.$$

A second simple fact gets to the desired equality: for a tangent vector D at X, we have the equality $2\|D\|_X^2 = \|W_X(D^*)\|_F^2$. This can be seen from decomposition $D = XA + X_\perp B$, by writing:

$$\begin{aligned} W_X(D^*) &= \left(\frac{1}{2}XA + X_\perp B\right) X^\top - X\left(\frac{1}{2}XA + X_\perp B\right)^\top \\ &= XAX^\top + X_\perp BX^\top - XB^\top X_\perp^\top\,. \end{aligned}$$

The three terms on the right hand side are orthogonal to each other for the Frobenius inner product, so that

$$\begin{aligned} \|W_X(D^*)\|_F^2 &= \left\|XAX^\top\right\|_F^2 + \left\|X_\perp BX^\top\right\|_F + \left\|XBX_\perp^\top\right\|_F \\ &= \|A\|_F^2 + 2\|B\|_F^2 = 2\|D\|_X^2\ . \end{aligned}$$

The previous two facts have two consequences. First, using Proposition 1, we obtain that

$$\|D\|_X^2 = \|W_X(D^*)\|_F^2/2 = \|W_Y(E^*)\|_F^2/2 = \|E\|_Y^2\ .$$

Second, using $\|D\|_X^2 = \frac{1}{2}\|A\|_F^2 + \|B\|_F^2$, we similarly have

$$\|D\|_F^2 = \|A\|_F^2 + \|B\|_F^2 = \|A\|_F^2 + \|D\|_X^2 - \frac{1}{2}\|A\|_F^2 = \|A\|_F^2 + \|E\|_X^2 - \frac{1}{2}\|A\|_F^2 = \|E\|_F^2\ .$$

As a consequence, the isotropic Gaussian probabilities $p_{X_t}(D_t)$ and $p_{Y_t}(E_t)$ are equal, as they only depend on the norms of D_t and E_t, which are equal. Note that, since the norm equality holds for both the canonical norm and the Euclidean norm, both can be used in practice to sample D_t.

Remark. It is likely that stronger relationships can be derived between D and E and their norms. For instance, it can be checked numerically that the weighted norm defined by

$$\|D\|_w^2 = w_1\|D_1\|^2 + \cdots + w_p\|D_p\|^2\ ,$$

with D_k the k-th column of D, coincides for D and E.

Equality of the Jacobian determinants. The equality between the determinants comes from a relationship between the Jacobians $\mathrm{d}C_{X_t}(D_t)$ and $\mathrm{d}C_{Y_t}(E_t)$, seen as linear applications. Before going further, we introduce the extended Cayley transform $\widetilde{C}_X$, which takes as inputs all matrices D such that the expression of $C_X(D)$ is well-defined. The relation between C_X and $\widetilde{C}_X$ is formalized by introducing the canonical injection $\iota_X : T_X\mathcal{V}_{np} \to \mathbb{R}^{n\times p}$: we have $C_X = \widetilde{C}_X \circ \iota_X$. We define $\widetilde{C}_Y(E)$ similarly. Finally, we define the orthogonal matrix

$$K_{XD} = \left(I - \frac{1}{2}W_X(D^*)\right)^{-1}\left(I + \frac{1}{2}W_X(D^*)\right)\ ,$$

such that $\widetilde{C}_X(D) = K_{XD}X$. With these definitions, we can state the relationships between the Jacobians of the Cayley transforms.

Proposition 2. *Let X, Y, D, E as in Proposition 1. For $H \in \mathbb{R}^{n\times p}$, we have the relations*

1. $\mathrm{d}C_X(D) = \mathrm{d}\widetilde{C}_X(D) \circ \iota_X$
2. $\mathrm{d}\widetilde{C}_X(D)(\mathrm{d}D) = \left(I - \frac{1}{2}W_X(D^*)\right)^{-1} W_X(\mathrm{d}D^*) \left(I - \frac{1}{2}W_X(D^*)\right)^{-1} X$
3. $\mathrm{d}\widetilde{C}_Y(E) = M_{XD} \circ \mathrm{d}\widetilde{C}_X(-D) \circ P_X \circ M^*_{XD} \circ D_Y$

with M_{XD} the matrix multiplication by K_{XD}, D_X the bijection mapping $D = XA + X_\perp B$ to $D^* = \frac{1}{2}XA + X_\perp B$, and P_X its inverse. As a consequence, denoting $R_{XD} = \iota^*_X P_X M^*_{XD} D_Y \iota_Y$, we have

$$\mathrm{d}C_Y(E)^* \mathrm{d}C_Y(E) = R^*_{XD} \mathrm{d}C_X(-D)^* \mathrm{d}C_X(-D) R_{XD}\,.$$

Furthermore, R_{XD} is an isometry, so that $J(Y, E) = J(X, -D)$.

Remark. Note that, until this point, D^* and $\mathrm{d}D^*$ denoted $D_X(D)$ and $D_X(\mathrm{d}D)$; E^* and $\mathrm{d}E^*$ denoted $D_Y(E)$ and $D_Y(\mathrm{d}E)$. We keep these conventions in the proof in order to alleviate the equations.

We prove the above proposition below. In order to conclude on the symmetry of the Metropolis proposal, we need to show the equality of matrix determinants $J(X, -D) = J(X, D)$. Unfortunately, we did not manage to prove this equality formally, even in the case $X = I_{np}$ where more explicit formulas are provided by Jauch et al. [2020b]. However, this property can be verified numerically: when drawing random matrices X and D for various values of n and p (X uniform over $\mathcal{V}_{np}$ and D following an isotropic Gaussian distribution with unit variance), and computing the Jacobian matrices $\mathrm{d}C_X(D)$ and $\mathrm{d}C_X(-D)$ using finite differences, we find that the relative difference between $J(X, D)$ and $J(X, -D)$ is always of the order of 10^{-7}. Computing the same quantities using the formulas of [Jauch et al., 2020b, App. C] with tangent vectors at I_{np} gives a relative difference of less than 10^{-15}. The variance used in this experiment is typically larger than the Metropolis transition variance used in this thesis (of the order of 10^{-2}), for which the numerical gap is even smaller.

With this numerical evidence, it comes that $q(X_t, Y_t) \simeq q(Y_t, X_t)$. As a consequence, the Cayley transform of an isotropic Gaussian distribution (isotropic either for the Euclidean or for the canonical measure) produces a (very close to) symmetric random walk over the Stiefel manifold. The Metropolis-Hastings ratio simplifies to

$$\alpha(X_t, Y_t) = \frac{\pi(X_t)}{\pi(Y_t)}\,,$$

which avoids computing the distribution's normalizing constant and the density of the transition kernel.

Proof of Proposition 2. We will prove each statement separately. The adjoint in the definition of $J(X, D)$ in Equation (3.5) can be taken with respect to either the canonical of the Euclidean metric without altering the proof (in particular, R_{XD} is an isometry for both the canonical metric and the Euclidean metric).

Proof of the three relations The first relation $\mathrm{d}C_X(D) = \mathrm{d}\widetilde{C}_X(D) \circ \iota_X$ comes directly from the definition of $\widetilde{C}_X$. The second relation

$$\mathrm{d}\widetilde{C}_X(D)(\mathrm{d}D) = \left(I - \frac{1}{2}W_X(D^*)\right)^{-1} W_X(\mathrm{d}D^*) \left(I - \frac{1}{2}W_X(D^*)\right)^{-1} X$$

is obtained by differentiating each term in $\widetilde{C}_X(D)$ with basic differentiation rules and composing them into the result. The third relation takes stock on the fact that

$$W_Y(E^*) = K_{XD} W_X(K_{XD}^\top \mathrm{d}E^*) K_{XD}^\top\,,$$

which is proved as follows, recalling that $K_{XD}^\top = K_{XD}^{-1}$:

$$\begin{aligned}
W_Y(E^*) &= \mathrm{d}E^* Y^\top - Y(\mathrm{d}E^*)^\top \\
&= \mathrm{d}E^* X^\top K_{XD}^\top - K_{XD} X (\mathrm{d}E^*)^\top \\
&= K_{XD}(K_{XD}^\top \mathrm{d}E^* X^\top - X(\mathrm{d}E^*)^\top K_{XD}) K_{XD}^\top \\
&= K_{XD} W_X(K_{XD}^\top \mathrm{d}E^*) K_{XD}^\top\,.
\end{aligned}$$

With this expression of $W_Y(\mathrm{d}E^*)$, the third equation of the theorem is deduced as follows (using the fact $W_Y(E^*) = -W_X(D^*)$).

$$\begin{aligned}
\mathrm{d}\widetilde{C}_Y(E) &= \left(I - \frac{1}{2}W_Y(E^*)\right)^{-1} W_Y(\mathrm{d}E^*) \left(I - \frac{1}{2}W_Y(E^*)\right)^{-1} Y \\
&= \left(I + \frac{1}{2}W_X(D^*)\right)^{-1} K_{XD} W_X(K_{XD}^\top \mathrm{d}E^* K_{XD}^\top \left(I + \frac{1}{2}W_X(D^*)\right)^{-1} K_{XD} X \\
&= K_{XD} \left(I + \frac{1}{2}W_X(D^*)\right)^{-1} W_X(K_{XD}^\top \mathrm{d}E^*) \left(I + \frac{1}{2}W_X(D^*)\right)^{-1} X \\
&= K_{XD} \left(I + \frac{1}{2}W_X(D^*)\right)^{-1} W_X(D_X(P_X(K_{XD}^\top \mathrm{d}E^*))) \left(I + \frac{1}{2}W_X(D^*)\right)^{-1} X \\
&= \left[M_{XD} \circ \mathrm{d}C_X(-D) \circ P_X \circ M_{XD}^* \circ D_Y\right](\mathrm{d}E)
\end{aligned}$$

Proof that R_{XD} is an isometry The map R_{XD} takes inputs from $T_Y\mathcal{V}_{np}$, and maps them into $T_Y\mathcal{V}_{np}$. The isometry property of R_{XD} can be proved either w.r.t the canonical metric or w.r.t the Euclidean metric. The proof of both cases is very similar, and we only show the Euclidean case. Let $E = YA + Y_\perp B \in T_Y\mathcal{V}_{np}$. We have:

$$\begin{aligned}
M_{XD}^* D_Y \iota_Y(E) &= \frac{1}{2}K_{XD}^\top YA + K_{XD}^\top Y_\perp B \\
&= XA + K_{XD}^\top Y_\perp B\,.
\end{aligned}$$

Since YA and $Y_\perp B$ are orthogonal to each other, and since $K_{XD}^\top$ is an orthogonal matrix, XA and $K_{XD}^\top Y_\perp B$ are orthogonal to each other. In particular, $K_{XD}^\top Y_\perp B$ can be written as $X_\perp C$, with $C = X_\perp^\top K_{XD}^\top Y_\perp B$. We thus get:

$$P_X M_{XD}^* D_Y \iota_Y(E) = XA + X_\perp C\,,$$

which in turn implies that

$$\begin{aligned}
\|R_{XD}(E)\|_F^2 &= \|\iota_X^* P_X M_{XD}^* D_Y \iota_Y\|_F^2 \\
&= \|XA\|_F^2 + \|X_\perp C\|_F^2 \\
&= \|XA\|_F^2 + \left\|K_{XD}^\top Y_\perp B\right\|_F^2 \\
&= \|A\|_F^2 + \|B\|_F^2 \\
&= \|E\|_F^2\,,
\end{aligned}$$

which proves that R_{XD} is an isometry. Combining this property with the third formula of the theorem, we obtain:

$$J(Y,E) = |\mathrm{d}C_Y(E)^* \mathrm{d}C_Y(E)| = |R_{XD}^*| \cdot |\mathrm{d}C_X(-D)^* \mathrm{d}C_X(-D)| \cdot |R_{XD}| = J(X,-D)\,.$$

□

3.5 Model selection with information criteria

This section briefly reviews the derivations of the two main criteria used for model selection: the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC). These criteria will be used and compared in Chapter 7.

Both methods aim at selecting the best model p_m among a family of competitors $p_1, \ldots, p_K$. They proceed by comparing the log-likelihoods $\log p_m(y)$ of each model m for the data y. Comparing the "raw" log-likelihoods however gives a flawed view of the models' performance: it favors the complex models, which for the same amount of data use more parameters to describe the variability. They are thus naturally prone to overfitting the data. Information criteria penalize

the log-likelihoods with terms accounting for the complexity of the model, often measured by the number of parameters.

Note that this section (as well as Chapter 7) does not pretend to exhaustiveness: a very large number of model selection criteria exist and have their own purposes and motivations. We chose to focus on the two most widely used criteria for their simple formulation, their performance and their interpretability. In particular, we chose not to focus on recent criteria taking stock on Bayesian formulations of the model selection problem, namely the Watanabe-Akaike Information Criterion and the Deviance Information Criterion [Watanabe, 2013, Spiegelhalter et al., 2014]. For a general introduction to information criteria and model selection, we refer the reader to Konishi and Kitagawa [2008].

3.5.1 The Akaike Information Criterion

The Akaike Information Criterion (AIC) was initially introduced by Akaike [1998]. At the core, it consists in approximating a model's generalization error. Given a model $p(y \mid \theta)$, we compute $\hat{\theta}(y)$ the MLE given the data y. The data is assumed to come from an unknown distribution q, which might not be of the form $p(\cdot \mid \theta)$. The MLE is an estimator of the point θ_* that minimizes the Kullback-Leibler divergence to the true data distribution:

$$\theta_* = \operatorname{argmin}_\theta \mathrm{KL}[\, q \,||\, p(\cdot \mid \theta)\,] \,.$$

We assume that θ_* is unique and well-defined. Under mild regularity hypotheses, it can be shown that $\hat{\theta}(y)$ converges to θ_* almost surely as the number of samples goes to infinity [van der Vaart, 1998].

Measuring the model performance with $\log p(y \mid \hat{\theta}(y))$ over-estimates the expected log-likelihood on new unseen data. Suppose that the data y was generated from a model $q(y)$. Then the expected error on unseen data is measured by

$$\mathbb{E}_{q(x)}[\log p(x \mid \hat{\theta}(y))] \,.$$

If we average this error over the input data y, we obtain the model's generalization performance:

$$\mathbb{E}_{q(x)}\mathbb{E}_{q(y)}[\log p(x \mid \hat{\theta}(y))] \,.$$

This performance cannot be observed in practice, as only the sample y is available. The AIC assumes that $\hat{\theta}(y)$ is sufficiently close to θ_* to estimate the above quantity using Taylor expansions around $\log p(y \mid \hat{\theta}(y))$. For a clear and detailed derivation of the AIC, we refer the reader to [Burnham and Anderson, 1998, Ch. 6]. After some computations, the authors derive the following asymptotic relation (Eq. 6.31):

$$\mathbb{E}_{q(x)}\mathbb{E}_{q(y)}[\log p(x \mid \hat{\theta}(y))] \simeq \mathbb{E}_{q(y)}\left[\log p(y \mid \hat{\theta}(y))\right] - \mathrm{Tr}\left[\mathcal{J}(\theta_*)\mathcal{I}(\theta_*)^{-1}\right] \,,$$

with the definitions

$$\begin{cases} \mathcal{I}(\theta) = \mathbb{E}_{p(x|\theta)}\left[\left[\nabla_\theta \log p(x \mid \theta)\right]\left[\nabla_\theta \log p(x \mid \theta)\right]^\top\right] \\ \mathcal{J}(\theta) = \mathbb{E}_{q(x)}\left[\left[\nabla_\theta \log p(x \mid \theta)\right]\left[\nabla_\theta \log p(x \mid \theta)\right]^\top\right] \,. \end{cases}$$

The second term of the right-hand side, also called the *trace term*, is the expected difference between the likelihood on unseen data y and the likelihood on the training data y. The AIC then consists in a very simple estimation of the trace term: assuming that the best model $p(y \mid \theta_*)$ is a good approximation of $q(x)$, we have that $\mathcal{J}(\theta_*) \simeq \mathcal{I}(\theta_*)$, so that the trace term is approximately equal to $\dim(\theta)$.

In other words, the AIC, which is computed as $2\log p(y \mid \hat{\theta}(y)) - 2\dim(\theta)$, assumes that the sample size is large enough and that the best model is sufficiently close to $q(y)$ to estimate its generalization performance. In practice, it penalizes the model complexity by the number of degrees of freedom in the parameter space.

As a summary, the crucial hypothesis for AIC to be valid is that the models of interest are good approximations of the true unknown model $q(y)$. It can be shown in specific cases that AIC is equivalent to cross-validation. Both procedures seek to select the model that gives the best generalization performances, measured in terms of log-likelihood.

Remark. Other estimations of the trace term can be considered, and lead to new information criteria. For instance, Monte-Carlo estimation of the matrices $\mathcal{I}$ and $\mathcal{J}$ in the trace term leads to the Takeuchi Information Criterion [Takeuchi, 1976].

3.5.2 The Bayesian Information Criterion

The Bayesian Information Criterion (BIC) was first introduced by Schwarz [1978]; we refer the reader to [Konishi and Kitagawa, 2008, Ch. 9] for a derivation of its formula, which we briefly expose here. The BIC adopts a Bayesian perspective on the model selection problem: the model of interest is a mixture of all candidate models, with a prior $\pi(m)$ on the set of models. In other words, the true model is considered a random variable M chosen among the candidate models $1, \dots, K$:

$$p(y) = \sum_{m=1}^{K} \pi(M = m) p_m(y) \,.$$

Bayesian model selection then consists in choosing model with the highest posterior probability, i.e., find the maximum value of

$$p(M = m \mid y) \propto \pi(M = m) p_m(y) \,.$$

In the above expressions, the model likelihood $p_m(y)$ expresses as an integral over θ_m:

$$p_m(y) = \int p_m(y \mid \theta_m) p(\theta_m) \mathrm{d}\theta_m \,.$$

As the number of samples grows large, the likelihood $p_m(y \mid \theta_m)$ concentrates around the value of the MLE $\hat{\theta}_m(y)$, so that the values outside a small region around $\hat{\theta}_m$ bring almost no contribution to the integral. The BIC proposes to approximate $p_m(y)$ using a Laplace expansion of this integral, which formalizes this intuition. The Laplace expansion of a highly concentrated integrand consists in approximating the integrand by a non-normalized Gaussian distribution. The approximation then equals the normalizing constant of this distribution.

In the case of interest here, it can be shown that:

$$\begin{aligned} \log p_m(y \mid \theta_m) = \log p_m(y \mid \hat{\theta}_m) + (\theta_m - \hat{\theta}_m)^\top \underbrace{\nabla_\theta \log p_m(y \mid \hat{\theta}_m)}_{=0} \\ - \frac{n}{2} (\theta_m - \hat{\theta}_m)^\top I_m(\hat{\theta}_m)(\theta_m - \hat{\theta}_m) + o(\|\theta_m - \hat{\theta}_m\|^2) \,, \end{aligned}$$

with $\mathcal{I}_m$ defined as in the previous section for the model p_m, and n the number of model samples in y. This expression is coherent with the classical result that the asymptotic variance of $\hat{\theta}_m$ is given by $\frac{1}{n}\mathcal{I}(\hat{\theta}_m)^{-1}$ [van der Vaart, 1998].

The Laplace expansion theorem can be applied to the density above: assuming that $\mathcal{I}_m(\hat{\theta}_m)$ is non-singular, we get the approximation

$$p_m(y) \simeq p_m(y \mid \hat{\theta}_m)(2\pi)^{\dim \theta_m / 2} \left[(1/n)^{\dim \theta_m / 2} |\mathcal{I}(\hat{\theta}_m)|^{-1/2} \right] p(\theta_m) \,.$$

Using this equation in the definition of $p(M = m \mid y)$, we obtain:

$$\begin{aligned} \log p(M = m \mid y) &= \log p_m(y) + \log \pi(M = m) - \log p(y) \\ &\simeq \log p_m(y \mid \hat{\theta}_m) - \frac{\dim \theta_m}{2} \log(n) - \log p(y) \\ &\quad - \frac{1}{2} \log |\mathcal{I}(\hat{\theta}_m)| + \frac{\dim \theta_m}{2} \log(2\pi) + \log p(\theta_m) + \log \pi(M = m) \,. \end{aligned}$$

In the above expression, only the first three terms scale with the number of samples n. The third term $-\log p(y)$ does not depend on m, and does thus not affect the model choice. Hence, neglecting the constant factors, we obtain the BIC approximation:

$$BIC = 2 \log p_m(y \mid \hat{\theta}_m) - \dim(\theta_m) \log(n) \,.$$

This approximation is theoretically consistent, in the sense that if the true model is among the candidate models, then the BIC will almost surely choose this model asymptotically. From a conceptual point of view, this criterion is different from the AIC, which seeks the model that best generalizes to unseen data. The BIC asymptotic consistency cannot be achieved in practice, since there is never a "true model". Most statistical models are meant to provide a useful description of a phenomenon rather than a deep understanding of its inner mechanisms.

From a practical perspective, the main difference between BIC and AIC thus resides in the strength of the penalty on model complexity. While the AIC penalty remains constant as the number of samples grows large, the BIC applies a heavier penalty that scales with the logarithm of the number of samples. For this reason, the BIC naturally tends to favor simpler models. As discussed in Chapter 7, the stronger penalty of the BIC makes it robust to optimization errors in the parameter estimation.

Chapter 4

Sparse Low Rank Decomposition for Graph Data Sets

This chapter introduces a non-parametric model for network data sets. We characterize graphs as the sum of a sparse low rank common template and sparse low rank deviations from it. This structure allows accounting for real-world network properties: their adjacency matrices are sparse, and a low rank reflects their organization into clusters as well as the role of core nodes. We propose a variational approach to estimate the template and the deviations based on combined sparse and low rank regularizers. We solve the related optimization problem using classical proximal methods. We demonstrate the performance of our decomposition model on both simulated and real data sets. Analyzing air traffic data, we show that sparsity and low rank lead to interpretable results on the structure of airline traffic.

Contents

4.1 Introduction

Network science and graph theory are at the core of a wide range of applications. When dealing with a data set of networks defined on a same set of nodes, understanding the variability among samples is a crucial issue. In neuroscience, the connections between well-defined brain regions are studied on groups of subjects to better understand the human brain anatomy and function [Obando et al., 2019]. Computational social science requires also to analyze the evolution of interactions in a fixed population [Hanneke et al., 2010]. In this setup, it is interesting to work directly on the networks' adjacency matrices. Yet, up to our knowledge, comparing such matrices remains a difficult problem. Kernel methods can be employed to evaluate distances between networks [Ghosh et al., 2018], but

many theoretically interesting graph kernels require solving NP-hard problems. Other metrics like the cut norm are used in graph theory, but cannot be computed exactly in polynomial time [Alon and Naor, 2004]. Consequently, statistical modeling for networks has so far focused on parametric models like Exponential Random Graph Models, which take stock on a small set of graph features to characterize a data set's variability.

Non-parametric modeling for multiple networks has only very recently started to draw some attention in the literature. Chandna and Maugis [2020] represent networks having a common set of nodes as samples from an extended graphon model. However, this model does not account for classical properties of real-world networks like edge sparsity and low rank. On the other hand, much work has been devoted to extracting a denoised estimate having a sparse structure or a low rank from a noisy matrix [Oymak and Hassibi, 2011, Richard et al., 2012, Zorzi and Chiuso, 2017, Kanada et al., 2018], but these techniques have not been used yet in a statistical framework for graph data sets modeling. The issue we wish to address can be formulated as follows: how can the variability of graph data sets be accounted for in a non-parametric way? For example, given an airline network, can we identify the base traffic and how daily traffic fluctuations are structured ?

In this chapter, we model the adjacency matrices of weighted networks sharing a common set of nodes as additive deviations from a template network which represents the reference interactions. These template and deviations have a sparse low rank structure corrupted by a sparse noise. We estimate this decomposition using variational matrix recovery techniques. The Douglas-Rachford algorithm and the Generalized Forward-Backward algorithm [Raguet et al., 2013] allow solving the related convex optimization problems. We show that taking advantage of the networks' common structure allows efficiently estimating the uncorrupted data. Finally, we show how this decomposition operates on real data sets.

4.2 Related work

The recovery of low rank patterns has been widely addressed in the literature, in domains like robust principal component analysis [Candès et al., 2011], covariance matrix estimation [Koltchinskii et al., 2016] or link prediction/matrix completion [Candes and Plan, 2010]. Many of them rely on singular values analysis and the use of the nuclear norm, and consider sparse noise [Oymak and Hassibi, 2011, Zorzi and Chiuso, 2017, Kanada et al., 2018].

More recently, efforts have been made to denoise matrices with both low rank and sparse structure. Richard et al. [2012] reconstruct such matrices by combining the nuclear and ℓ_1 penalties and obtain convincing results on both simulated and real data. The same idea is used by Zhou et al. [2015] to reconstruct covariance matrices, with additional theoretical guarantees on the convergence rate.

When it comes to averaging networks or adjacency matrices, little has been done in the literature so far. Kernel methods allow computing the mean of several graphs, but the result is not always a graph in itself. Stochastic block-models divide the nodes of a graph into a fixed known number K of clusters with simple interactions [Peixoto, 2020]. This amounts to approximating the adjacency matrix by a block-wise constant matrix with rank less than K. The recent work of Chandna and Maugis [2020] allows handling multiple adjacency matrices in a single mathematical object. However, their work does not compare directly to ours, as they focus on a generalized smooth graphon estimation procedure which assumes that each network is represented by a one-dimensional latent code. Aggregating adjacency matrices however seems an important step toward developing coherent complete statistical framework for graph data sets analysis, and is at the core of the model and algorithms we propose.

4.3 Model and algorithms

We denote $\|\cdot\|_F$, $\|\cdot\|_1$ and $\|\cdot\|_*$ respectively the Frobenius (ℓ_2) norm, the ℓ_1-norm and the nuclear norm (sum of singular values) over the set of adjacency matrices with a given number of nodes m.

4.3.1 Model setup

We are interested in data sets of weighted, possibly non-symmetric adjacency matrices $A_1, ..., A_n$. We model each $A_i \in \mathbb{R}_+^{m \times m}$ as deriving from a common template matrix T. In the example of the airline network in the introduction, T can be the daily traffic load and $A_i - T$ the traffic difference due to sample-specific circumstances and random fluctuations. The difference between A_i and T is thus decomposed into a deviation V_i and noise ε_i. The decomposition reads:

$$A_i = T + V_i + \varepsilon_i \tag{4.1}$$

The template adjacency matrix T has non-negative coefficients, as do the samples $(A_i)_{i=1}^n$. The template T is assumed to be sparse and have a low-rank structure. Although the network's perturbation V_i is not an adjacency matrix in itself, it also seems reasonable to consider it sparse and low rank. For instance, if a matrix A_i differs from the template only in the connections of one given node with its neighbors, the resulting deviation has rank 2. Finally, since A_i is sparse the noise ε_i is also sparse.

We want to remove the noise and separate the template from the deviations *simultaneously*. To that end, we propose a variational formulation for the decomposition of Eq. (4.1). Our approach follows the idea of Richard et al. [2012]. In order to denoise an adjacency matrix A, Richard et al. solve a convex optimization problem to estimate an underlying sparse and low rank ground truth:

$$S \in \operatorname*{argmin}_S \left\{ \ell(S, A) + \tau \|S\|_1 + \tau \|S\|_* \right\}, \tag{SPLR}$$

where ℓ is a convex loss. This formulation takes stock on both the sparsity-inducing property of the ℓ_1 penalization and the nuclear norm regularizer, often considered as the convex relaxation of the matrix rank. Richard et al. [2012] showed that using simultaneously both penalties improves the matrix recovery in terms of support and Root Mean Square Error (RMSE). Later, a better penalty was proposed by Richard et al. [2013] to account for both the sparsity and low rank property in a single regularizer, but the related estimation procedure turns out to be very computationally demanding, which led us to stick to the first regularization method.

As a generalization of the previous framework, we propose to reconstruct the template T and the deviations $V = (V_i)_{i=1}^n$ by solving the following convex optimization problem:

$$(T, V) \in \operatorname*{argmin}_{T, V_1, \dots, V_n} \left\{ \lambda \|T\|_* + \rho \|T\|_1 + \sum_{i=1}^n \ell(A_i, T + V_i) + \mu \|V_i\|_* + \nu \|V_i\|_1 \right\}. \tag{SPLRD}$$

As the noise $\varepsilon = (\varepsilon_i)_{i=1}^n$ in our model is sparse, the ℓ_1 loss seems a natural candidate for ℓ. This choice has proven useful to denoise adjacency matrices [Kanada et al., 2018]. We also implemented another version of the algorithm using the ℓ_2 loss for comparison purposes.

4.3.2 Algorithms

We propose two algorithms to solve the optimization problems related respectively to the models with the ℓ_1 and ℓ_2 losses. The objective functions we consider are non-differentiable, and gradient descent methods can not be employed. The standard way to deal with this problem is to use proximal methods. Given a non-differentiable, yet simple function $g(x)$, the proximal operator of g is given by:

$$\operatorname{prox}_{\tau g}(x) = \operatorname*{argmin}_z \left\{ \frac{1}{2} \|x - z\|_F^2 + \tau g(z) \right\}.$$

Algorithm for the ℓ_1 loss

The objective function is the sum of simple functions whose proximal operator is known. The Douglas-Rachford algorithm [Boyd et al., 2011] can be employed here. We introduce duplicate

variables T_*, T_1, V_*, V_1 and Y and add equality constraints. The loss becomes:

$$\mathcal{L}(T_*, T_1, V_*, V_1, Y) = \lambda \|T_*\|_* + \rho \|T_1\|_1 + \chi_{\{T_*=T_1\}} + \sum_{i=1}^{n} \|Y_i - A_i\|_1 + \mu \|V_{*,i}\|_* + \nu \|V_{i,1}\|_1 + \chi_{\{V_{*,i}=V_{1,i}\}} + \chi_{\{V_{*,i}+T_*=Y\}}$$

where χ_E denotes the characteristic function of the set E, which takes value 0 over E and $+\infty$ elsewhere. Let $X = (T_*, T_1, V_*, V_1, Y)$, the full loss thus writes $\mathcal{L}(X) = f(X) + \chi_{(T_1,V_1,Y)=P(T_*,V_*)}$ with P a linear operator and $f(X) = f_1(T_*) + f_2(T_1) + f_3(V_*) + f_4(V_1) + f_5(Y)$, where each f_i has known proximal operator.

Known proximal operators. The proximal operator of f is given by those of the f_i:

$$\text{prox}_{\tau f}(X) = (\text{prox}_{\tau f_1}(T_*), ..., \text{prox}_{\tau f_5}(Y)) .$$

Furthermore, each $\text{prox}_{\tau f_i}$ derives from the following [Parikh and Boyd, 2014]:

- ℓ_1*-norm.* Let $\text{ST}_\tau(X) = \text{sgn}(X) \odot (\text{abs}(X) - \tau)_+$ the soft thresholding operator. Then the proximal operator for the ℓ_1-norm is given by $\text{prox}_{\tau\|\cdot\|_1}(X) = \text{ST}_\tau(X)$. Consequently,

$$\text{prox}_{\tau\|\cdot - A\|_1}(X) = \text{ST}_\tau(X - A) + A .$$

- *Nuclear norm.* Given a matrix X, let $X = U\text{Diag}(\lambda)V^T$ be the singular value decomposition of X. Then we have $\text{prox}_{\tau\|\cdot\|_*}(X) = U\text{Diag}(\text{ST}_\tau(\lambda))V^T$

Linear projection. The proximal operator of the indicator function corresponds to projecting onto the set of constraints $\mathcal{C} = \{X \mid (T_1, V_1, Y) = P(T_*, V_*)\}$. This projection can be computed explicitly:

Proposition (Peyré [2018], proposition 42)**.** *Let P be a linear operator. The projection of (x, y) onto the set $\{(x, y) \mid y = Px\}$ is given by $\tilde{x} = (I + P^TP)^{-1}(P^Ty + x)$, $\tilde{y} = P\tilde{x}$.*

For $x \in \{V_*, V_1, Y\}$, let $\bar{x} = \frac{2}{2n+6}\sum_i x_i$. In our optimization problem, this proposition writes:

$$\begin{cases} \tilde{T}_* = \frac{3}{2n+6}(T_* + T_1) + \bar{V}_1 - \bar{V}_* - \bar{Y} \\ \tilde{V}_{*,i} = -\frac{1}{2n+6}(T_* + T_1) - \frac{1}{3}\bar{V}_1 + \frac{1}{6}\bar{V}_* + \frac{1}{6}\bar{Y} + \frac{1}{3}(V_{*,i} + V_{1,i} + Y_i) \end{cases}$$

And $\tilde{T}_1 = \tilde{T}_*, \tilde{V}_{1,i} = \tilde{V}_{*,i}, \tilde{Y}_i = \tilde{V}_{*,i} + \tilde{T}_*$. For the computation details, we refer the reader to Appendix 4.A.

Douglas-Rachford splitting. The numerical scheme is detailed in algorithm 4.3.1. It only requires choosing parameters θ_{DR} for the total step size and τ_{DR} for the proximal step size. Unless specified, the Douglas-Rachford algorithm is run for 200 iterations for every experiment in this chapter with $\theta_{DR} = 0.9$ and $\tau_{DR} = 0.1$. We return the global variable X_t in its non-projected version for two reasons. First, once the algorithm has converged, X_t is extremely close to $\text{Proj}_{\mathcal{C}}(X_t)$. Second, the projection $\text{Proj}_{\mathcal{C}}$ does not preserve sparsity nor low rank. On the contrary, T_1 and V_1 have null coefficients, and T_* and V_* have null singular values. Returning these different values facilitates the quantitative analysis of the solution.

Algorithm 4.3.1: Optimization for the model with the ℓ_1 loss (Douglas-Rachford)

Initialize X_0 with $T_* = T_1 = \frac{1}{n}\sum_i A_i$, $V_{*,i} = V_{1,i} = A_i - T_*$, $Y_i = T_* + V_{*,i}$
repeat
  $Z^0_{t+1} = 2X_t - \text{Proj}_{\mathcal{C}}(X_t)$
  $Z^1_{t+1} = 2Z^0_{t+1} - \text{prox}_{\tau f}(Z^0_{t+1})$
  $X_{t+1} = (1 - \theta)X_t + \theta Z^1_{t+1}$
until *convergence*
return $X_{final} = (T_*, T_1, V_*, V_1, Y)$

Algorithm for the Frobenius norm

The case of the Frobenius norm is simpler:

$$(T,V) \in \underset{T,V_1,\dots,V_n}{\operatorname{argmin}} \left\{ \lambda \|T\|_* + \rho \|T\|_1 + \sum_{i=1}^{n} \|A_i - T - V_i\|_F^2 + \mu \|V_i\|_* + \nu \|V_i\|_1 \right\}.$$

Since the objective function now contains a differentiable term, we use the Generalized Forward-Backward algorithm [Raguet et al., 2013]. Let $F(T,V) = \sum_i \|T + V_i - A_i\|_F^2$. The GFB algorithm uses the gradient of F and results in a simpler procedure. The numerical scheme is detailed in algorithm 4.3.2. It requires two parameters θ_{GFB} for the total step size and τ_{GFB} for the gradient step and proximal step sizes. In the experiments of this chapter the Generalized Forward-Backward algorithm is run for 200 iterations with $\theta_{GFB} = 0.1$ and $\tau_{GFB} = 1.5$.

Algorithm 4.3.2: Optimization for the model with the ℓ_2 loss (Generalized Forward-Backward)

Initialize $T = \frac{1}{n}\sum_i A_i$, $V_i = A_i - T$, $T_* = T_1 = T$, $V_* = V_1 = V$
repeat
  $T_* = T_* + \theta(\text{prox}_{\tau\lambda\|\cdot\|_*}(2T - T_* - \tau\nabla_T F(T,V)) - T)$
  $T_1 = T_1 + \theta(\text{prox}_{\tau\rho\|\cdot\|_1}(2T - T_1 - \tau\nabla_T F(T,V)) - T)$
  for $i \in [\![1,n]\!]$ **do**
    $V_{*,i} = V_{*,i} + \theta(\text{prox}_{\tau\mu\|\cdot\|_*}(2V_i - V_{*,i} - \tau\nabla_{V_i} F(T,V)) - V_i)$
    $V_{1,i} = V_{1,i} + \theta(\text{prox}_{\tau\nu\|\cdot\|_1}(2V_i - V_{1,i} - \tau\nabla_{V_i} F(T,V)) - V_i)$
  end
  $T = (T_* + T_1)/2$
  $V = (V_* + V_1)/2$
until *convergence*
return $X_{final} = (T_*, T_1, V_*, V_1)$

As a summary, both algorithms take as input n adjacency matrices $A_1, ..., A_n$ and two step size parameters. Once they have converged they yield estimates for the template $T_1 \simeq T_*$ and the deviations $V_{1,i} \simeq V_{i,*}$. As the most time-consuming iteration is the SVD applied in the proximal operators, the time complexity for one step is $O(nm^3)$. With a 2.9GHz Intel Core i5 double core processor, 200 iterations are completed on 10 graphs with 100 nodes under twenty seconds. The implementation for both algorithms as well as reproducible code for the experiments can be found in the supplementary material.

4.4 Experiments on simulated data

We present two experiments on simulated data. First, we perform sparse low rank (SPLR) decomposition on a simple visual example. In the second experiment we consider more complex simulated data. We study a case where T and V are not always non-negative and have null expectation, and a case where both T and V have non-negative symmetric coefficients. The latter allows obtaining adjacency matrices and compare graph features.

4.4.1 A visual example

In order to get a first grasp on the decomposition action on graph data sets, we first study the simple case of a population with 100 nodes split into five communities sparsely interacting with each other. We consider a binary template T corresponding to five fully connected communities with random sizes, and no interaction between communities. In $n = 10$ occurrences, all individuals between two random indices $p < q$ interact with individuals between random indices $k < l$, forming an additional block of interactions. Finally, we randomly flip 20% of the edges and get directed adjacency matrices A_i. We wish to identify the communities, and for each network A_i the additional interaction.

We tune the parameters $(\lambda, \rho, \mu, \nu)$ using the `forest_minimize` function as a black box optimizer from the `scikit-optimize` library [Head et al., 2018]. We search the parameters in the space $[0, 50] \times [0, 5] \times [0, 50] \times [0, 5]$ with a uniform prior for 100 iterations, minimizing the RMSE for the deviations $V_i = A_i - T$. We used $\theta_{DR} = 0.1$ and ran the algorithm for 400 iterations because of the strong noise amplitude. The results are shown in Figure 4.1. It can be seen that the template is recovered accurately despite the small number of samples ($n = 10$), and most of the noise has been removed from the variation V_i, even though the estimation is less accurate than T's. This is coherent since the template benefits from multiple samples and the noise considered in strong. Up to a 0.1 threshold which accounts for the residual noise, 100% of the support was correctly recovered for T and 92% for V. The stripes in the estimates are characteristic of low rank reconstruction methods.

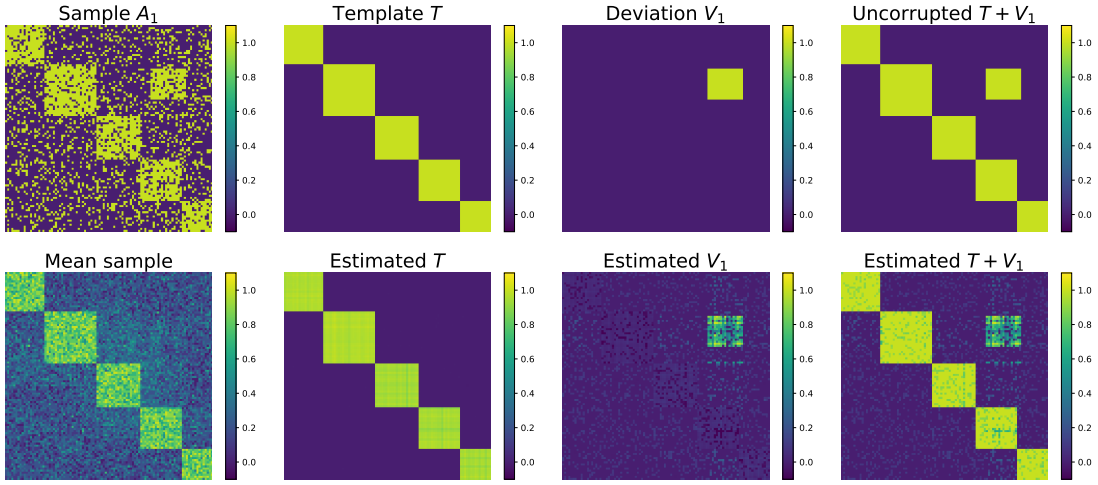

Figure 4.1: Estimation results for a simulated example with simple cluster structure. This figure shows the sample, the mean sample, the ground truth template and deviation and their estimates.

4.4.2 More complex simulated data

Unconstrained sign. We generate non-symmetric random sparse low rank matrices T and V_i using the method described in Appendix 4.B to obtain a data set of matrices $A_i = T + V_i + \varepsilon_i$. The matrix coefficients are not non-negative and thus do not represent actual networks. The ranks of T and the V_i's, and the sparsity level in T, V and ε can be chosen freely. Here we use $n = 10$ samples, $m = 100$ nodes, $\mathrm{rk}(T) = \mathrm{rk}(V_i) = 10$, and 70% sparsity in T and the V_i's. The results we present do not vary a lot with these parameters. As proved in Appendix 4.B, the non-zero coefficients in T and V have standard deviation $\sqrt{\mathrm{rk}(T)}(1 - s(T)^{1/r})/(1 - s(T)) \simeq 1$. The noise ε_i also has sparsity 0.7, and non-zero coefficients follow a Gaussian distribution with standard deviation 1.

Non-negative symmetric coefficients. Next, we simulate matrices that may represent actual weighted undirected networks. To that end we perform the same experiment, except now we impose that T and V have non-negative, symmetric coefficients. The noise ε is drawn from a symmetric sparse Gaussian distribution ε_0 thresholded to get non-negative coefficients: $\varepsilon = \max(\varepsilon_0, -T - V)$.

Optimization details. We perform parameter selection with the `forest_optimize` function of the `scikit-optimize` library with default arguments, and select the parameters with smallest RMSE for V. We search the parameters in the space $[0, 10] \times [0, 10] \times [0, 10] \times [0, 10]$ for the ℓ_1 loss, and $[0, 20] \times [0, 10] \times [0, 20] \times [0, 10]$ for the ℓ_2 loss. This metric is optimized on 5 random data sets and evaluated on 5 other random data sets drawn with the same parameters.

Model evaluation. For both data sets we compute the relative RMSE for the estimation of T and V. To the best of our knowledge, there is no direct concurrent method to compare our algorithm with. In order to highlight the benefit of the joint estimation of T and V, we perform a sparse low rank estimation on the empirical mean of the adjacency matrices $M = \frac{1}{n}\sum_i A_i$, using the original method from Richard et al. [2012] with the ℓ_1 loss, selecting the parameters on the training data set with `scikit-optimize` as for our model. This gives us an estimate T^M of T. We then apply the same method to $A_i - T^M$ after performing parameter selection on the training data set, and thus get an estimate V^M of V. These results, as well as the naive decomposition $A_i = M + (A_i - M)$, are compared to the solutions of the joint optimization problems in Table 4.1.

Graph features. Table 4.1 also shows the relative RMSE for the value of several graph features. We compared the average weighted node degree d, the clustering coefficient C and the average shortest path length L. The average shortest path length L was computed with the NetworkX package [Hagberg et al., 2008] using a decreasing function of the edge weights, accounting for the cases where these weights represent a connection intensity rather than a cost. We computed a weighted version C of the clustering coefficient proposed by Opsahl and Panzarasa [2009]. For more details on the definition and computation of graph parameters, we refer the reader to Appendix 4.C.

Comments. In both cases, our decomposition method greatly improves the estimation of the template compared to the denoised mean T^M. The recovery of T is achieved with very accurately with few samples ($n = 10$). Consequently, the estimation of V is improved with respect to V^M, which is only natural since the estimation of each V_i^M is based on a biased matrix $A_i - T^M$. On the other hand, the error of our decomposition model for V_i is within the expected values for the one-sample SPLR estimation of Richard et al. [2012]. It can be noticed that, while the estimation performance of the mean sample and its denoised version worsen when using non-negative coefficients, the performance of the SPLR decomposition stays good. This result points out the difference between the template T and the average sample. The ℓ_1 loss almost always beats the ℓ_2 loss in Table 4.1, which is coherent since the data was simulated with a sparse noise at first. The results on graph features also show that the SPLR decomposition estimates the initial samples in a way that is consistent with a graph structure. This is not surprising for the average degree which depends linearly on the adjacency matrix, but the average shortest path length and the clustering coefficient estimations are also improved with respect to the noisy samples. Again, the ℓ_1 loss outperforms the ℓ_2 loss, except for the average shortest path. This difference will be analyzed in future work.

Table 4.1: Mean and standard deviation of the relative RMSE. For matrices with unconstrained signs we show the error for T and V, and for symmetric non-negative coefficients the error for T, V and the graph features presented in the text.

		SPLRD(ℓ_1)	SPLRD(ℓ_2)	(T^M, V^M)	$(M, A_i - M)$
Unconstrained sign	T	$\mathbf{.04 \pm .01}$	$.34 \pm .06$	$.42 \pm .08$	$.55 \pm .10$
	V	$\mathbf{.33 \pm .02}$	$.41 \pm .02$	$.49 \pm .02$	$.91 \pm .05$
Positive symmetric	T	$\mathbf{.07 \pm .02}$	$.25 \pm .08$	$.55 \pm .22$	$.64 \pm .27$
	V	$\mathbf{.30 \pm .01}$	$.32 \pm .01$	$.55 \pm .02$	$.77 \pm .03$
Graph features	d	$\mathbf{.01 \pm .01}$	$.12 \pm .02$	$.09 \pm .04$	$.27 \pm .09$
	L	$.15 \pm .07$	$\mathbf{.09 \pm .07}$	$.29 \pm .08$	$.49 \pm .07$
	C	$\mathbf{.03 \pm .01}$	$.34 \pm .04$	$.35 \pm .03$	$.09 \pm .02$

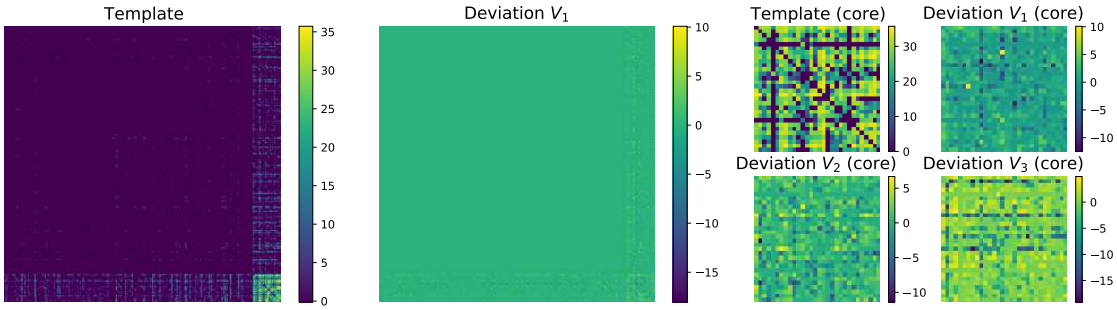

Figure 4.2: Decomposition result for US flights data set. This figure shows the template T and the deviation V_1, and a zoom on core cities for T, V_1, V_2 and V_3.

4.5 Experiments on real data

We apply our model on two data sets, which illustrate the decomposition into sparse low rank template and deviations. Both data sets are available online under CC-By Attribution 4.0 International license. In view of the results in the previous section, we use the decomposition model with ℓ_1 loss. For both applications, we need to select parameters to use our algorithm. There is no definite rule to select these parameters, as often with variational methods. We chose the parameters which allow observing low rank patterns while keeping the decomposition $T + V_i$ close to A_i.

4.5.1 Airplane traffic network

We use a data set of networks from Williams and Musolesi [2016] listing US domestic airplane flights every hour for 10 days. We sum the total flight count per day to average day/night variability and get $n = 10$ directed weighted adjacency matrices. The obtained networks have $m = 299$ nodes corresponding to airports. The matrices have 95% null coefficients and rank at most 123, with very rapidly decaying singular values. We perform SPLR decomposition with the ℓ_1 penalty. We used $(\lambda, \rho, \mu, \nu) = (10, 0.1, 5, 0.1)$, which allowed to observe an interesting result while staying close to the data (8% relative RMSE between samples and the $T + V_i$'s for the noise removed).

The estimated template has rank 113 and sparsity 94%. The deviations have rank below 85 and average sparsity 95%. As a comparison, the samples have rank at least 110, with the mean sample having rank 123. The template and the three first deviations are shown in Figure 4.2. The cities are ordered according to an algorithmically detected core/periphery structure [Kojaku and Masuda, 2018] and grouping the nodes from the core and the periphery together.

The recovered template T contains the regular traffic; it is high between the core airline hubs, weak between the core and the periphery and almost null between nodes from the periphery. The deviations V_i's account for the fluctuations in the flights planning, as well as events affecting airports. In Figure 4.2, V_1 is mainly constituted of fluctuations in the core and the impact a few of the core cities onto the periphery nodes. This impact directly translates into a low rank perturbation: the strongest pattern in V_1 is a reduced activity for New York's western airport (10th core node), which is close to a rank 2 perturbation. Two symmetric bright spots indicate unusually high traffic between Denver and New York's eastern airport. Similar observations can be made for V_2 and V_3.

4.5.2 Functional brain networks

Next we consider a data set of human brains analyzed with functional Magnetic Resonance Imaging (fMRI) in Waschke et al. [2018]. The nodes represent brain regions and the connections the temporal correlations between their activity signals. The resulting network represents the brain functional connectivity, i.e., which brain region tends to activate with each other. Correlation matrices are not actual adjacency matrices since they take negative values, however the conversion can be done easily by taking the weights' absolute value. The study considers $n = 49$ subjects with $m = 312$ brain regions. Each subject is recorded during a resting period and while performing

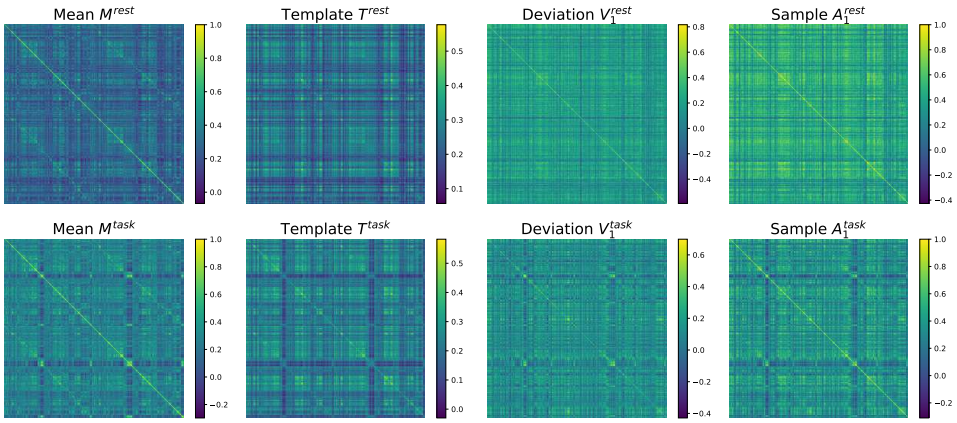

Figure 4.3: Mean sample matrices M^{rest}, M^{task}, templates, and deviation and original sample for the first subject at rest (top row) and performing a task (bottom row)

a listening task. For more details on the data processing, we refer the reader to Waschke et al. [2018] The adjacency matrices in the data set are not sparse but have very low rank, with only one or two leading singular values. Our decomposition model can be employed using $\rho = \nu = 0$ and turn into a low rank template + low rank deviation estimation method. Setting $\lambda = 50, \mu = 5$, we run the algorithm first separately on the 49 resting state and the 49 active listening matrices, then together on the 98 matrices. The three decompositions $(T^{\text{rest}}, V^{\text{rest}}), (T^{\text{task}}, V^{\text{task}}), (T^{\text{full}}, V^{\text{full}})$ approximate the samples with relative RMSE at most 3.9%. The results are shown in Figure 4.3.

The templates $T^{\text{rest}}, T^{\text{task}}$ have relatively low rank (resp. 70 and 166), whereas the template T^{full} has rank 284 out of 312, showing that our model is more relevant for each separate task than for the global data set. Over all three models, the deviation ranks range between 54 and 97. Moreover, the ℓ_2 distance between T^{task} and T^{full} is 37% larger than that between T^{rest} and T^{full}. It indicates that the best template for the whole data set is closer to the resting state brain. This observation can be linked with the results on simulated data with non-negative deviations V_i: the decomposition identifies a template that best fits the $T + V_i$ model rather than the average sample. The templates provide a sharper perspective than the means for which region's activity changes when performing a task. Note that these templates do not have a unit diagonal, which is inconsistent with a correlation matrix structure, but coherent since the diagonal is a full rank term.

4.6 Conclusion

Summary. We have presented a new method to decompose data sets of matrices into a sparse low rank template, sparse low rank deviations and noise. This model relies on observable properties of real-world adjacency matrices. We have proposed a procedure to fit this model by optimizing a convex loss function using known reliable algorithms. These algorithms have proven very efficient to estimate the underlying template from few samples, recovering the deviations with acceptable accuracy. We proved that this model is relevant to handle graph data sets and showed that the algorithm is able to recover graph features from noisy simulated data. This decomposition is suited to real networks like airplane flights and gives a meaningful interpretation for the variability.

Future directions. The framework studied here opens a natural question: what would be the statistical modeling counterpart of the optimization problem we considered? The objective function to be minimized does not correspond to the log-likelihood of a model likely to produce sparse samples, and could thus not be used directly to define a probabilistic distribution. In the next chapter we try to address this issue. We propose a hierarchical statistical model that naturally accounts for the low-rank property of the observed matrices, while allowing to produce sparse matrices.

An interesting transition between both chapters can be made by mentioning the work of Durante et al. [2017]. In this paper, the authors propose to model binary network matrices as Bernoulli samples of weighted matrices. The weighted matrix of each individual is defined as the sum of a fixed template and a random low-rank deviation. The authors propose a statistical model to generate these low-rank deviations comparable to the method we used to generate synthetic data in this study. In the next chapter, we will be considering that the individual deviation consists in a variation of the template eigendecomposition, rather than an addition to it. Working with a parametric model for low-rank matrices will allow estimating these new deviations.

Finally, a comparison can be made between the approach proposed in this chapter and low-rank tensor approximation methods, which here would consist in finding a low-rank truncation of the (N, n, n) tensor $(A_1, \dots, A_N)$ or the tensor $(V_1, \dots, V_N)$. Such methods provide an interesting alternative to ours, in the sense that the shared low-rank constraint allows leveraging the similarities between the deviations from the template. They are thus well adapted to describe data sets where the deviations are strongly structured and share common characteristics from one network to another. On the contrary, our model makes no assumption on the similarities between the deviations from the template; it is thus likely to perform better on examples such as in Section 4.4.1, where the deviation structure may strongly vary from one network to another.

4.A Douglas Rachford linear projection

The problem we consider writes as

$$\min_{T_*, T_1, V_*, V_1, Y} \lambda\|T_*\|_* + \rho\|T_1\|_1 + \sum_{i=1}^{n} \|Y_i - A_i\|_1 + \mu\|V_{*,i}\|_* + \nu\|V_{1,i}\|_* + \chi_{\{Y_i = T_* + V_{*,i}\}} + \chi_{\{V_1 = V_*\}} + \chi_{\{T_1 = T_*\}} \,.$$

The constraints can be summarized as $P\binom{T_*}{V_*} = \begin{pmatrix} T_1 \\ V_1 \\ Y \end{pmatrix}$ with P defined as

$$P = \begin{bmatrix} 1 & 0_n^T \\ 0_n & I_n \\ \mathbb{1}_n & I_n \end{bmatrix} \otimes I_m = P_0 \otimes I_m \in \mathbb{R}^{m(2n+1) \times m(n+1)} \,.$$

As explained in the main text, the linear projection onto this set of constraints is known: we seek to compute $\tilde{x} = (I + P^T P)^{-1}(P^T y + x)$ and $\tilde{y} = P\tilde{x}$, with $x = \binom{T_*}{V_*}$ and $y = \begin{pmatrix} T_1 \\ V_1 \\ Y \end{pmatrix}$. We have:

$$\begin{aligned} I_{m(n+1)} + P^T P &= (I_{n+1} + P_0^T P_0) \otimes I_m \\ &= \left(I_{n+1} + \begin{bmatrix} 1 & 0_n^T & \mathbb{1}_n^T \\ 0_n & I_n & I_n \end{bmatrix} \begin{bmatrix} 1 & 0_n^T \\ 0_n & I_n \\ \mathbb{1}_n & I_n \end{bmatrix} \right) \otimes I_m \\ &= \left(I_{n+1} + \begin{bmatrix} m+1 & \mathbb{1}_n^T \\ \mathbb{1}_n & 2I_n \end{bmatrix} \right) \otimes I_m \\ &= \begin{bmatrix} m+2 & \mathbb{1}_n^T \\ \mathbb{1}_n & 3I_n \end{bmatrix} \otimes I_m = Q_0 \otimes I_m \end{aligned}$$

Furthermore, is can be checked that the inverse of Q_0 is:

$$Q_0^{-1} = \frac{1}{2m+6} \begin{bmatrix} 3 & -\mathbb{1}_n^T \\ -\mathbb{1}_n & \frac{1}{3}\mathbb{1}_n\mathbb{1}_n^T \end{bmatrix} + \frac{1}{3} \begin{bmatrix} 0 & 0_n^T \\ 0_n & I_n \end{bmatrix}$$

On the other hand, we have:

$$\begin{aligned} P^T \begin{pmatrix} T_1 \\ V_1 \\ Y \end{pmatrix} + \begin{pmatrix} T_* \\ V_* \end{pmatrix} &= \left(\begin{bmatrix} 1 & 0_n^T & \mathbb{1}_n^T \\ 0_n & I_n & I_n \end{bmatrix} \otimes I_m \right) \begin{pmatrix} T_1 \\ V_1 \\ Y \end{pmatrix} + \begin{pmatrix} T_* \\ V_* \end{pmatrix} \\ &= \begin{pmatrix} Y_1 + ... + Y_n + T_1 + T_* \\ V_1 + Y + V_* \end{pmatrix} \end{aligned}$$

Which gives:

$$\begin{aligned}
\begin{pmatrix}\tilde{T}_* \\ \tilde{V}_*\end{pmatrix} &= (I_{m(n+1)} + P^T P)^{-1} \left(P^T \begin{pmatrix} T_1 \\ V_1 \\ Y \end{pmatrix} + \begin{pmatrix} T_* \\ V_* \end{pmatrix} \right) \\
&= (Q_0^{-1} \otimes I_m) \left(P^T \begin{pmatrix} T_1 \\ V_1 \\ Y \end{pmatrix} + \begin{pmatrix} T_* \\ V_* \end{pmatrix} \right) \\
&= \begin{pmatrix} \frac{3}{2m+6}(T_* + T_1) + \bar{Y} - \frac{1}{2}\bar{V}_* - \frac{1}{2}\bar{V}_1 \\ -\frac{1}{2m+6}(T_* + T_1) - \frac{1}{3}\bar{Y} + \frac{1}{6}\bar{V}_* + \frac{1}{6}\bar{V}_1 \\ \vdots \\ -\frac{1}{2m+6}(T_* + T_1) - \frac{1}{3}\bar{Y} + \frac{1}{6}\bar{V}_* + \frac{1}{6}\bar{V}_1 \end{pmatrix} + \frac{1}{3} \begin{pmatrix} 0 \\ Y_1 + V_{*,1} + V_{1,1} \\ \vdots \\ Y_n + V_{*,n} + V_{1,n} \end{pmatrix} \\
&= \begin{pmatrix} \frac{3}{2m+6}(T_* + T_1) + \bar{Y} - \frac{1}{2}\bar{V}_* - \frac{1}{2}\bar{V}_1 \\ -\frac{1}{2m+6}(T_* + T_1) - \frac{1}{3}\bar{Y} + \frac{1}{6}\bar{V}_* + \frac{1}{6}\bar{V}_1 + \frac{1}{3}Y_1 + \frac{1}{3}V_{*,1} + \frac{1}{3}V_{1,1} \\ \vdots \\ -\frac{1}{2m+6}(T_* + T_1) - \frac{1}{3}\bar{Y} + \frac{1}{6}\bar{V}_* + \frac{1}{6}\bar{V}_1 + \frac{1}{3}Y_n + \frac{1}{3}V_{*,n} + \frac{1}{3}V_{1,n} \end{pmatrix}.
\end{aligned}$$

And we thus also obtain $\tilde{T}_1 = \tilde{T}_*$, $\tilde{V}_1 = \tilde{V}_*$ and $\tilde{Y} = \tilde{T}_* + \tilde{V}_*$.

4.B Generating random sparse low rank matrices

Random low rank matrices can be generated from their Singular Value Decomposition (SVD):

$$M = U\mathrm{Diag}(\sigma_1, ..., \sigma_r, 0, ..., 0)V^T$$

With U and V chosen uniformly among orthogonal matrices and $\sigma \sim \mathcal{N}(0, I_r)$. However, this method almost never produces sparse matrices: random orthogonal matrices are not sparse, and there does not seem to be a simple way to draw uniformly from a distribution of sparse orthogonal matrices.

A convenient way to overcome this hurdle is to give up on the orthogonality constraint on the columns of an orthogonal matrix. We keep the matrix form $M = U\mathrm{Diag}(\sigma_1, ..., \sigma_r, 0, ..., 0)V^T$ with σ defined as before. U, V are now generated with $s\%$ null coefficients selected with Bernoulli variables, and the others drawn from a Gaussian distribution. Such matrices U and V may not have full rank, so we further impose $\mathrm{rk}(M) = r$. Numerically, it amounts to repeating the process as long as M does not have sufficient rank.

While the new matrix distribution induced by the form of M is certainly not equivalent to the one we could be hoping for (sample U and V from sparse orthogonal matrices), it does produce matrices with desired rank and sparsity level. Let us denote $s(M)$ the percentage of null coefficients in M. The expected sparsity $\mathbb{E}[s(M)] = \mathbb{P}(M_{ij} = 0)$ can be deduced from that of U and V $s(U) = s(V)$. Hopefully, both are related by the following simple formula:

Lemma 1. *Let $M = U\mathrm{Diag}(\sigma_1, ..., \sigma_r, 0, ..., 0)V^T$ with U and V defined as above, except for the rank constraint. We have*

$$\mathbb{E}[s(M)] = (1 - (1 - \mathbb{E}[s(U)])^2)^r$$

or, equivalently, $\mathbb{E}[s(U)] = 1 - \sqrt{1 - \sqrt[r]{\mathbb{E}[s(M)]}}$. Furthermore:

$$\mathrm{Var}(M_{ij} \mid M_{ij} \neq 0) = r\frac{1 - \sqrt[r]{\mathbb{E}[s(M)]}}{1 - \mathbb{E}[s(M)]}.$$

Proof. Let $M = U\mathrm{Diag}(\sigma_1, ..., \sigma_r, 0, ..., 0)V^T$. Every coefficient in U is defined as follows: we choose n^2 random coefficients $B_{ij} \overset{\text{i.i.d}}{\sim} \mathcal{B}(1 - s_U)$, and we set $U_{ij} = N_{ij}B_{ij}$, with $N_{ij} \overset{\text{i.i.d}}{\sim} \mathcal{N}(0, 1)$.

The same goes for matrix V with matrices B' and N'. We have, for any indices i, j:

$$\begin{aligned}
\mathbb{E}[s(M)] = \mathbb{P}(M_{ij} = 0) &= \mathbb{P}\left(\sum_{k=1}^{r} \sigma_k U_{ik} V_{jk} = 0\right) \\
&= \mathbb{P}\left(\sum_{k=1}^{r} \sigma_k B_{ik} B'_{jk} N_{ik} N'_{jk} = 0\right) \\
&= \mathbb{E}\left[\mathbb{P}\left(\sum_{k=1}^{r} \sigma_k B_{ik} B'_{jk} N_{ik} N'_{jk} = 0 \;\middle|\; B, B'\right)\right] \\
&= \mathbb{E}\left[\prod_{k=1}^{r} (1 - B_{ik} B'_{jk})\right] \\
&= (1 - \mathbb{P}(B_{ik} B'_{jk} = 1))^r = (1 - (1 - \mathbb{E}[s(U)])^2)^r
\end{aligned}$$

In order to derive the variance of non-zero coefficients, we use the law of total variance:

$$\begin{aligned}
& \mathrm{Var}(M_{ij}) = \mathbb{E}[\mathrm{Var}(M_{ij} \mid \mathbb{1}_{M_{ij}=0})] + \mathrm{Var}(\mathbb{E}[M_{ij} \mid \mathbb{1}_{M_{ij}=0}]) \\
\iff & \mathrm{Var}\left(\sum_{i=1}^{r} \sigma_k B_{ik} B'_{jk} U_{ik} V_{jk}\right) = \mathbb{P}(M_{ij} = 0)\mathrm{Var}(M_{ij} \mid M_{ij} = 0) \\
& \qquad + \mathbb{P}(M_{ij} \neq 0)\mathrm{Var}(M_{ij} \mid M_{ij} \neq 0) + \mathrm{Var}(\mathbb{E}[M_{ij} \mid \mathbb{1}_{M_{ij}=0}])
\end{aligned}$$

We also have $\mathrm{Var}\left(\sum_{i=1}^{r} \sigma_k B_{ik} B'_{jk} U_{ik} V_{jk}\right) = r(1 - \mathbb{E}[s(U)])^2$, $\mathrm{Var}(M_{ij} \mid M_{ij} = 0) = 0$ and $\mathbb{E}[M_{ij} \mid M_{ij} = 0] = \mathbb{E}[M_{ij} \mid M_{ij} \neq 0] = 0$, so we get:

$$\begin{aligned}
& r(1 - \mathbb{E}[s(U)])^2 = \mathbb{P}(M_{ij} \neq 0)\mathrm{Var}(M_{ij} \mid M_{ij} \neq 0) \\
\iff & \mathrm{Var}(M_{ij} \mid M_{ij} \neq 0) = r\frac{(1 - \mathbb{E}[s(U)])^2}{1 - \mathbb{E}[s(M)]} = r\frac{1 - \mathbb{E}[s(M)]^{1/r}}{1 - \mathbb{E}[s(M)]}
\end{aligned}$$

using the previous result on $\mathbb{E}[s(M)]$.

Remark. The additional constraint we add on $\mathrm{rk}(M)$ changes the expected proportion of null coefficient in U and V, so in theory the matrix M has a slightly different sparsity level than in lemma 1. However, in practice $s(M)$ and $(1 - (1 - s(U))^2)^r$ remains very close for a wide range of values of $s(U)$.

□

In the second experiment, we need random sparse low rank matrices with non-negative symmetric coefficients. We generate them using the same procedure, replacing N_{ij} by $|N_{ij}|$ and taking $U = V$. This distribution does not cover the whole set of random sparse positive symmetric matrices, but this problem seems even more difficult than the previous one: there is no orthogonal matrix with non-negative coefficients other than permutation matrices. The expected sparsity changes as U and V are no longer independent, however the difference is numerically negligible compared to the standard deviation of $s(M)$.

4.C Computing features for weighted networks

Let A be the adjacency matrix of an undirected weighted graph. In the experiments on simulated sparse low rank matrices, we compute the following graph features:

- The **weighted average degree** is defined as $\frac{1}{m^2} \sum_{i,j=1}^{m} A_{ij}$.

- The **average shortest path length** is computed using the NetworkX Python package [Hagberg et al., 2008] which implements the Dijkstra algorithm. When running the experiments, we faced a recurrent problem: randomly generated sparse low rank adjacency matrices are often not connected, which yields an infinite average shortest path. In order to circumvent this issue in the numerical computation of the shortest path, we attributed to each edge (i, j) a cost $1/(A_{i,j} + 0.1)$. This attributes a cost equal to 10 to non-existing connections, which is twice as big as the largest coefficients in the simulated data.

- The **clustering coefficient** -originally defined for binary graphs- measures the tendency of nodes to cluster together. In this study we compute a weighted version of this coefficient proposed by Opsahl and Panzarasa [2009]. A triplet of nodes is a set $\tau = \{i, j, k\}$ such that two or three edges exist between those nodes. Each triplet of nodes is represented by the geometric mean of its coefficients, denoted by $\omega(\tau)$. If the triplet forms a triangle (closed triplet), the three coefficients are used. If one edge is missing, only the two existing edges are used. The formula for the weighted clustering coefficient writes:

$$C_{\text{weighted}} = \frac{\sum_{\tau_\Delta \in \text{closed triplets}} \omega(\tau_\Delta)}{\sum_{\tau \in \text{all triplets}} \omega(\tau)}$$

Chapter 5

A Spectral Model for Populations of Networks

In this chapter, we study the variability within groups of networks, i.e., the structure of connection similarities and differences across a set of networks. We propose a statistical framework to model these variations based on manifold-valued latent factors. Each network adjacency matrix is decomposed as a weighted sum of matrix patterns with rank one. Each pattern is described as a random perturbation of a dictionary element. We apply our model on a large data set of functional brain connectivity matrices from the UK Biobank. Our results suggest that the proposed model accurately describes the complex variability in the data set with few degrees of freedom.

Contents

5.1 Introduction

Network science is at the core of an ever-growing range of applications. Network analysis [Newman, 2012] aims at studying the natural properties of complex systems of interacting components or individuals through their connections. It provides many tools to detect communities [Ni et al., 2019], predict unknown connections [Martínez et al., 2016] and covariates [Shen et al., 2017], measure population characteristics [Banks and Carley, 1994, Rubinov and Sporns, 2010] or build unsupervised low-dimensional representations [Simonovsky and Komodakis, 2018]. The need to understand and model networks arises in multiple fields, such as social networks analysis [Pozzi et al., 2016], recommender systems [Monti et al., 2017], gene interactions networks [Narayanan and Subramaniam, 2013], neuroscience [He and Evans, 2010] or chemistry [Duvenaud et al., 2015]. Network analysis allows accounting for very diverse phenomenons in similar mathematical frameworks, which lend themselves to theoretical and statistical analysis [Lovász, 2012]. In this chapter, we are interested in groups of undirected networks that are defined on the same set of nodes. This situation describes the longitudinal evolution of a given network throughout time or the case where the nodes define a standard structure identical across the networks. The former is of interest in computational social science, which studies the evolution of interactions within a fixed population [Hanneke et al., 2010]. The latter arises naturally in neuroscience, where the connections between well-defined brain regions are studied on large groups of subjects. The analysis of brain networks is the main application of the present study. It has proven an efficient tool to discover new aspects of the anatomy and function of the human brain [Fornito et al., 2016] and remains a very active research topic [Zheng et al., 2019b].

In this study, we are interested in the variability of undirected graph data sets, i.e., how graphs defined on a common set of nodes vary from one network to another. Accounting for this variability is a crucial issue in neuroscience: predicting neurodegenerative diseases or understanding the complex mechanisms of aging requires robust, coherent statistical frameworks that model the diversity among a population. Working on such graphs sharing the same nodes allows comparing them to one another through their adjacency matrices.

The comparison and statistical modeling of such matrices are difficult problems. If all the graphs have n nodes, a Gaussian model on the $n \times n$ adjacency matrices has a covariance matrix with n^4 coefficients, which is hard to interpret and difficult to estimate from a reasonable number of observations. Considering adjacency matrices as large vectors allows using classical statistical methods, such as Principal Component Analysis (PCA), but does not take advantage of the strong structures underlying the interactions between the nodes. Tailored kernel methods can be employed to evaluate distances between networks, but many theoretically interesting graph kernels require solving NP-hard problems [Ghosh et al., 2018]. In the field of brain network analysis, graphs are often modeled and summarized by features like the average shortest path length, which only partially characterize their structure [Rubinov and Sporns, 2010]. Recent methods relying on graphs neural networks often consider the nodes of the network to be permutation invariant, whereas nodes in brain networks play a specific role likely to remain stable across subjects [Fornito et al., 2016, Damoiseaux, 2017].

In this chapter, we propose a generative statistical model to express the variability in undirected graph data sets. We decompose the network adjacency matrices as a weighted sum of orthonormal matrix patterns with rank one. The patterns and their weights vary around their mean values. Using rank-one patterns allows understanding each decomposition term, while using only few parameters. This is comparable to PCA where each observation is decomposed onto orthogonal elements, which in this case would be matrices. The orthogonal patterns are seen as elements of the Stiefel manifold of rectangular matrices X such that $X^\top X$ is the identity matrix [Chikuse, 2003c]. This model allows us to use known distributions and perform a statistical estimation of the mean patterns and weights. We use a restricted number of patterns to get a robust model, which captures the main structures and their variations. This low-dimensional parametric representation provides a simple interpretation of the structure and the variability of the distribution. Our model accounts for two sources of variability: the perturbations of the patterns and their weight. In contrast, current approaches in the literature only consider one of them, as with dictionary-based models and graph auto-encoders.

The proposed framework is expressed as a generative statistical model so that it can easily be

generalized to analyze heterogeneous populations. This corresponds to a mixture of several copies of the former model where each cluster has its own center and variance parameters.

In Section 5.2, we recall relevant literature references for network modeling and statistics on the Stiefel manifold. Section 5.3 defines our model and further motivates its structure. Section 5.4 proposes an algorithm based on Expectation-Maximization (EM) to perform Maximum Likelihood Estimation of the model parameters. In Section 5.5, we present numerical experiments on synthetic and real data. We use our model to predict missing links using the parameters given by the algorithm. We show how our model can be used to perform clustering on network data sets, allowing to distinguish different modes of variability better than a classical clustering algorithm. Applying our method to the UK Biobank collection of brain functional connectivity networks, we demonstrate that our model is able to capture a complex variability with a limited number of parameters. Note that the tools we present here could also be used on any type of network, such as the ones we mentioned above or gene interaction networks.

5.2 Background

5.2.1 Statistical Modeling for Graphs Data Sets

The analysis of graph data sets is a wide area of research that overlaps with many application domains. In this section, we review the principal trends of this field that are used in statistics and machine learning.

The first category of statistical models characterizes graphs in a data set (with possibly varying number of nodes) by a set of features that can be compared across networks, rather than matching the nodes of one graph to those of another. These features can be, for example, the average shortest path length, the clustering coefficient, or the occurrence number of certain patterns. Two examples of such models are Exponential Random Graphs Models [Harris, 2014] and graph kernel methods [Ghosh et al., 2018]. Other models are defined by a simple, interpretable generative procedure that allows testing hypotheses on complex networks. The Erdős-Rényi model [Erdős and Rényi, 1959] assumes that each node has an equal probability of connecting with one another. The Stochastic Block Model (SBM, [Peixoto, 2020]) extends this model and introduces communities organized in distinct clusters with simple interactions. In the limit of a large number of nodes, the same idea gives rise to the graphon model, which has also recently been used to model graph data sets [Chandna and Maugis, 2020]. Finally, recent machine learning models leverage the power of graph neural networks [Zhang et al., 2020b] to perform classification or regression tasks. They are used, for instance, in brain network analysis to predict whether a patient is affected by Alzheimer's disease or how the disease will evolve [Banka and Rekik, 2019, Ma et al., 2020].

In this chapter, we consider undirected graphs on a fixed given set of n nodes connected by weighted or binary edges. This situation arises when studying the evolution of a given network across time [Westveld and Hoff, 2011] or when considering several subjects whose networks have the same structure, for instance, brain networks and protein or gene interaction networks. This constraint allows building models based on the ideas of mean and covariance of adjacency matrices, otherwise ill-defined when the nodes change across networks. In particular, little work has been done in the literature so far on the analysis of the variability of graphs in a data set sharing a common set of nodes. Dictionary-based graph analysis models [D'Souza et al., 2019b] and graph auto-encoders [Banka and Rekik, 2019, Liu et al., 2019] are interesting frameworks in that regard. They allow concisely representing a network in a form that compresses the $O(n^2)$ adjacency matrix representation into a smaller space of dimension $O(p)$ or $O(np)$ (where p is the encoding dimension that characterizes the model). However, they each focus on one aspect of the variability of graph data sets, either the variations of patterns for graph auto-encoders or the variations of patterns weights for dictionary-based models. The model proposed in Section 5.3 builds on these ideas and accounts for both sources of variability in two latent variables that are combined to obtain the adjacency matrices. These variables are the dominant eigenvalues and the related eigenvectors.

These eigenvectors are regrouped in matrices with orthonormal columns, which makes them points on the Stiefel manifold introduced in the next section. Statistical modeling of these matrices requires taking their geometry into account with manifold-valued distributions.

5.2.2 Models and Algorithms on the Stiefel Manifold

Compact Stiefel Manifolds of Orthonormal Frames

In this chapter, we will be considering latent variables taking values in the compact Stiefel manifold $\mathcal{V}_{n,p}$, which is defined, for $p \leq n$, as the set of n-dimensional p-frames of orthonormal vectors: $\mathcal{V}_{n,p} = \{X \in \mathbb{R}^{n\times p} \mid X^\top X = I_p\}$. Since an element of $\mathcal{V}_{n,p}$ can be obtained by taking the p first columns of an orthogonal matrix, the Stiefel manifold can be seen as a quotient manifold from the orthogonal group, and thus inherits a canonical Riemannian manifold structure. A detailed and clear introduction to algorithms for optimization and geodesic path computation on the Stiefel manifold can be found in [Edelman et al., 1998]. More recently, Zimmermann [2017] proposed an algorithm to compute the Riemannian logarithm associated with the canonical metric, solving the inverse problem of the geodesic computation.

Von Mises-Fisher Distributions

Various difficulties arise when dealing with statistical distributions on Riemannian manifolds: for instance, computing the barycenter of a set of points can be a difficult problem, if not even ill-posed. The normalizing constant of a distribution is often impossible to compute analytically from its non-normalized density, so Maximum Likelihood Estimation cannot be performed by standard optimization.

Luckily, tractable distributions on the Stiefel manifolds circumventing some of these problems have been brought up and studied over the last decades in the research field of directional statistics. The most well-studied of them is the von Mises-Fisher (vMF) distribution (also called the Matrix Langevin distribution in some papers) first introduced in [Khatri and Mardia, 1977], which is the one we will be using in this chapter. Given a matrix-valued parameter $F \in \mathbb{R}^{n\times p}$, the von Mises-Fisher distribution on the Stiefel manifold is defined by its density: $p_{\text{vMF}}(X) \propto \exp(\text{Tr}(F^\top X))$. Written differently, if we denote by $f_1, ..., f_p$ the columns of F and by $x_1, ..., x_p$ those of X, we have

$$p_{\text{vMF}}(X) \propto \exp(\langle f_1, x_1\rangle + ... + \langle f_p, x_p\rangle)\,.$$

In this expression, each x_i is drawn toward $f_i/|f_i|$ (up to the orthogonality constraint). The norm $|f_i|$ can be interpreted as a concentration parameter that determines the strength of the attraction toward $f_i/|f_i|$. The von Mises-Fisher distribution can be considered analogous to a Euclidean Gaussian distribution with a diagonal covariance matrix: the density imposes no interaction between the components of X, so that the only dependency between the columns is the orthogonality constraint. The equivalent of the Gaussian mode (which is the same as the Gaussian mean) is given by the following lemma:

Lemma 2. *The von Mises-Fisher distribution with parameter F reaches its maximum density value at $X = \pi_V(F)$, where π_V is an orthogonal projection onto the Stiefel manifold.*

Proof. From the definition of the von Mises-Fisher density, we have:

$$\begin{aligned}\text{argmax}_{X^\top X=I_p}\, \text{Tr}(F^\top X) &= \text{argmax}_{X^\top X=I_p} -\frac{1}{2}\text{Tr}(F^\top F) + \text{Tr}(F^\top X) - \frac{1}{2}\text{Tr}(X^\top X)\\ &= \text{argmin}_{X^\top X=I_p}\, \frac{1}{2}\left\|F - X\right\|^2\,,\end{aligned}$$

with $\|\cdot\|$ the Frobenius norm. Hence, by definition, $\pi_V(F)$ maximizes the von Mises-Fisher density. Note that the projection onto the Stiefel manifold is not uniquely defined, as $\mathcal{V}_{n,p}$ is not convex. □

The following lemma allows us to compute such a projection.

Lemma 3. *Let $M \in \mathbb{R}^{n\times p}$, and $M = UDV^\top$ ($U \in \mathbb{R}^{n\times p}, D \in \mathbb{R}^{p\times p}, V \in \mathbb{R}^{p\times p}$) the Singular Value Decomposition of M. If M has full rank, then $UV^\top$ is the unique projection of M onto the Stiefel manifold $\mathcal{V}_{n,p}$.*

Proof. Let us consider the Lagrangian related to the constrained optimization problem $\pi_V(M) \in \text{argmin}_{X^\top X=I_p}\, \frac{1}{2}\left\|M - X\right\|^2$:

$$\mathcal{L}(X, \Lambda) = \frac{1}{2} \|M - X\|^2 - \mathrm{Tr}(\Lambda^\top (I_p - X^\top X)) \,.$$

Then the Karush-Kuhn-Tucker theorem [Karush, 1939] shows that, if X^* is a local extremum of $X \mapsto \frac{1}{2} \|X - M\|^2$ over $\mathcal{V}_{n,p}$, then there exists Λ^* such that $\nabla_X \mathcal{L}(X^*, \Lambda^*) = 0$. This gradient writes:

$$\begin{aligned} \nabla_X \mathcal{L}(X^*, \Lambda^*) &= X^* - M + X^*(\Lambda^* + \Lambda^{*\top}) \\ &= X^*(I + \Lambda^* + \Lambda^{*\top}) - M = 0 \,. \end{aligned}$$

Since $X \in \mathcal{V}_{n,p}$ and M has full rank, the symmetric matrix $\Omega = I + \Lambda^* + \Lambda^{*\top}$ must be invertible, so that $X^* = M\Omega^{-1}$. Hence,

$$I_p = X^{*\top} X^* = \Omega^{-1} M^\top M \Omega^{-1} \iff \Omega^2 = M^\top M = V D^2 V^\top \,.$$

The matrix square roots of $M^\top M$ are exactly given by the Ω's of the form $VRV^\top$, with

$$R = \mathrm{Diag}(\pm D_{11}, ..., \pm D_{pp}) \,.$$

We get $X^* = M\Omega^{-1} = UDR^{-1}V^\top$, which gives the following value for the objective function:

$$\|M - X^*\|^2 = \left\|U(D - DR^{-1})V^\top\right\|^2 = \left\|D - DR^{-1}\right\|^2 \,.$$

As D has a positive diagonal, this function is globally minimized by $R = D$, so that the unique projection is $X^* = UV^\top$. □

The simple, interpretable density of the von Mises-Fisher distribution comes with several important advantages. First, it allows using classical Markov Chain Monte Carlo (MCMC) methods to sample efficiently from the distribution (see Figure 5.1 for examples of distributions over $\mathcal{V}_{3,2}$). Next, the form of the density makes it a member of the exponential family, which is a key requirement to perform latent variable inference with the MCMC-Stochastic Approximation Expectation-Maximization algorithm (MCMC-SAEM, [Kuhn and Lavielle, 2004]) used in this chapter. Finally, reasonably efficient algorithms exist to perform Maximum Likelihood Estimation (MLE) of the parameter F. This point will be further developed in Section 5.4.

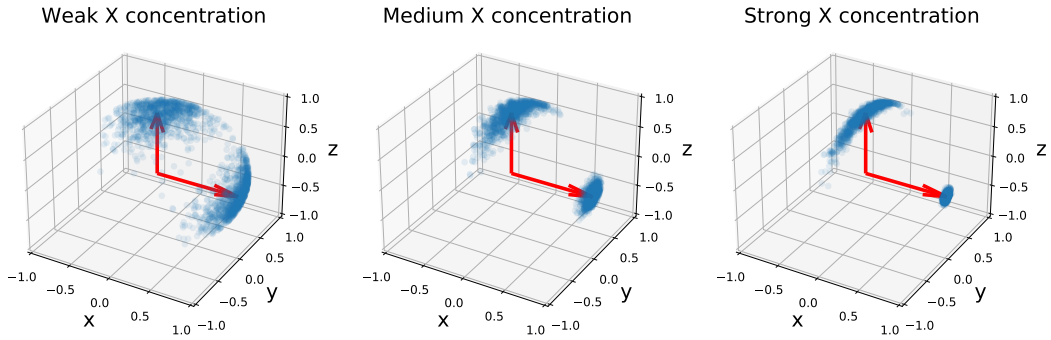

Figure 5.1: One thousand samples of three von Mises-Fisher distributions on $\mathcal{V}_{3,2}$. The mode of the distribution is represented by two red arrows along the x and z axes, and the two vectors in each matrix by two blue points. The concentration parameters are set to $|f_z| = 10$ and $|f_x| \in [10, 100, 500]$ (from left to right). Samples are drawn with an adaptive Metropolis-Hastings sampler using the transition kernel described in Section 5.4. A stronger concentration of the x vector impacts the spread of the z vector.

Application to Network Modeling

Statistical modeling on the Stiefel manifold has proven relevant to analyze networks. By considering the matrix of the p eigenvectors associated with the largest eigenvalues of an adjacency matrix as an element of $\mathcal{V}_{n,p}$, Hoff [2007a, 2009a,b] showed that probabilistic modeling of the eigenvector matrix on the Stiefel manifold provides a robust representation while allowing to quantify the uncertainty of each edge and estimate the probability of missing links. In a similar vein, the recent work of Duan et al. [2020] models data sets of spiked Laplacian matrices by exploiting their low rank structure; the authors propose a Bayesian approach to estimate the model parameters. Our model will be compared to these contributions in more detail in the final section.

5.3 A Latent Variable Model for Graph Data Sets

5.3.1 Motivation

We model graphs in a data set by studying the eigendecomposition of their adjacency matrices. Given such a symmetric weighted adjacency matrix $A \in \mathbb{R}^{n \times n}$, the spectral theorem grants the existence of a unique decomposition $A = X\Lambda X^\top = \sum_{i=1}^{r} \lambda_i x_i x_i^\top$, where r is the rank of A, and $\lambda_1 \geq ... \geq \lambda_r$ and $x_1, ..., x_r$ are the eigenvalues and the orthonormal eigenvectors of the matrix. This decomposition is unique up to the sign of the eigenvectors, as long as the non-zero eigenvalues values have multiplicity-one, which always holds in practice. The interest of this decomposition for graph adjacency matrices is threefold.

First, the eigendecomposition of the adjacency matrix reflects the modularity of a network, i.e., the extent to which its nodes can be divided into separate communities. For instance, in the case of the Stochastic Block Model (SBM), each node i is randomly assigned to one cluster $c(i)$ among p possible ones. Nodes in clusters c, c' are connected independently with probability $P_{cc'}$. In expectation, the adjacency matrix is equal to the matrix $(P_{c(i)c(j)})$, which has the rank of p at most. In samples of the SBM as well as real modular networks, the decay of the eigenvalues allows estimating the number of clusters. The eigenvectors related to non-zero eigenvalues are used to perform clustering on the nodes to retrieve their labels.

Furthermore, this decomposition provides a natural expression of A as a sum of rank-one patterns $x_i x_i^\top$. Modeling vectors as a weighted sum of patterns is at the core of dictionary learning-based and mixed effects models, which have proven of great interest to the statistics and machine learning research communities. In the specific case of graph data sets, such a model was recently proposed by D'Souza et al. [2019b] in the context of brain networks analysis. The authors learn a set of rank-one patterns without orthogonality constraints, and estimate the adjacency matrices as weighted sums of these patterns, in order to use the weights as regression variables. However, they consider the patterns as population-level variables only. This choice prevents taking into account potential individual-level variations.

Finally, the dominant eigenvectors yield strong patterns that are likely to remain stable among various networks in a data set, up to a certain variability. In other words, given N adjacency matrices $A^{(1)}, ..., A^{(N)}$ and their eigendecompositions $(X^{(1)}, \Lambda^{(1)})$, ..., $(X^{(N)}, \Lambda^{(N)})$, the first columns of the $X^{(k)}$'s should remain stable among subjects (up to a column permutation and/or change of sign). On the contrary, smaller eigenvalues should be expected to correspond to eigenvectors with greater variability. The recent work of Chen et al. [2020] takes stock of this remark to analyze the Laplacian matrices of brain networks (the Laplacian is a positive matrix that can be computed from the adjacency matrix). The authors propose to compute the ℓ_1 mean of the $X^{(k)}$'s first p columns in order to get a robust average X representative of the population. As the $X^{(k)}$'s are composed of p orthonormal vectors, their average should have the same property: it ensures that the obtained matrix can be interpreted as a point that best represents the distribution. Its definition thus formulates as an optimization problem over the Stiefel manifold $\mathcal{V}_{n,p}$. The authors show that taking this geometric consideration into account leads to better results than computing a Euclidean mean.

In the next section, we introduce our statistical analysis framework. We model the perturbations of the adjacency matrix eigendecomposition to account for the variability within a network data set.

5.3.2 Model Description

We propose to account for the variability in a set of networks by considering the random perturbation of both the patterns (X variable) that compose the networks and their weight (λ variable). In this study, we consider each pattern x_i (column of X) and each weight λ_i to be independent of one another. This assumption, although a first approximation, leads to a tractable inference problem and interpretable results. Future works could consider interactions between the x_i's or the λ_i's, as well as the dependency between both.

The model decomposition of each adjacency matrix $A^{(k)}$ in a data set writes

$$A^{(k)} = X^{(k)}\mathrm{Diag}(\lambda^{(k)})X^{(k)\top} + \varepsilon^{(k)} \tag{5.1}$$

with $X^{(k)}$ a pattern matrix, $\lambda^{(k)}$ the pattern weight vector and $\varepsilon^{(k)}$ the symmetric residual noise. The $X^{(k)}$ and $\lambda^{(k)}$ are independent unobserved variables that determine the individual-level specificity of network k. We model these variables as follows:

$$\begin{cases} X^{(k)} \overset{\text{i.i.d}}{\sim} \mathrm{vMF}(F) \\ \lambda^{(k)} \overset{\text{i.i.d}}{\sim} \mathcal{N}(\mu, \sigma_\lambda^2 I_p) \\ \varepsilon^{(k)} \overset{\text{i.i.d}}{\sim} \mathcal{N}(0, \sigma_\varepsilon^2 I_{n(n+1)/2}). \end{cases} \tag{5.2}$$

The matrix $F \in \mathbb{R}^{n\times p}$ parametrizes a von Mises-Fisher distribution for the eigenvectors matrix $X^{(k)}$, and the eigenvalues $\lambda^{(k)}$ follow a Gaussian distribution with mean $\mu \in \mathbb{R}^p$ and independent components with variance σ_λ^2. We further impose that the columns of F are orthogonal: this constraint ensures that the maximum of the log-density $\langle f_1, x_1\rangle + ... + \langle f_p, x_p\rangle$ is reached at $\pi_V(F) = (f_1/|f_1|, ..., f_p/|f_p|)$. In this model, the matrix $\pi_V(F)$ is the mode of the distribution of patterns and plays a role similar to the mean of a Gaussian distribution. The mode of the full distribution of latent variables thus refers to $(\pi_V(F), \mu)$. In the particular case where F has orthogonal columns, the column norms of F correspond to its singular values. In the remainder of the chapter we call them the *concentration parameters* of the distribution. The variability of the adjacency matrices is thus fully characterized by σ_ε, σ_λ and the concentration parameters. The pattern weights $\lambda^{(k)}$ are the eigenvalues of the $X^{(k)}\mathrm{Diag}(\lambda^{(k)})X^{(k)\top}$ term, and we thus call them eigenvalues even though they are not the actual spectrum of the real adjacency matrices $A^{(k)}$. Our model is summarized in Figure 5.2.

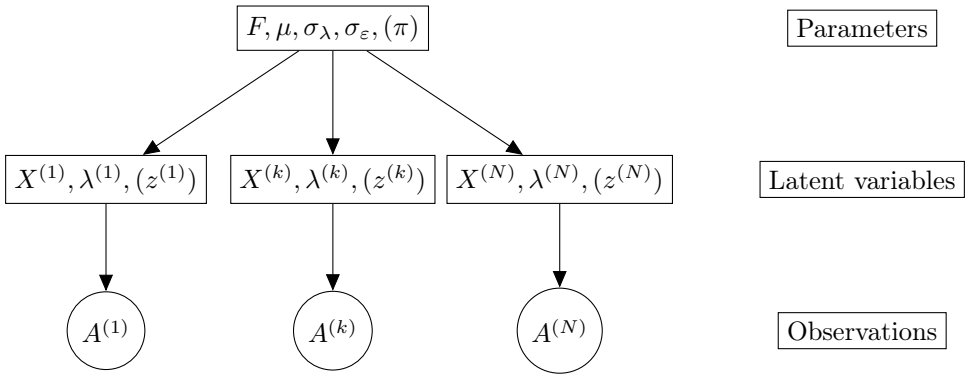

Figure 5.2: Graphical model for a data set of adjacency matrices $A_1, ..., A_N$. The variables π and $z^{(k)}$ can be added to get a mixture model.

Note that this model may be adapted to deal with other types of adjacency matrices. The distribution for $\lambda^{(k)}$ can be effortlessly changed to a log-normal distribution to model data sets of positive matrices like covariance matrices. Binary networks can be modeled by removing the $\varepsilon^{(k)}$ noise and adding a Bernoulli sampling step, considering $X^{(k)}\mathrm{Diag}(\lambda^{(k)})X^{(k)\top}$ as a logit. Adjacency matrices with positive coefficients are considered by adding the softplus function $x \mapsto \log(1 + e^x)$ in Equation (5.1). These extensions bring a wide range of possible statistical models for adjacency matrices for which the estimation procedure is the same as the one developed below.

Equation (5.1) theoretically requires each $A^{(k)}$ to be close to a rank p matrix. While this assumption is reasonable for well-clustered networks like samples of an SBM, some real-life networks exhibit heavy eigenvalue tails and cannot be approximated accurately using low rank matrices. While our model should not be expected to provide a perfect fit on general networks data sets, its main goal is to retrieve the principal modes of variability and their weight in an interpretable way, comparable to probabilistic Principal Component Analysis (PCA) or probabilistic Independent Component Analysis (ICA) [Allassonnière and Younes, 2012]. An important difference with these methods is that our model expresses each of the p components using only an n-dimensional vector, whereas PCA and ICA require an $n \times n$ matrix per component to model adjacency matrices.

In the case of well clustered networks, our model can be seen as a refinement of the SBM better suited to data sets of networks. The SBM is designed to handle one single network and mainly addresses the problem of identifying the communities. In the case of network data sets, all subjects share the same node labels and the communities can be more easily identified by averaging the edge weights over the subjects. The main assumption of the SBM that the connections between the nodes are independent of one another prevents from further analyzing individual-level variability. In contrast, our model can account for the impact of a node variation on its connections, as well as pattern variations affecting the whole network. In the limit where the concentration parameters become very large and the weight variance is small, the patterns become constant, and our model becomes equivalent to an SBM for networks organized in distinct clusters.

Another remark can be made on the identifiability of the model: the manifold of matrices of the form $X\mathrm{Diag}(\lambda)X^\top$ with $X \in \mathcal{V}_{n,p}, \lambda \in \mathbb{R}^p$ (also known as the non-compact Stiefel manifold) has a tangent space T with dimension $\dim(\mathcal{V}_{n,p}) + p = np - p(p-1)/2$ at $X^{(k)}\mathrm{Diag}(\lambda^{(k)})X^{(k)\top}$. The noise $\varepsilon^{(k)}$ can be decomposed into components in T and its orthogonal complement $T^\top$ with dimension $n^2 - np + p(p-1)/2$. The component in T thus induces an implicit source of variability on X and λ, which depends on σ_ε. We show in the experiment section that it may lead to underestimating the concentration parameters $(|f_1|, ..., |f_p|)$. While aware of this phenomenon, we consider it an acceptable trade-off regarding the simple formulation of Equation (5.2). This identifiability problem is a rather practical concern, as we will show in the next chapter that the model parameters can be formally identified.

5.3.3 Mixture Model

The matrix distribution introduced in the previous section can be integrated in a mixture model to account for heterogeneous populations with a multi-modal distribution. It amounts to considering K clusters with, for each cluster, a probability π^c and a parameter $\theta^c = (F^c, \mu^c, \sigma_\varepsilon^c, \sigma_\lambda^c)$. The mixture model writes hierarchically:

$$\begin{cases} z^{(k)} \sim \text{Categorical}(\pi) \\ (X^{(k)} \mid z^{(k)} = c) \sim \text{vMF}(F^c) \\ (\lambda^{(k)} \mid z^{(k)} = c) \sim \mathcal{N}(\mu^c, (\sigma_\lambda^c)^2 I_p) \\ (A^{(k)} \mid X^{(k)}, \lambda^{(k)}, z^{(k)} = c) \sim \mathcal{N}(X^{(k)}\mathrm{Diag}(\lambda^{(k)})X^{(k)\top}, (\sigma_\varepsilon^c)^2 I_{n(n+1)/2}). \end{cases} \tag{5.3}$$

We show in the next section on parameter estimation that the mixture layer only comes at a small algorithmic cost.

5.4 A Maximum Likelihood Estimation Algorithm

We now turn to the problem of estimating the model parameters $\theta = (F, \mu, \sigma_\lambda, \sigma_\varepsilon)$ given a set of observations $(A^{(k)})_{k=1}^N$. Let us denote $\lambda \cdot X = X\mathrm{Diag}(\lambda)X^\top$. The complete likelihood is expressed as:

$$p((A^{(k)}), (X^{(k)}), (\lambda^{(k)}); \theta) = \prod_{k=1}^{N} p(A^{(k)} \mid X^{(k)}, \lambda^{(k)}; \theta) p(X^{(k)}; \theta) p(\lambda^{(k)}; \theta)$$

with

$$\begin{cases} p(A^{(k)} \mid X^{(k)}, \lambda^{(k)}; \theta) = \frac{1}{|\sigma_\varepsilon|^{n^2}(2\pi)^{n^2/2}} \exp\left[-\frac{1}{2\sigma_\varepsilon^2}\|A^{(k)} - \lambda^{(k)} \cdot X^{(k)}\|^2\right] \\ p(X^{(k)}; \theta) = \frac{1}{\mathcal{C}_{n,p}(F)} \exp\left[\mathrm{Tr}(F^\top X^{(k)})\right] \\ p(\lambda^{(k)}; \theta) = \frac{1}{|\sigma_\lambda|^p (2\pi)^{p/2}} \exp\left[-\frac{1}{2\sigma_\lambda^2}\|\lambda^{(k)} - \mu\|^2\right] \end{cases}$$

We compute the maximum of the observed likelihood $p((A^{(k)}); \theta)$ using the MCMC-SAEM algorithm introduced in the next section. The MLE is not unique, as a permutation or a change of sign in the columns of X (together with a permutation of λ) yield the same model. This invariance can be broken by sorting the eigenvalues μ in increasing order as long as they are sufficiently spread. However, in practice, several eigenvalues may be close, and imposing such an order hinders the convergence of the algorithm. We thus choose to leave the optimization problem unchanged and deal with the permutation invariance by adding a supplementary step to the MCMC-SAEM algorithm.

5.4.1 Maximum Likelihood Estimation with the MCMC-SAEM Algorithm

When dealing with latent variable models, the main tool for MLE is the Expectation-Maximization (EM) algorithm [Dempster et al., 1977]. Given a parametric model $p(y, z; \theta)$ with y an observed variable and z a latent variable, the MLE amounts to maximizing $\log p(y; \theta) = \log \int p(y, z; \theta) \mathrm{d}z$, which is intractable in practice with classical optimization routines. The EM algorithm allows indirectly maximizing this objective by looping over two alternating steps:

1. *E-step*: Using the current value of the parameter θ_t, compute the expectation
$$Q_t(\theta) = \mathbb{E}_{p(z|y;\theta_t)}[\log p(y, z; \theta)]\,;$$
2. *M-step*: Find $\theta_{t+1} \in \mathrm{argmax}_\theta\, Q_t(\theta)$.

While the EM algorithm proves efficient to deal with simple models like mixtures of Gaussian distributions, it requires adaptation for the cases of more complicated models where the expectation in the $Q_t(\theta)$ function is intractable, and the distribution $p(z \mid y, \theta_n)$ cannot be explicitly sampled from to approximate the expectation.

The Markov Chain Monte Carlo-Stochastic Approximation EM algorithm (MCMC-SAEM) developed by Kuhn and Lavielle [2004] aims at overcoming these hurdles in the case of models belonging to the Curved Exponential Family. For such models, the log-density expresses as $\log p(y, z; \theta) = \langle S(y, z), \varphi(\theta)\rangle + \psi(\theta)$, where $S(y, y)$ is a sufficient statistic. The Q_t function then simply rewrites $Q_t(\theta) = \langle \mathbb{E}_{p(z|y;\theta_t)}[S(y, z)], \varphi(\theta)\rangle + \psi(\theta)$. In the MCMC-SAEM algorithm, the expectation of sufficient statistics is computed throughout iterations using Stochastic Approximation. The samples from $p(z \mid y; \theta_t)$ are drawn using a MCMC kernel $q(z \mid z_t; \theta_t)$ with invariant distribution $p(z \mid y; \theta_t)$. The procedure is recalled in Algorithm 5.4.1. Under additional assumptions on the model and the Markov kernel, the MCMC-SAEM algorithm converges toward a critical point of the initial objective $\log p(y; \theta)$ [Allassonnière et al., 2010].

In the case of the model proposed in this chapter, the MCMC-SAEM is well suited to the problem at hand as we have to deal with a latent variable model. In a setting with manifold-valued latent variables, the E-step of the SAEM algorithm becomes intractable; using the MCMC-SAEM allows overcoming this hurdle. Following the outline of Algorithm 5.4.1, we need to draw samples from $p(X^{(k)}, \lambda^{(k)} \mid A^{(k)}; \theta)$ and perform the maximization step using the stochastic approximation of sufficient statistics.

5.4.2 E-Step with Markov Chain Monte Carlo

Transition Kernel

The target density $p(X^{(k)}, \lambda^{(k)} \mid A^{(k)}; \theta)$ is known up to a normalizing constant, and it is sufficient to use MCMCs based on the Metropolis-Hastings acceptance rule [Hastings, 1970]. Using a general sampler such as Metropolis-Hastings allows keeping a unified MCMC structure, only slightly varying when changing the type of observed matrices, e.g., weighted matrices and binary matrices.

Algorithm 5.4.1: The MCMC-SAEM Algorithm

Initialize θ_0, z_0 and S_0
repeat
  Simulate $z_{t+1} \sim q(\cdot \mid z_t; \theta_t)$ using MCMC
  Update $\overline{S}_{t+1} = (1 - \alpha_t)\overline{S}_t + \alpha_t S(y, z_{t+1})$
  Find $\theta_{t+1} \in \operatorname{argmax}_\theta \langle \overline{S}_{t+1}, \varphi(\theta) \rangle + \psi(\theta)$
until *convergence*
return θ_T, $(z_t)_{t=1}^T$

The MCMC is structured as a Metropolis within Gibbs sampler alternating simulations of $X^{(k)}$ and $\lambda^{(k)}$ for each individual. Note that conditional density $p(\lambda^{(k)} \mid X^{(k)}, A^{(k)}; \theta)$ is a Gaussian distribution. However, when experimenting with the MCMC-SAEM, we find that using Metropolis-Hastings-based transitions rather than sampling directly from the true conditional distribution accelerates the Markov chain convergence. This is why we perform a Metropolis-Hastings within Gibbs sampler for both variables [Robert and Casella, 2010]. We generate proposals for λ with a symmetric Gaussian kernel with adaptive variance in order to reach a target acceptance rate. We also use a symmetric Metropolis Hastings transition for X, with the constraint that the variable stays on the Stiefel manifold. As explained in Chapter 3 (Section 4.3), a viable option is to use the curves given by the Cayley transform as in Li et al. [2020b]: Cayley curves can be considered a fast first-order approximation of the exponential map and produce symmetric transitions. Symmetric Metropolis-Hastings was already used on the Stiefel manifold by [Ouyang, 2008, Ch. 2, p. 20]; such methods allow for a proposal-agnostic MCMC, particularly convenient in cases like the model proposed in this chapter where the posterior density corresponds to no canonical distribution.

Remark. Our numerical implementation offers the possibility to use the Metropolis Adjusted Langevin Algorithm (MALA) instead of Metropolis-Hastings, as the gradient of the log-likelihood can be computed explicitly. While the experiments we have presented rely on the Metropolis-Hastings kernel, which is faster overall, we find that in some cases where the dimensions n and p grow large the MALA kernel allows accelerating the convergence.

Permutation Invariance Problem

The non-uniqueness of the MLE translates into a practical hurdle to the convergence of the MCMC: if two eigenvalues μ_i, μ_j are close, we get $(\mu_i, \mu_j) \cdot (x_i, x_j) \simeq (\mu_j, \mu_i) \cdot (x_i, x_j)$. As a consequence, the distribution $p(X^{(k)}, \lambda^{(k)} \mid A^{(k)}; \theta)$ is multi-modal in $X^{(k)}$, with a dominant mode close to $\pi_V(F)$ and other modes corresponding to column sign variations and permutations among similar eigenvalues. These modes are numerical artifacts rather than likely locations for the true value of $X^{(k)}$. Exploring them in the MCMC-SAEM hinders the global convergence: they encourage the samples to spread over the Stiefel manifold, which in turn yields a very bad estimation of F by inducing a bias toward the uniform distribution.

We address the permutation invariance problem by adding a column matching step every five SAEM iterations for the first third of the SAEM iterations. This step is a greedy algorithm that aims at finding the column permutation of a sample $X^{(k)}$ that makes it closest to $M = \pi_V(F)$. It proceeds recursively by choosing the columns m_i, x_j with the greatest absolute correlation. The steps are summarized in Algorithm 5.4.2. The greedy permutation algorithm causes the MCMC samples to stabilize around a single mode, allowing estimation of the F parameter.

5.4.3 M-Step with Saddle-Point Approximations

The maximization step of the MCMC-SAEM algorithm has a closed form expression, except for the parameter F. In this section, we recall a method to estimate F in a general setting and apply this method to get the optimal model parameters given sufficient statistics.

Algorithm 5.4.2: Greedy column matching

input $F \in \mathbb{R}^{n\times p}, X \in \mathcal{V}_{n,p}$
Compute $M = \pi_V(F), D = (\langle m_i, x_j\rangle)_{i,j=1}^p$
Let $I = J = \{1, ..., p\}$
Let $\sigma = (0, ..., 0)$ (column order), $\eta = (0, ..., 0)$ (column sign)
for $t \in [1, ..., n]$ **do**
  Find $i_t, j_t \in \text{argmax}_{i\in I, j\in J} |D_{ij}|$
  Set $\sigma(j_t) = i_t$, $\eta(i_t) = \text{sign}(D_{i_t j_t})$
  Set $I = I\backslash\{i_t\}, J = J\backslash\{j_t\}$
end
return σ, η

Maximum Likelihood Estimation of von Mises-Fisher Distributions

The main obstacle to retrieving the parameter F given samples $X_1, ..., X_N$ is the normalizing constant of the distribution: though analytically known, it is hard to compute in practice (see Pal et al. [2020] for a computation procedure when $n = 2$). Jupp and Mardia [1979] proved that the MLE exists and is unique as long as $p < n$ and $N \geq 2$, or $p = n$ and $N \geq 3$. Khatri and Mardia [1977], who first studied the properties of the MLE, showed the following result:

Theorem 1 (Khatri and Mardia [1977])**.** *Let $X_1, ..., X_N$ be N samples from a von Mises-Fisher distribution and $\overline{X} = \frac{1}{n}\sum_{i=1}^n X_i$. Let $\overline{X} = \overline{U}\overline{D}\overline{V}^\top$ be the Singular Value Decomposition (SVD) of $\overline{X}$. Then the Maximum Likelihood Estimator can be written under the form $\hat{F} = \overline{U}\text{Diag}(\hat{s})\overline{V}^\top$, with $\hat{s} \in \mathbb{R}_+^p$.*

Maximizing the log-likelihood of samples $X_1, ..., X_N$ is thus equivalent to solving the optimization problem

$$\text{argmax}_{s\in\mathbb{R}^p} \text{Tr}[\overline{V}\text{Diag}(s)\overline{U}^\top\overline{X}] - \log \mathcal{C}_{n,p}(\overline{U}\text{Diag}(s)\overline{V}^\top), \tag{5.4}$$

where $\mathcal{C}_{n,p}(F)$ is the normalizing constant of the vMF distribution.

Several methods were proposed to solve this problem: the authors of Khatri and Mardia [1977] provide approximate formulas when the singular values of F are all either very large or very small. The authors of Kume et al. [2013] propose a method to approximate the normalizing constant, which in turn yields a surrogate objective for the MLE giving satisfactory results. Finally, in Ali and Gao [2018], a different formula is proposed, which applies when the singular values are small. When experimenting with von Mises-Fisher distributions, we found that the method proposed by Kume et al. [2013] gives the most robust results for a wide range of singular values of F, even in a high-dimensional setting.

Application to the Proposed Model

Computational details for the likelihood rearrangement are deferred to Appendix 5.A. The model belongs to the curved exponential family, and its sufficient statistics are:

$$S(A, X, \lambda) = \begin{cases} S^1 = \frac{1}{N}\sum_{k=1}^N X^{(k)} \\ S^2 = \frac{1}{N}\sum_{k=1}^N \lambda^{(k)} \\ S^3 = \frac{1}{N}\sum_{k=1}^N \left\|\lambda^{(k)}\right\|^2 \\ S^4 = \frac{1}{N}\sum_{k=1}^N \left\|A^{(k)} - \lambda^{(k)} \cdot X^{(k)}\right\|^2 \end{cases}.$$

These sufficient statistics are updated using the MCMC samples $(X_t^{(k)}, \lambda_t^{(k)})_{k=1}^N$ with the stochastic approximation $\overline{S}_{t+1} = (1 - \alpha_t)\overline{S}_t + \alpha_t S(A, X_t, \lambda_t)$. The optimization problem defined

by the M-step of the SAEM algorithm gives the following results:

$$\hat{\theta}_t = \begin{cases} \hat{F} & = \hat{F}(\overline{S}_t^1) \\ \hat{\mu} & = \overline{S}_t^2 \\ \hat{\sigma}_\lambda^2 & = \frac{1}{p}\left(\|\hat{\mu}\|^2 - 2\langle \hat{\mu}, \overline{S}_t^2 \rangle + \overline{S}_t^3\right) \\ \hat{\sigma}_\varepsilon^2 & = \frac{1}{n^2}\overline{S}_t^4\,, \end{cases} \tag{5.5}$$

where $\hat{F}(\overline{S}_t^1)$ denotes the MLE of the von Mises-Fisher distribution. As explained in the section above, the method proposed by Kume et al. [2013] allows estimating the normalizing constant of general Fisher-Bingham distributions. The approximation relies on rewriting the constant to make it depend on a density that fits into the framework of Saddle-Point Approximations [Butler, 2007]. We recall the main steps of the computation procedure for this approximation in Appendix 5.A for the specific, simple case of vMF distributions.

In the definition of our model, we impose that the columns of F are orthogonal. As recalled in Section 5.2.2, the MLE for the vMF mode is $\overline{M} = \overline{U}\overline{V}^\top$, where $\overline{X} = \overline{U}\overline{D}\overline{V}^\top$ is the SVD of the empirical arithmetic mean of samples. Since the column norms correspond to the singular values when the columns are orthogonal, the MLE under this constraint can be sought under the form $\overline{M}\mathrm{Diag}(s)$. Hence, the optimization problem is used to estimate F:

$$\mathrm{argmax}_{s\in\mathbb{R}^p}\, \mathrm{Tr}[\mathrm{Diag}(s)\overline{M}^\top \overline{X}] - \log \widehat{\mathcal{C}}_{n,p}(\overline{M}\mathrm{Diag}(s))\,, \tag{5.6}$$

with $\widehat{\mathcal{C}}_{n,p}$ the approximation of the normalizing constant. We solve this optimization problem using the open source optimization library `scipy.optimize`.

The complete procedure is summarized in Algorithm 5.4.3.

5.4.4 Algorithm for the Mixture Model

The mixture model adds a cluster label $z^{(k)}$ for each subject and a list π of cluster probabilities. The model still remains in the curved exponential family, and the MCMC-SAEM algorithm can still be used. The Gibbs sampler now also updates $z^{(k)}$: it consists of sampling from the probabilities $p(z^{(k)} \mid X^{(k)}, \lambda^{(k)}, A^{(k)}; \pi, \theta)$, which can be computed explicitly. The sufficient statistics S^1, S^2, S^3, S^4 are defined and stored for each cluster. The statistics of cluster c are updated using only the indices k such that $z^{(k)} = c$. The variable π adds new sufficient statistics: $S^\pi = (\#\{k \mid z^{(k)} = c\}/N)_{c=1}^K$. The related MLE estimate of π is $\hat{\pi} = S^\pi$.

In our implementation, we initialize the clusters using the K-Means algorithm. We use the tempering proposed by Debavelaere et al. [2020] for the z sampling step in order to encourage points moving between clusters at the beginning of the algorithm. The vMF parameters F^c are aligned every 5 SAEM iterations using Algorithm 5.4.2 in order to allow the latent variables to move between the regions of influence of different clusters through small Metropolis-Hastings steps. The resulting algorithm is detailed in Appendix 5.C.

5.4.5 Numerical Implementation Details

We initialize the algorithm by taking the first eigenvectors and eigenvalues of each adjacency matrix. Algorithm 5.4.2 is used to align the eigenvectors between samples. In order to accelerate the convergence, we perform a small number of hybrid MCMC-SAEM steps at the start of the algorithm, where the MCMC step on X is replaced with a gradient ascent step on the log-likelihood. These first steps move the $X^{(k)}$'s to an area of $\mathcal{V}_{n,p}$ with high posterior probability, which accelerate the convergence of the MCMC, as the X variable is the slowest to evolve along the MCMC-SAEM iterations. The Riemannian gradient ascent is detailed in Appendix 5.B.

The Metropolis-Hastings transition variance is selected adaptively throughout the iterations using stochastic approximation. At SAEM step $t+1$, the proportion of accepted Metropolis transitions is computed. The logarithm of the variance is then incremented according to the rule $\log \sigma_{MH}^{t+1} = \log \sigma_{MH}^t + \ell_t/2t^{0.6}$, with $\ell_t = \pm 1$ depending on whether the proportion of accepted jumps should be increased or decreased.

Algorithm 5.4.3: Maximum Likelihood Estimation algorithm for $\theta = (F, \mu, \sigma_\varepsilon, \sigma_\lambda)$

Initialize θ_0, X_0, λ_0 and S_0
for $t = 1$ *to* T **do**
  if $t \leq T/3$ *and* $(t \bmod 5) = 0$ **then**
    for $k = 1$ *to* N **do**
      Use Algorithm 5.4.2 to align $X_t^{(k)}$ with $\pi_V(F_t)$.
      Permute $\lambda_t^{(k)}$ accordingly.
    end
  end
  Set $\widetilde{X}_0^{(k)} = X_t^{(k)}$ and $\widetilde{\lambda}_0^{(k)} = \lambda_t^{(k)}$
  for $\ell = 1$ *to* n_{MCMC} **do**
    for $k = 1$ *to* N **do**
      Sample $\widetilde{X}_\ell^{(k)}$ from the Metropolis kernel $q_X(\cdot \mid \widetilde{X}_{\ell-1}^{(k)}, \widetilde{\lambda}_{\ell-1}^{(k)}; \theta_t)$
       targeting $p(X^{(k)} \mid A^{(k)}, \widetilde{\lambda}_{\ell-1}^{(k)}; \theta_t)$
      Sample $\widetilde{\lambda}_\ell^{(k)}$ from the Metropolis kernel $q_\lambda(\cdot \mid \widetilde{X}_\ell^{(k)}, \widetilde{\lambda}_{\ell-1}^{(k)}; \theta_t)$
       targeting $p(\lambda^{(k)} \mid A^{(k)}, \widetilde{X}_{\ell-1}^{(k)}; \theta_t)$
    end
  end
  Set $X_{t+1}^{(k)} = \widetilde{X}_{n_{\text{MCMC}}}^{(k)}$ and $\lambda_{t+1}^{(k)} = \widetilde{\lambda}_{n_{\text{MCMC}}}^{(k)}$
  Update the sufficient statistics $\overline{S}_{t+1} = (1 - \alpha_t)\overline{S}_t + \alpha_t S(A, X_{t+1}, \lambda_{t+1})$
  Compute μ_{t+1}, $(\sigma_\varepsilon)_{t+1}$ and $(\sigma_\lambda)_{t+1}$ using Equation (5.5).
  Compute F_{t+1} by solving problem (5.6).
end
return θ_T, $(X_t, \lambda_t)_{t=1}^T$

During the first half of the T iterations we set $\alpha_t = 1$ in order to minimize the impact of poor initializations. Then α_t decreases as $1/(t - T/2)^{0.6}$, which ensures the theoretical convergence of the algorithm.

The algorithms as well as all the experiments presented in this chapter are implemented with Python 3.8.6. The package `numba` [Lam et al., 2015] is used to accelerate the code. We provide a complete implementation[1], which allows reproducing the experiments on synthetic data and running the algorithm on new data sets.

5.5 Experiments

5.5.1 Experiments on Synthetic Data

Parameters Estimation Performance

First we investigate the ability of the algorithm to retrieve the correct parameters when the data are simulated according to Equations (5.1) and (5.2). We test the case $(n = 3, p = 2)$, referred to as low-dimensional, where X can be visualized in three dimensions as well as the case $(n = 40, p = 20)$, referred to as high-dimensional.

Small Dimension. We choose F with two orthogonal columns uniformly in $\mathcal{V}_{3,2}$ with column norms $(25, 10)$. Using these low concentration parameters makes the results simple to visualize. We set $\mu = (20, 10)$ and $\sigma_\lambda = 2$, and generate $N = 100$ matrices $A^{(k)}$ with $\sigma_\varepsilon = 0.1$ and 100 other matrices with the same $X^{(k)}$'s and $\lambda^{(k)}$'s but a much stronger noise standard deviation $\sigma_\varepsilon = 4$. We run the MCMC-SAEM algorithm for 100 iterations with 20 MCMC steps for each maximization step. The results are shown in Figure 5.3. In both cases, the mode of the vMF distribution $\pi_V(F)$ is well recovered. In the small noise case, the posterior X samples closely match the true X

[1] `https://github.com/cmantoux/graph-spectral-variability`

samples, and the estimated concentration parameters $(23.7, 8.0)$ remain close to ground truth. In the strong noise case, the posterior samples spread much farther around $\hat{F}$ than the true samples: the estimated concentration is $(9.9, 2.8)$. This result highlights the remark in Section 5.3.2 on the bias induced by the Gaussian noise on the latent variable spread: the best X variable to estimate the matrix $A^{(k)}$ is moved apart from the true $X^{(k)}$ in a random direction because of the noise $\varepsilon^{(k)}$ living outside the manifold.

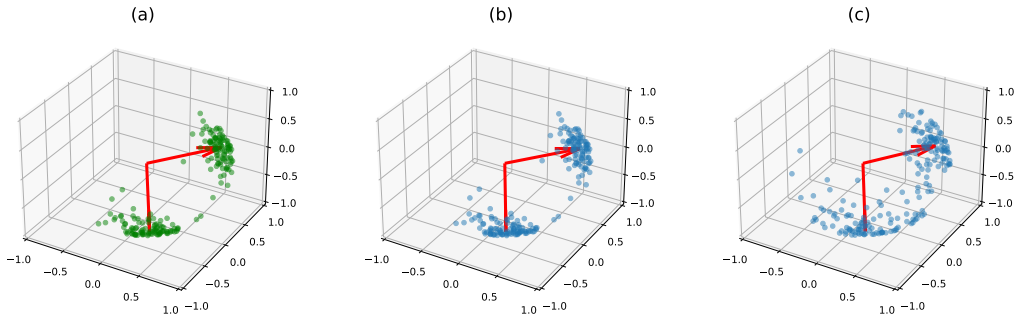

Figure 5.3: True latent variables $X^{(k)}$ and their posterior MCMC mean estimation. The red arrows represent the true $\pi_V(F)$ parameter and its estimate $\pi_V(\hat{F})$. (**a**) The true mode and samples. (**b**) Mode and samples estimates when $\sigma_\varepsilon = 0.1$. (**c**) Mode and samples estimates when $\sigma_\varepsilon = 4$. The columns are rearranged using Algorithm 5.4.2 to ease visualization. The latent variables are accurately estimated when the noise is small. A stronger noise causes the estimated latent variables to spread over the Stiefel manifold.

High Dimension. We now consider a synthetic data set of $N = 200$ samples generated from 20 latent patterns in dimension 40 and $\sigma_\varepsilon = 1, \sigma_\lambda = 2$, with various sizes of concentration parameters and eigenvalues, pairing large eigenvalues together with high concentrations. We run the MCMC-SAEM algorithm for 100 iterations with 20 MCMC steps per SAEM iteration to obtain convergence. The convergence of the parameters is shown in Figure 5.4. For both the concentration parameters and the eigenvalues, the algorithm starts by finding the highest values, only identifying lower values progressively afterward. The lowest values are associated to patterns with low weight, hence their recovery is naturally more difficult. As in the previous sections, the concentration parameters tend to be underestimated, indicating wider spreading around the mode vectors $f_i/|f_i|$ than the original latent variable. However, the ordering and orders of magnitude of the concentrations stay coherent, which, in practice, allows interpreting them and comparing them to each other. The estimation $\hat{F}$ matches the true parameter with a relative Root Mean Square Error (rRMSE) of 28%. As can be seen in Figure 5.5, the estimated normalized columns closely correspond to the original ones except when the concentration parameters get too small to allow for a good estimation, as explained above.

We use this example to illustrate the role of the algorithm hyperparameters on the practical convergence, namely the number of MCMC steps per SAEM iteration and the column matching step. We consider the same data set, but we initialize the MCMC-SAEM algorithm with random latent variables instead of the method described in Section 5.4: this worst-case initialization highlights the differences between the settings more easily. It is also closer to the case of real data sets: the MCMC and model parameters are slower to converge on real data, as the adjacency matrices are not actual samples of the theoretical model distribution. For different numbers of MCMC steps per SAEM iterations, we run the MCMC-SAEM algorithm for 200 iterations 10 times to average out the random initialization dependency, with and without the column matching step. Then we compute the relative RMSE of the parameters F and μ at the end of the algorithm. The rRMSE averaged over the 10 runs is shown in Figure 5.6. It can be seen that when the column matching step is used, increasing the number of MCMC steps at a fixed number of SAEM iterations improves the estimation. It allows accelerating the convergence, as MCMC steps are faster than the maximization step (which requires repeated vMF normalizing constant computations). However, when the number of MCMC steps gets too large, the performance improvement stagnates while

the execution time increases. We find that, in practice, using between 20 and 40 MCMC steps per SAEM iterations is a good compromise in terms of convergence speed. Figure 5.6 also illustrates the need for the column matching step proposed in Section 5.4: when not used, the parameters hardly converge to the right values, even with many MCMC steps per SAEM iteration. When the eigenvectors are permuted differently across the samples, the related eigenvalues cannot be estimated accurately, as they mix together when averaged in the maximization step. The absence of permutations also spreads the eigenvectors over the Stiefel manifold, which prevents estimating the von Mises-Fisher parameter. Since Algorithm 5.4.2 is very fast to execute, it is not a computational bottleneck. In our experiments, the number of SAEM iterations between successive column permutation steps did not have a significant impact as long as it was not too high: values between 5 and 20 produced similar results.

Model Selection. In all the experiments on simulated data presented in this chapter, we use the correct number of columns p, which we assume to be known. However, when studying real data sets, classical model selection procedures like the Bayesian Information Criterion cannot be applied to our model: they require computing the complete probability of the observations $p(A \mid \theta_m) = \int_{\mathcal{V}_{n,p}} \int_{\mathbb{R}^{p_m}} p(A \mid X, \lambda, \theta_m) \, \mathrm{d}X \, \mathrm{d}\lambda$ for each model θ_m. This probability cannot be computed explicitly, as it requires integrating over the Stiefel manifold, which results in intractable expressions using the matrix hypergeometric function [Kume et al., 2013].

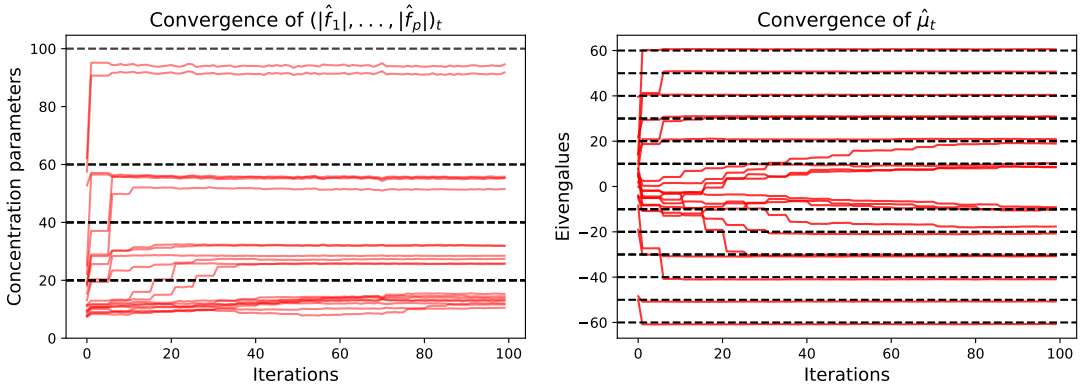

Figure 5.4: Convergence of the concentration parameters $(|f_1|, \ldots, |f_p|)$ (**left**) and the mean eigenvalues μ (**right**) over the SAEM iterations. The red lines represent the values of the parameters along the iterations. The black dotted lines represent the true values, which are grouped in batches to ease visualization. The convergence is fastest for the large eigenvalues and concentration parameters. At the start of the algorithm, the biggest changes in the parameters come from the greedy permutation performed every 5 iterations. As explained in the text, the concentration parameters are underestimated. However, they keep the right order of magnitude, which allows interpreting the output of the algorithm in practice.

In practice, several tools can be used to choose the number of latent patterns. First, the marginal likelihood $p(A \mid X, \lambda; \theta)$ or the error $\|A - \lambda \cdot X\|$ can be used to evaluate the model expressiveness. As p increases, the error will naturally diminish and should be very small for $p = n$. As with linear models, the proportion of the variance captured by $\lambda \cdot X$ can be computed to evaluate the improvement gained by adding new patterns. The concentration parameters of the von Mises-Fisher distribution also give important information on pattern relevance: if a pattern has a very low concentration parameter, it means that the related eigenvectors are widely spread across the Stiefel manifold. Smaller concentrations are thus related to overfitting, as they do not correspond to actual patterns contributing to the data set variability. The relative importance of concentration parameters can be compared numerically with the vMF concentration obtained on samples from the uniform distribution gathered with Algorithm 5.4.2.

Remark. In this chapter, we approximate the posterior mean of MCMC samples of $X^{(k)}$ by projecting their arithmetic mean over the Stiefel manifold. We find this procedure a very convenient

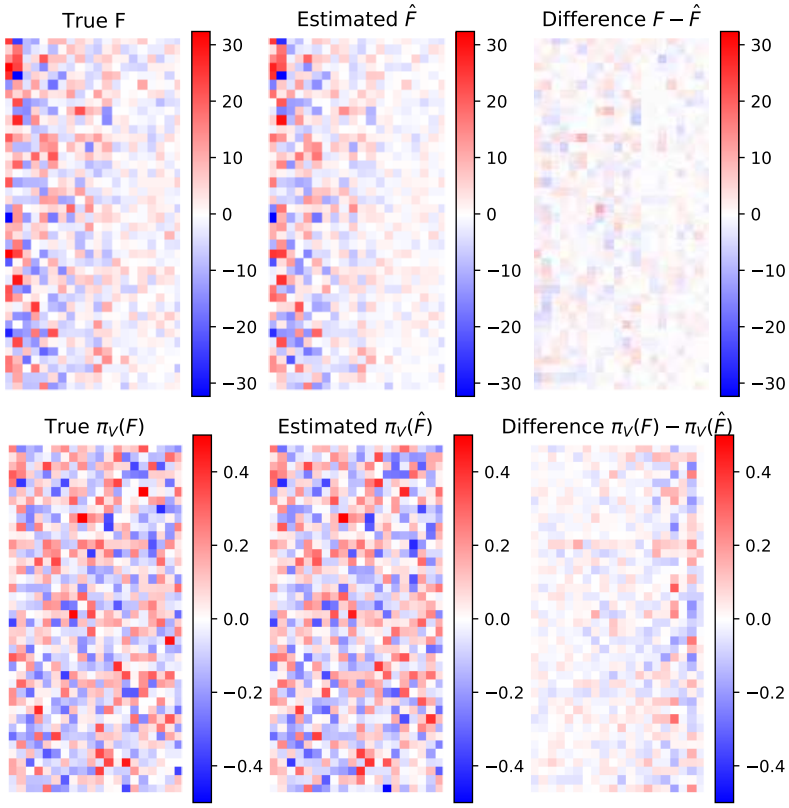

Figure 5.5: Von Mises-Stiefel distribution parameter F and its estimation $\hat{F}$. (**Top row**): the two parameters and their difference. (**Bottom row**): mode of the true distribution (given by $\pi_V(F)$), mode of the estimated distribution $\pi_V(\hat{F})$ and their difference. The images show each matrix as an array of coefficients, with pixel color corresponding to coefficient amplitude. Since the matrix columns are orthonormal, the projection just consists of normalizing the columns. The columns are sorted by decreasing the concentration parameter. The normalized columns of F corresponding to the smallest concentration parameters are estimated with less precision.

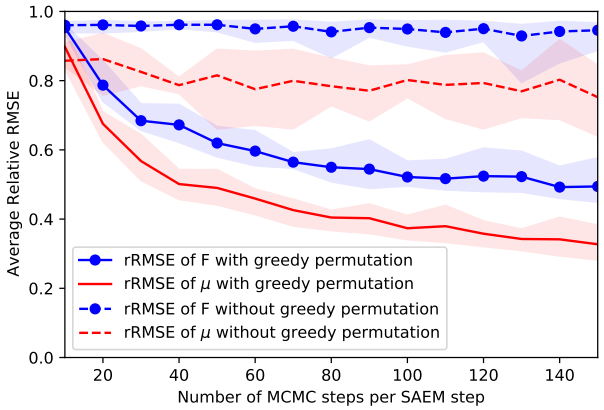

Figure 5.6: Relative RMSE of parameters F and μ after 100 MCMC-SAEM iterations depending on the number of MCMC steps per SAEM iteration. Results are averaged over 10 experiments to reduce the variance. The shaded areas indicate the extremal values across the repeated experiments. When using the greedy permutation, the rRMSE decreases rapidly when the number of MCMC steps increases before stabilizing. On the other hand, without the permutation step, the performance stays poor for any number of MCMC steps per maximization, as the parameters cannot be estimated correctly. In this experiment only, the latent variables are initialized at random to highlight the result.

alternative to computing the Fréchet mean (i.e., the Riemannian center of mass) over the manifold for two reasons. First, computing the Fréchet mean requires an extensive use of the Riemannian logarithm. Although a recent paper [Zimmermann, 2017] allows computing this logarithm, the proposed algorithm heavily relies on matrix logarithm computations and requires points to remain very close to the mean. Similar iterative algorithms to compute the mean based on other retraction and lifting maps than the Riemannian exponential and logarithm were proposed and analyzed in Kaneko et al. [2013], but in our experiments, these alternatives also turn out to require samples close to the mean point, especially in high dimensions. Second, projecting the mean sample onto the Stiefel manifold amounts to computing the mode of a vMF distribution. As shown in Appendix 5.D, the vMF distribution is symmetric around its mode, which makes this mode a summary variable similar to the Gaussian mean.

Missing Links Imputation

Once the parameters $\hat{\theta}$ are estimated from adjacency matrices $A_1, ..., A_N$, missing links can be inferred on a new adjacency matrix A. Suppose that only a subset Ω of the edge weights is known: the weights of masked edges $\overline{\Omega}$ can be obtained by considering the posterior distribution $p(A_{\overline{\Omega}} \mid A_\Omega; \theta)$. This distribution is obtained as a marginal density of the full posterior $p(A_{\overline{\Omega}}, X, \lambda \mid A_\Omega; \theta)$. Sampling from this distribution yields a posterior mean as well as confidence intervals for the value of missing links. In the case of binary networks, the posterior distribution gives the probability of a link existing for each masked edge. Samples are obtained by Gibbs sampling using the same method as in Section 5.4. We also compute the Maximum A Posteriori (MAP) by performing gradient ascent on the posterior density of $(A_{\overline{\Omega}}, X, \lambda)$ given A_Ω.

We generate a synthetic data set of $N = 200$ adjacency matrices with $n = 20$ nodes and $p = 5$. The noise level σ_ε is chosen such that the average relative difference between the coefficients of $A^{(k)}$ and $\lambda^{(k)} \cdot X^{(k)}$ is 25%. We estimate the model parameters using the MCMC-SAEM algorithm. Then, we generate another 200 samples from the same model. We mask 16% of the edge weights corresponding to the interactions between the last eight nodes. The posterior estimation is compared with the ground truth for one matrix in Figure 5.7. Both the MAP and posterior mean allow estimating the masked coefficients better than the mean sample $(A_1 + ... + A_N)/N$, which is the base reference for missing data imputation. They achieve, respectively, 58% (±28%) and 57% (±24%) rRMSE on average, whereas the mean sample has an 85% (±10% over the data set) relative difference to the samples on average. Finally, we perform the same experiment except we select the masked edges uniformly at random, masking 40% of the edges. This problem is easier than the former despite the larger amount of hidden coefficients because the missing connections are not aligned with each other. The posterior mean and the MAP achieve, respectively, 34%(±9% over the data set) and 35%(±7%) rRMSE, against 75%(±5%) for the mean sample.

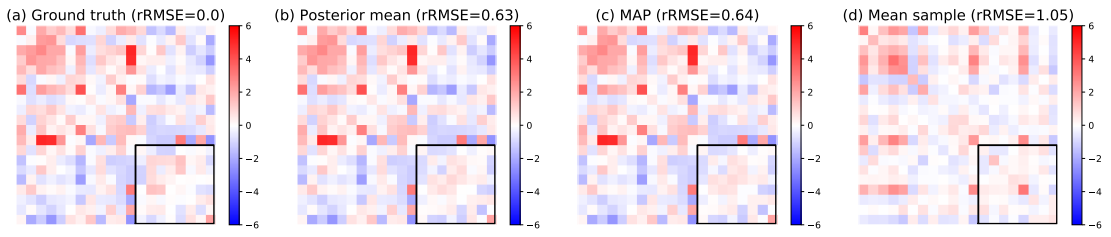

Figure 5.7: Result for missing link inference using the posterior distribution. (**a**) Ground truth input matrix A. (**b**) Posterior mean of the masked coefficients. (**c**) MAP estimator. (**d**) Mean of model samples for comparison. The area of masked edges is highlighted by a black square. Aboveeach matrix is the rRMSE with the ground truth. Both the posterior mean and the MAP give a reasonable estimation for the missing weights, significantly better than the empirical mean of all adjacency matrices, which is the base reference for missing data imputation. The images show each matrix as an array of coefficients, with pixel color corresponding to coefficient amplitude.

Link prediction has been a very active research topic in network analysis for several decades, and numerous methods can be employed to address this problem depending on the setting [Lu and

Zhou, 2011, Martínez et al., 2016, Zhang and Chen, 2018]. However, the most commonly used approaches are designed to perform inference on a single network or consider the nodes as permutation invariant. In turn, the new approach we propose allows for population-informed prediction and uncertainty quantification. It could be used in practice to compare specific connection weights of a new subject with their distribution given the other coefficients and the population parameters. This comparison provides a tool to detect anomalies in the subject's connectivity network stepping out of the standard variability.

Remark. The error uncertainties reported in this chapter refer to the variance of the estimation error across the adjacency matrices.

Clustering on Synthetic Data

As explained in Section 5.3.3, our model can be used within a mixture to account for multi-modal distributions of networks. When experimenting with the clustering version of our algorithm on data sets with distinctly separated clusters, we noticed that the algorithm provides results similar to running K-Means and estimating the parameters on each K-Means cluster separately. However, the clusters in complex populations often overlap, and the ideal case where all groups are well separated rarely occurs. In this section, we show two examples of simulated data sets where the variabilities of the clusters makes them hard to distinguish with the sole application of the K-Means algorithm.

Small Dimension. We test the mixture model estimation in the small dimensional case ($n = 3$, $p = 2$) where results can be visualized. We simulate three clusters of matrices as in Section 5.5.1 with $N = 500$ samples overall. In order to make the problem difficult, we use the same mean eigenvalues for two clusters. We set the Stiefel modes of these clusters to be very close, differing mainly by their concentration parameters. We run the tempered MCMC-SAEM for 1000 iterations with a decreasing temperature profile $T_t = 1 + 50/t^{0.6}$. Once the convergence is achieved, the estimated clusters are mapped to the true clusters. The eigenvalue parameters are estimated accurately with 2% rRMSE. The original and estimated von Mises-Fisher distributions are compared in Figure 5.8. We can see that each cluster distribution is well recovered. In particular, the overlapping distributions of cluster 1 and 2 are separated, and the higher concentration of cluster 1 is recovered in the estimation. This example also highlights the relevance of the MCMC-SAEM clustering procedure compared with its K-Means initialization: up to a label permutation, 50.4% of the K-Means proposed labels are correct, whereas the posterior distribution $p(z^{(k)} \mid A^{(k)}; \hat{\theta})$ computed with the final MCMC samples predicts the correct answer for 79.6% of the model samples.

Larger Dimension. We now test the mixture model on a synthetic data set of 500 samples in dimension ($n = 20, p = 10$). We generate four clusters with Stiefel modes close to one another, with equal concentration parameters. The modes mainly differ by their mean eigenvalues μ^c. The eigenvalue standard deviation σ_λ is set to be of the same order of magnitude as the means μ, larger than most of its coefficients. The resulting data set is hard to estimate with classical clustering: the K-Means algorithm retrieves 53.6% of correct labels at best. In contrast, running the tempered MCMC-SAEM algorithm for 1000 iterations yields 99.4% of correct labels. The algorithm achieves this result by identifying the template patterns of each cluster despite the large variation in their weights. Once these template patterns are learned, the proportion of correctly classified samples increases and the mean eigenvalues of each cluster converge to a good estimation.

Model Selection. Selecting the number of clusters K is a known problem addressed for general mixture models [Celeux et al., 2019]. Although it is well understood for simple Gaussian mixture models or for low dimensional data, other cases remain challenging problems. For the model proposed in this chapter, likelihood-based procedures cannot be applied, as the complete likelihood is an integral over the Stiefel manifold (see Section 5.5.1). As with the selection of parameter p, the concentration parameters and the reconstruction errors could be used to choose the number of clusters. Using a K that is too small will result in stretching the latent von Mises-Fisher distributions toward low concentration parameters and large reconstruction errors. The reconstruction error should decrease slower once the right number of clusters has been reached.

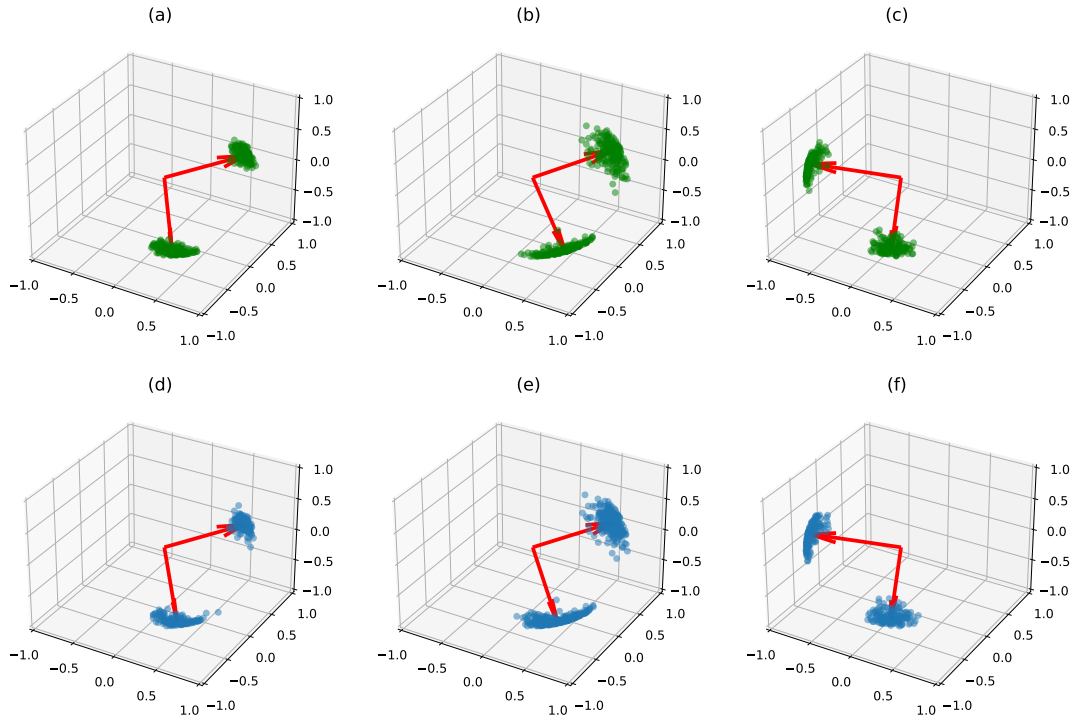

Figure 5.8: True latent variables $X^{(k)}$ and their posterior mean estimation for the clustering problem. (**Top row**): the plots (a,b,c) represent the true vMF modes (in red), as well as the true $X^{(k)}$ samples (in green) in their true class. (**Bottom row**): the plots (d,e,f) represent the three estimated vMF central modes (in red) and the estimated $X^{(k)}$ in their estimated class (in blue). The cluster centers are well recovered, as well as the concentration parameters. In particular, the two first clusters, which mainly differ by their concentration parameters, are correctly separated.

Remark. The link prediction procedure described in Section 5.5.1 could also be applied in the mixture model to infer the coefficients of new networks of which the class is unknown.

5.5.2 Experiments on Brain Connectivity Networks

We test our model on the UK Biobank data repository [Sudlow et al., 2015]. The UK Biobank is a large scale data collection project, gathering brain imaging data on more than 37,000 subjects. In this chapter, we are interested more specifically in the resting-state functional Magnetic Resonance Imaging data (rs-fMRI). The rs-fMRI measures the variations of blood oxygenation levels (BOLD signals) across the whole brain while the subject is in a resting state, i.e., receives no stimulation. The brain is then divided into regions through a spatial ICA that maximizes the signal coherence within each region [Kiviniemi et al., 2003]. Smaller regions give more detail on the brain structure but are less consistent across individuals. Finally, the raw imaging data are processed to obtain a matrix that gathers the temporal correlations between the mean blood oxygenation levels in each region. This matrix thus represents the way brain regions activate and deactivate with one another. It is called the *functional connectivity network* of the brain, as it provides information on the role of the regions rather than their physical connections. In the UK Biobank data used in the present study, the connectivity matrices are defined on a parcellation of the brain into $n = 21$ regions. These connectivity matrices illustrate our purpose well: as shown in Figure 5.9, the data set has a very large diversity of networks that express in patterns with varying weights.

Parameter Estimation

We run our algorithm on $N = 1000$ subjects for 1000 SAEM iterations with 20 MCMC steps per SAEM iteration. Working on a restricted number of samples allows for a fast convergence toward the final values. Indeed, we noticed that, while most of the parameters stabilize relatively fast, the time to convergence of the concentration parameters grows with the number of samples. Apart from these concentration parameters, we obtained very similar results when taking all the UK Biobank subjects. In this section, we consider a decomposition into $p = 5$ patterns. In Appendix 5.E.1, we show the results obtained by taking different values of p.

In Figure 5.10, we show the p normalized patterns $f_i f_i^\top / \|f_i\|^2$ obtained once the algorithm has converged. Patterns 3 and 5 have very high concentration parameters and only use a small subset of the nodes. The three other patterns have smaller concentration parameters. However, these concentrations are still high enough for the related columns of X to be significantly more concentrated than a uniform distribution: the average Euclidean distance between these three columns of $X^{(k)}$ and the related mode columns is 1.1 (± 0.2 over the data set). Comparatively, the average distance between two points drawn uniformly on the Stiefel manifold is 2.4 (± 0.2) (over 10,000 uniform samples).

Figure 5.11 displays data set matrices $A^{(k)}$ alongside the respective mean posterior estimates of $\lambda^{(k)} \cdot X^{(k)}$. For comparison purpose, we also compute the approximation given by the projection onto the subspace of the first five PCA components of the full data set, where each component has been vectorized. The $\lambda \cdot X$ matrices capture the main structure, whereas the PCA approximation relying on the same number of base components provides a less accurate reconstitution. Quantitatively, the $\lambda \cdot X$ term has a 47% ($\pm 5\%$ over the data set) relative distance to A, whereas the PCA approximation has a 92% ($\pm 12\%$) relative distance to A. The $\lambda \cdot X$ representation accounts for 60% of the total variance, whereas the corresponding PCA representation only accounts for 35%. This difference highlights the benefits of taking into account the variations of the patterns across individuals. In a classical dictionary-based representation model, the patterns do not vary among individuals. In contrast, accounting for the pattern variability only adds few parameters (one per pattern) and increases the representation power.

Pattern Interpretation

Once the patterns are identified, they can be interpreted based on the function of the related involved brain regions. All brain regions can be found on a web page of the UK Biobank project[2].

[2] `https://www.fmrib.ox.ac.uk/datasets/ukbiobank/group_means/rfMRI_ICA_d25_good_nodes.html`

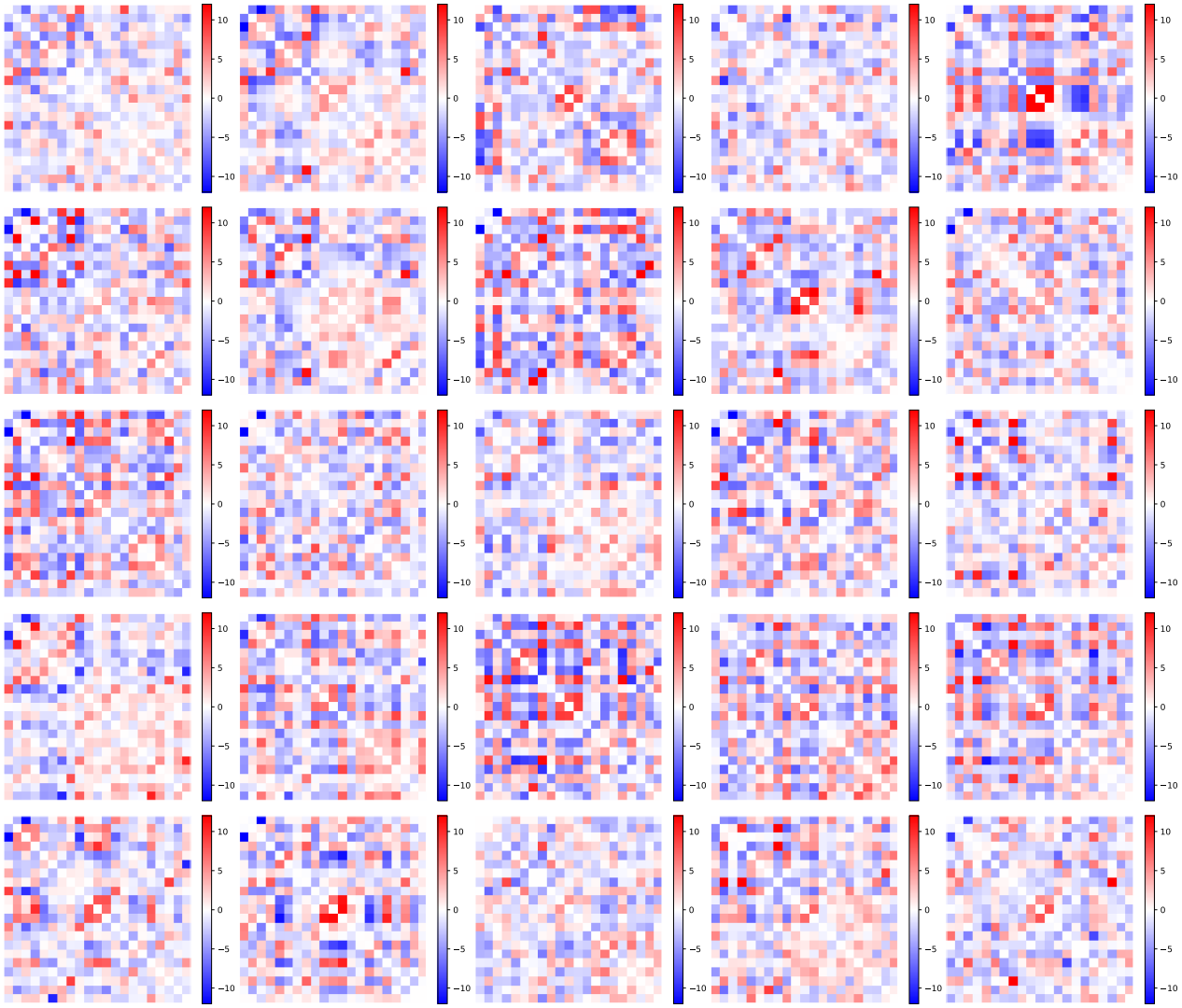

Figure 5.9: Functional connectivity matrices (21×21) of 25 UK Biobank subjects. The connectivity structure changes a lot depending on the subject, with various patterns expressing with different weights. The matrices in the data set have no diagonal coefficients; hence, the diagonals are shown as zero.

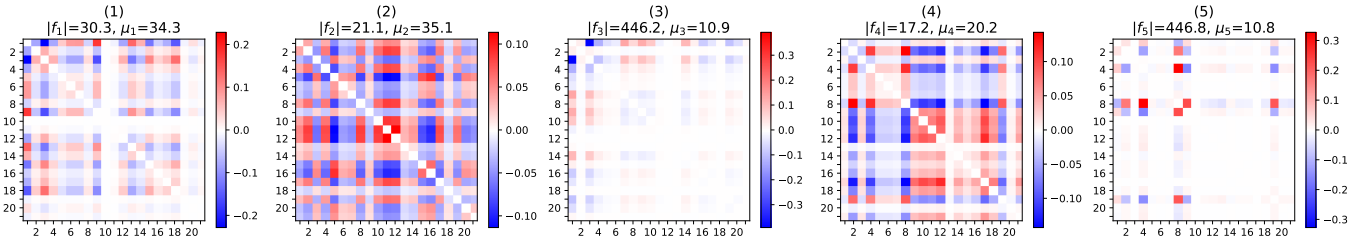

Figure 5.10: Normalized rank-one connectivity patterns. The matrix i represents $\text{sign}(\mu_i) f_i f_i^\top / \|f_i\|^2$. The caption above each pattern gives the related concentration parameter and mean eigenvalue. The diagonal coefficients are set to zero, as they do not correspond to values in the data set. The images show each matrix as an array of coefficients, with pixel color corresponding to coefficient amplitude.

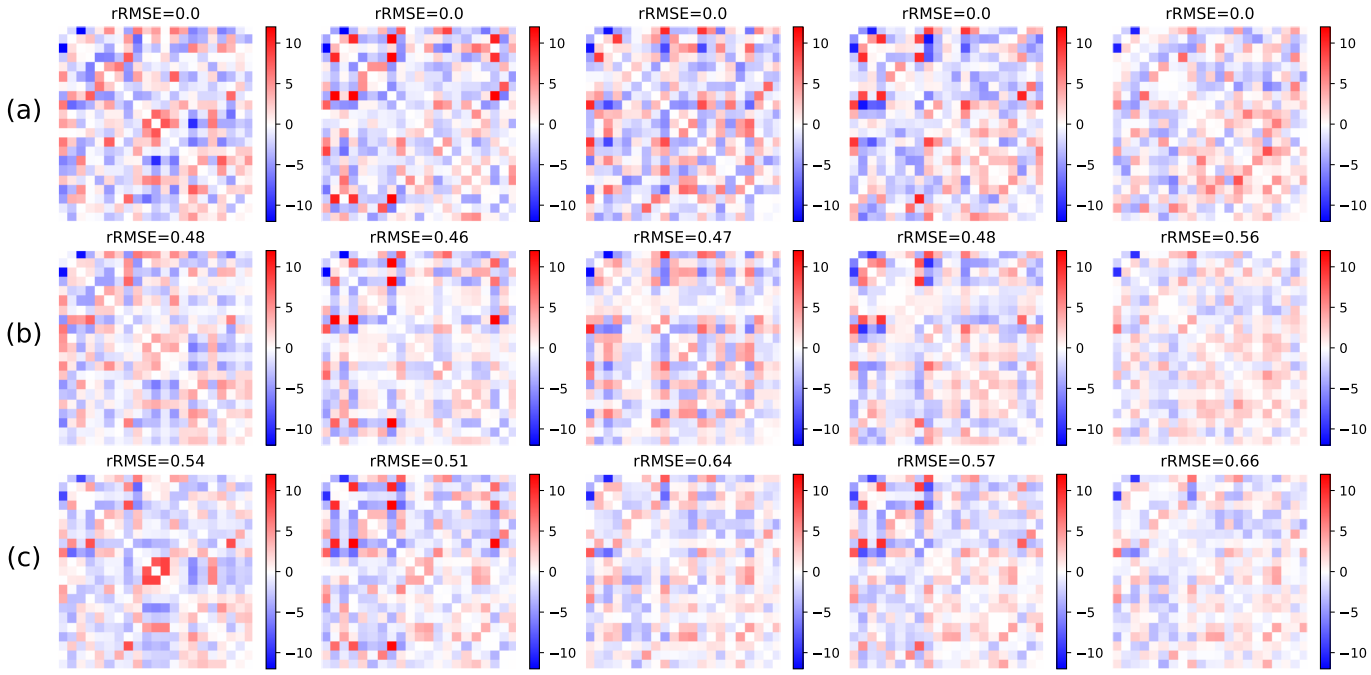

Figure 5.11: (a) UK Biobank connectivity matrices for 5 subjects. (b) Corresponding posterior mean value of $\lambda \cdot X$ estimated by the MCMC-SAEM. (c) Projection of the true connectivity matrices onto the subspace of the first five PCA components. The posterior mean matrix achieves a better rRMSE than PCA by capturing the main patterns of each individual matrix. As in Figure 5.10, the diagonal cofficients are set to zero.

The regions analyzed in this section can be visualized on brain cuts in Appendix 5.E.2.

Pattern 3 mainly represents the anti-correlation between regions 1 and 3. Region 1 comprises, among others, the inner part of the orbitofrontal cortex and the precuneus. These regions are parts of the Default Mode Network (DMN) of the brain, which is a large-scale functional brain network known to be active when the subject is at rest or mind-wandering [Horn et al., 2014]. Region 3 comprises part of the insular cortex and the post-central gyrus, which both play a role in primary sensory functions. The anti-correlation between regions 1 and 3 is a consequence of external sensations activating the sensory areas and decreasing the DMN activity. This anti-correlation is also one of the strongest coefficients in pattern 1.

Pattern 5 mainly features the dependency between nodes 2, 4, 8, 9, and 19, which are all related to the visual functions. Node 2 represents the parts of the occipital and temporal lobes forming the ventral and dorsal streams: they are theorized to process the raw sensory vision and hearing to answer the questions "what?" and "where?" [Eysenck, 2010]. Region 4 features the cuneus, which is a primary visual area in the occipital lobe. Region 8 spans over the whole occipital lobe, covering primary visual functions and associative functions like the recognition of color or movement. Region 9 comprises the continuation of the ventral and dorsal streams of region 2 in the parietal and medial temporal areas. Finally, Region 19 represents the V1 area that processes the primary visual information. Pattern 5 involving these regions has a very high concentration parameter, which means that this structure remains very stable among the subjects.

Considering that the subject's activity in the MRI scanner mainly consists of looking around and laying still, it is coherent that the most stable patterns (i.e., with the highest concentration parameters) during the resting-state fMRI measurement are the activity of the vision system and the anti-correlation between the DMN and sensory areas.

Pattern 4 also shows the interaction between the visual areas 2, 4, 8, and 19. It also includes the strong correlation between nodes 9, 10, 11, 12, and 17. Regions 10, 11, and 12 are involved in motor functions. Region 10 features part of the pre-central gyrus, which is central in the motor

control function, and part of the post-central gyrus, which is involved in sensory information processing. Region 11 encompasses the entire pre-central gyrus. Region 12 includes a part of the motor and pre-motor cortex in the frontal lobe and the insular cortex. It also includes the cerebellum, which plays an important role in motor control, and the insular cortex, which also acts on the motor control, for instance, in the face and hands motion control [Purves et al., 2017]. Region 17 comprises the medial face of the superior temporal gyrus and the hippocampus, which are involved in short and long-term memory and spatial navigation.

Pattern 2 combines, to some extent, the structure contained in patterns 4 and 5. It features, among others, interactions between the visual areas and the correlation between the motor function areas.

Remark. The results and interpretation we present in this experiment depend on the state of the subjects -in this case, a resting state- and the brain parcellation used to obtain the definition of the regions. If we were to analyze another data set of subjects performing a different task, the connectivity patterns X would likely differ from their resting-state counterpart. It follows from the fact that two different phenomenons naturally require two different base dictionaries. Analyzing the pattern difference would thus provide a way to interpret the structure difference between the two settings. For instance, the role of the occipital lobe in the vision-involved patterns would likely change for tasks related to vision. However, if the brain regions are defined differently in the two experiments, the comparison can only be made in a qualitative way.

Link Prediction

We evaluate the relevance of our model on fMRI data by testing the missing link imputation method introduced in the previous section. First we fit the model on $N = 1000$ subjects. Then we take 1000 other test subjects and mask the edges corresponding to the interactions between the last nine nodes (except the diagonal coefficients, which are unknown and thus considered null). We compute the MAP estimator of the masked coefficients. For comparison purposes, we perform a linear regression to predict the masked coefficients given the visible ones. Finally, we truncate the matrix with masked coefficients to only keep the p dominant eigenvalues. This technique is at the core of low-rank matrix completion methods [Nguyen et al., 2019], and it relates naturally with the estimation derived from our model relying on low-rank variability. The result is shown for one sample in Figure 5.12. The linear model and the MAP estimator give comparable estimates, both close to the true masked coefficients. Over the 1000 test subjects, these estimators achieve on average 58% ($\pm$14% over the samples) rRMSE for the linear model and 65%($\pm$15%) rRMSE for the MAP. Interestingly, our model uses only $np + p + 2 = 112$ degrees of freedom, whereas the linear prediction model has dimension 26,640 and was specifically trained for the regression purpose.

Our model captures a faithful representation of the fMRI data set and uses far fewer coefficients than other models like PCA and linear regression by accounting for the structure of the interactions between the network nodes. It provides an explanation of the network variability using simple interpretable patterns, which correspond to known specific functions and structures of the brain. The variations of these patterns and their weight allow for a representation rich enough to explain a significant proportion of the variance and impute the value of missing coefficients.

5.6 Conclusion

Summary. This chapter introduces a new model for the analysis of undirected graph data sets. The adjacency matrices are expressed as a weighted sum of rank-one matrix patterns. The individual-level deviations from the population average translate into variations of the patterns and their weight. Sample graphs are characterized by these variations in a way similar to PCA. The form of the decomposition allows for a simple interpretation: each pattern corresponds to a matrix with rank one and is thus represented by a vector of node coefficients. The variability of this decomposition is captured within a small number of variance and concentration parameters.

We use the MCMC-SAEM algorithm to estimate the model parameters as well as the individual-level variable. The parameter of von Mises-Fisher distributions is recovered by estimating the vMF

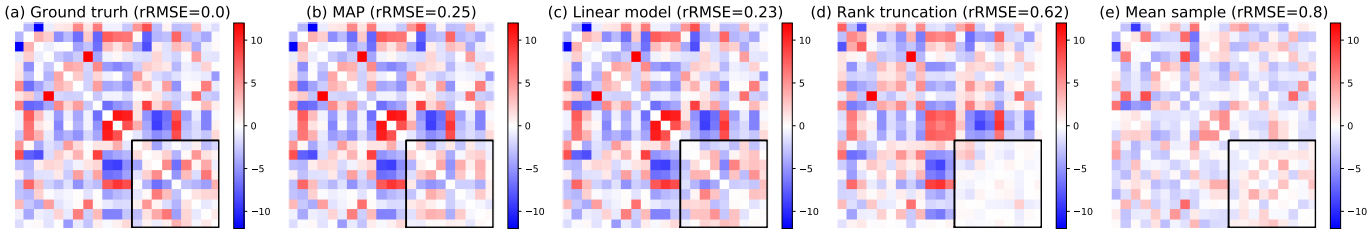

Figure 5.12: From left to right: (**a**) True connectivity matrix A. (**b**) MAP estimator for the masked coefficients framed in a black square. (**c**) Linear model prediction for the masked coefficients. (**d**) Rank 5 truncation of the matrix A with masked coefficients set to zero. (**e**) Mean of all data set matrices. Above each matrix is the rRMSE with the ground truth.

normalizing constant, which allows retrieving both the mode and its concentration parameters. Future work could further investigate the role of the approximation error induced by the use of saddle-point approximations, comparing its performance with a recently proposed alternative method [Kume and Sei, 2018]. The impact of noise on the underestimation of the vMF distribution concentration also requires further analysis.

Experiments on synthetic data show that the algorithm yields good approximations of the true parameters and covers the posterior distributions of the latent variables. Our model can be used to infer the value of masked or unknown edge weights once the parameters are estimated. In practice, the posterior distribution could be compared to the real connections to detect anomalous connections that step out of the expected individual variability.

The model we introduce is a hierarchical generative statistical model, which easily extends to mixture models. We show that a mixture of decomposition models can be estimated with a similar algorithmic procedure and allow disentangling between modes of variability that are indistinguishable by a traditional clustering method.

We demonstrate the relevance of the proposed approach for the modeling of functional brain networks. Using few parameters, it explains the main components of the variability. The induced posterior representation is more accurate than PCA and gives a link prediction performance similar to a linear model, which has a comparably simple structure, but requires far more coefficients and was trained specifically to that purpose. The estimated connectivity patterns have a simple structure and lead to an interpretable representation of the functional networks. We show that our model identifies specific patterns for the visual information processing system or the motor control. The related concentration parameters allow measuring the variability of the function of the related brain regions across the subjects.

Comparison with other eigenmodels. Our model closely relates to the approach proposed by Hoff [2009a]: in this chapter, the author models data sets of covariance matrices. The eigenvectors are drawn from Fisher-Bingham distributions of a specific form, and the individual level deviation is added under the form of a Wishart distribution. The use Fisher-Bingham is better suited than vMF distributions to model eigenvectors, as they are by construction invariant to eigenvector changes of sign. Our choice of using vMF distributions instead was motivated by the very simple interpretability of the distribution's parameters, which allows discussing on the interpretation of the results with neuroscientists. The sign indetermination problem is addressed by the column permutation step in the algorithm. The main computational hurdle - eigenvectors/eigenvalues permutation invariance - remains present for Fisher-Bingham distributions.

Our choice of computing the Maximum Likelihood Estimator (or the Maximum A Posteriori if a prior is used) rather working in a Bayesian framework is motivated by computational efficiency. The algorithm we propose converges very fast in practice, even for mixture models, and can scale to relatively large numbers of eigenvectors p (e.g., $p = 20$ in experiments on synthetic data). Extensions could be considered using Bayesian conjugate posterior distributions.

In terms of flexibility, the formulation of the MCMC-SAEM, with a MCMC method agnostic to the expression of the density, allows easily changing the model structure, e.g., to account for positive eigenvalues or binary coefficients, only by modifying the expression of the likelihood and

its gradient. The model of Hoff [2009a] could also adapt to different settings, by replacing the Wishart individual-level variability by other forms of perturbation, but this might require changes in the algorithm.

Another closely related model is the recent work of Duan et al. [2020]. In this chapter, the authors work with the Laplacian matrices of the networks. The *smallest* eigenvalues only are of interest, as opposed to our model. As in Hoff [2009a], the authors work with Fisher-Bingham distributions, in a Bayesian framework. Their model is used to rely on multiple observations of a similar network structure to perform clustering on brain regions.

Finally, the works of Wang et al. [2019] and Durante et al. [2017] provide an alternative low-rank modeling approach for populations of binary network matrices. The authors propose to find the optimal value for the individual variables characterizing the deviation from the mean, and solve the related optimization problem. In contrast, we propose a hierarchical model to describe the variability of the individual variables, and we maximize the likelihood of the observed matrices $p(A;\theta)$, as opposed to the complete likelihood $p(A, X, \lambda;\theta)$. As a central difference with our work, these approaches are template-based: they model binary matrices by low-rank logits, expressing the logit matrix of each individual as a sum of a common template and low-rank deviations from it. Roughly speaking, our method instead relies on the fact that the deviations from the mean are strongly related to the structure of the mean; this relation is accounted for by removing the template from the model and working with non-centered low-rank deviations $\lambda \cdot X$.

Future directions. This work focuses on cross-sectional network data sets, i.e., populations where each adjacency matrix belongs to a different subject and is independent of the others. Our model could also be used as a base framework for longitudinal network modeling using the tools proposed by Schiratti et al. [2015]. This would consist of considering time-dependent latent variables X and λ for each subject, evolving close to a population-level reference trajectory in the latent space. We could not elaborate this direction further during this thesis, as the UK BioBank currently only has at most two points per subject.

Future work could investigate the dependencies between the latent variables of the model. Correlation can be introduced between the patterns by using Fisher-Bingham distributions on the Stiefel manifold [Hoff, 2009b] and between pattern weights with full Gaussian covariance matrices. Another direction to develop is the quantification of the uncertainty: by adding prior distributions on F and μ, a Bayesian analysis would naturally provide posterior confidence regions for the model parameters [Pal et al., 2020]. Finally, our framework could be adapted to model graph Laplacian matrices instead of adjacency matrices. The analysis of the eigenvalues and eigenvectors of the graph Laplacian has proven of great theoretical [Hammond et al., 2011] and practical [Atasoy et al., 2016] interest in network analysis. Understanding the variability of the eigendecomposition of graph Laplacians could help to design robust models relying on spectral graph theory.

In the next chapter, we will be investigating on the theoretical soundness of the model we proposed. In particular, we will prove its identifiability and the convergence of the Maximum A Posteriori estimator.

5.A SAEM Maximization Step

5.A.1 Maximum Likelihood Estimates for μ, σ_λ^2, σ_ε^2

Up to a constant normalization term c, the complete log-likelihood of the model in the Gaussian case writes:

$$\begin{aligned}\log p((A^{(k)}),(X^{(k)}),(\lambda^{(k)});\theta) &= \sum_{k=1}^{N} \log p(A^{(k)},X^{(k)},\lambda^{(k)};\theta) \\ &= \sum_{k=1}^{N}\left[-\frac{1}{2\sigma_\varepsilon^2}\left\|A^{(k)}-\lambda^{(k)}\cdot X^{(k)}\right\|^2 - \frac{1}{2\sigma_\lambda^2}\left\|\lambda^{(k)}-\mu\right\|^2 + \mathrm{Tr}(F^\top X^{(k)})\right] \\ &\quad - Nn^2\log\sigma - Np\log\sigma_\lambda - N\log\mathcal{C}_{n,p}(F) + c \\ &= N\left[\mathrm{Tr}(F^\top S_1) + \langle S_2, \frac{1}{2\sigma_\lambda^2}\mu\rangle - S_3\frac{1}{2\sigma_\lambda^2} - S_4\frac{1}{2\sigma_\varepsilon^2} + \Psi(\theta)\right]\end{aligned} \tag{A1}$$

with $\Psi(\theta) = -n^2\log\sigma_\varepsilon - p\log\sigma_\lambda - \log\mathcal{C}_{n,p}(F) + c$, and

$$\begin{cases} S^1 = \frac{1}{N}\sum_{k=1}^N X^{(k)} \\ S^2 = \frac{1}{N}\sum_{k=1}^N \lambda^{(k)} \\ S^3 = \frac{1}{N}\sum_{k=1}^N \left\|\lambda^{(k)}\right\|^2 \\ S^4 = \frac{1}{N}\sum_{k=1}^N \left\|A^{(k)}-\lambda^{(k)}\cdot X^{(k)}\right\|^2 \end{cases}$$

The model thus belongs to the curved exponential family, and its sufficient statistics are given by (S^1,S^2,S^3,S^4). The log-likelihood is component-wise convex in μ, σ_λ^2 and σ_ε^2. Computing its gradient yields one single critical point, which is thus the maximum value.

In the case of the binary model, the log-likelihood writes:

$$\begin{aligned}\log p((A^{(k)}),(X^{(k)}),(\lambda^{(k)});\theta) &= \sum_{k=1}^{N}\sum_{i,j=1}^{n}\left[A_{ij}^{(k)}\log h(\lambda^{(k)}\cdot X^{(k)})_{ij} + (1-A_{ij}^{(k)})\log(1-h)(\lambda^{(k)}\cdot X^{(k)})_{ij}\right] \\ &\quad + \sum_{k=1}^{N} -\frac{1}{2\sigma_\lambda^2}\left\|\lambda^{(k)}-\mu\right\|^2 + \mathrm{Tr}(F^\top X^{(k)}) \\ &\quad - Np\log\sigma_\lambda - N\log\mathcal{C}_{n,p}(F) + c\end{aligned}$$

with $h(x) = 1/(1+\exp(-x))$ the sigmoid function, which applies component-wise on matrices. The distribution $(A \mid \lambda, X)$ is non-parametric and needs no estimation. Hence, for all the model parameters F, μ, σ_λ the MLE remains unchanged.

5.A.2 Saddle-Point Approximation of $\mathcal{C}_{n,p}(F)$

We recall the main steps to compute the approximation of $\mathcal{C}_{n,p}(F)$ proposed by Kume et al. [2013]. For more details on the justification of the approximation and applications to more general distributions, we refer the reader to the original paper. Our implementation provides a function `spa.log_vmf`, which computes this approximation. The approximation $\widehat{\mathcal{C}}_{n,p}(F)$ for von Mises-Fisher distributions is written in Equation (16) of Kume et al. [2013]:

$$\widehat{\mathcal{C}}_{n,p}(F) = \frac{2^p(2\pi)^{np/2-p(p+1)/4}}{|\hat{K}''|^{1/2}|\hat{C}|^{1/2}}\exp\left\{\frac{1}{2}\mathrm{vec}(F)^\top\hat{C}^{-1}\mathrm{vec}(F) - \sum_{i=1}^{p}\hat{\vartheta}_{ii}\right\}\exp(T-p/2)\,. \tag{A2}$$

Using the following definitions:

- $C(\vartheta) = -2I_{np} - 2\sum_{1\le i\le j\le p}\vartheta_{ij}(J_{ij}+J_{ji})$. The matrix J_{ij} is composed of p^2 blocks. Block (i,j) is the identity I_n and all the other blocks are set to zero.

- $K(\vartheta) = -\frac{1}{2}\log|C(\vartheta)| - \frac{1}{2}\mu^\top C(\vartheta)^{-1}\mu - \frac{1}{2}\mathrm{vec}(\mu)^\top \mathrm{vec}(\mu)$. In this formula, μ is the $n \times p$ diagonal matrix with diagonal p singular values ω of F. The function $K(\vartheta)$ is the cumulant generating function.
- $\hat{\vartheta}$ is the unique solution of the so-called saddle-point equation $K'(\vartheta) = \vartheta$. It has the explicit expression $\hat{\vartheta} = -1/(2\mathrm{Diag}(\hat{\phi}))$, with $\hat{\phi}_r = \left(n + \sqrt{n^2 + 4\omega_r^2}\right)/(2\omega_r^2)$
- $\hat{C}$ is given by $C(\hat{\vartheta})$ and $\hat{K}''$ by $K''(\hat{\vartheta})$.
- $\hat{K}''$ can be computed explicitly:

$$\hat{K}''_{(r_1,s_1),(r_2,s_2)} = \begin{cases} 0 & r_1 \neq r_2 \text{ or } s_1 \neq s_2, \\ n\hat{\phi}_r\hat{\phi}_s + \hat{\phi}_r\hat{\phi}_s(\omega_r^2\hat{\phi}_r + \omega_s^2\hat{\phi}_s) & r_1 = r_2 < s_1 = s_2, \\ 2n\hat{\phi}_r^2 + 4\omega_r^2\hat{\phi}_r^3 & r_1 = r_2 = s_1 = s_2 \end{cases}$$

- The parameter T is defined in Equation (8) of Kume et al. [2013] and computed in the supplementary material of the paper in the case of vMF distributions. In first approximation, T can be considered zero.

As in the original paper, we validate our implementation by comparing the result with the Monte Carlo estimate of the normalizing constant produced by uniform sampling on the Stiefel manifold.

Remark. The $-p/2$ factor comes from using $B = -I_{n\times p}/2$ (and thus $V = I_{n\times p}$) and compensating with Equation (22) of Kume et al. [2013] to handle vMF distributions, which otherwise have $B = 0$. This point is not stated explicitly in the main text of the paper, but it is explained in the related MATLAB implementation provided by the authors.

5.B Gradient Formulas

The MCMC-SAEM initialization heuristic, as well as the MALA transition kernel, require the gradients of the log-likelihood with respect to the latent variables. In this section, we compute these gradients for the model with Gaussian perturbation and the model with binary coefficients.

5.B.1 Model with Gaussian Perturbation

Consider the log-likelihood for the variables of only one subject (X, λ, A). Using the formula in Equation (A1), we can compute its gradients with respect to X and λ. For λ, it writes:

$$\nabla_\lambda \log p(X, \lambda, A; \theta) = -\left(\frac{1}{\sigma_\varepsilon^2} + \frac{1}{\sigma_\lambda^2}\right)\lambda + \frac{1}{\sigma^2}(x_i^\top A x_i)_{i=1}^p + \frac{1}{\sigma_\lambda^2}\mu\,,$$

with $(x_i)_{i=1}^p$ the columns of X. Similarly, the Euclidean gradient for X is given by

$$\nabla_X \log p(X, \lambda, A; \theta) = \frac{1}{\sigma_\varepsilon^2}AX\mathrm{Diag}(\lambda) + F + 4X\mathrm{Diag}(\lambda)X^\top X\mathrm{Diag}(\lambda)\,.$$

Following Edelman et al. [1998], the Riemannian gradient on the Stiefel manifold is then given by:

$$\nabla_X^{\mathcal{V}} \log p(X, \lambda, A; \theta) = \nabla_X \log p(X, \lambda, A; \theta)X^{(k)\top} - X\nabla_X \log p(X, \lambda, A; \theta)\,.$$

5.B.2 Binary Model

Similarly, the log-likelihood gradients can be derived for the binary model. Let $\tilde{x}_i$ be the i-th *row* of X and $\odot$ denote the entry-wise product. We have:

$$\nabla_\lambda \log p(X,\lambda,A;\theta) = -\frac{1}{\sigma_\lambda^2}(\lambda-\mu) + \sum_{i,j=1}^{n}[A_{ij}h(-(\lambda\cdot X)_{ij}) - (1-A_{ij})h((\lambda\cdot X)_{ij})](\tilde{x}_i \odot \tilde{x}_j)$$

$$\begin{aligned}\nabla_X \log p(X,\lambda,A;\theta) = F &+ \sum_{i\neq j}[A_{ij}h(-(\lambda\cdot X)_{ij}) - (1-A_{ij})h((\lambda\cdot X)_{ij})]H_{ij} \\ &+ \sum_{i=1}^{n}[A_{ii}h(-(\lambda\cdot X)_{ii}) - (1-A_{ii})h((\lambda\cdot X)_{ii})]K_i\,.\end{aligned}$$

In the latter formula, H_{ij} is an $n\times p$ matrix with zeros everywhere except the i-th row equal to $\lambda\odot\tilde{x}_j$ and the j-th row equal to $\lambda\odot\tilde{x}_i$. K_i is the $n\times p$ matrix with zeros everywhere except the i-th row equal to $2\lambda\odot\tilde{x}_i$.

5.C Algorithm for the Clustering Model

We summarize in Algorithm 5.C.1 the procedure to estimate the MLE of a mixture model.

5.D Symmetry of von Mises-Fisher Distributions

Let F be the parameter of a von Mises-Fisher distribution. Let $\exp_X$ be the Riemannian exponential map at X. We have the following result:

Proposition 3. *Suppose that the columns of F are orthogonal. Let $M=\pi_V(F)$ be the vMF distribution mode and $D\in T_M\mathcal{V}_{n,p}$ a tangent vector at M. Then $p_{\text{vMF}}(\exp_M(D)) = p_{\text{vMF}}(\exp_M(-D))$, i.e., the vMF distribution is symmetric around its mode.*

Proof. Since the columns of F are orthogonal, we can write $F=M\Lambda$ with $M=\pi_V(F)\in\mathcal{V}_{n,p}$ and $\Lambda=\text{Diag}(\lambda)$. Let $D\in T_M\mathcal{V}_{n,p}$. As proven in Edelman et al. [1998], the geodesic X_t starting at M with $X'(0)=D$ is then given by

$$X_t = (M, M_\perp)\exp(tK_M(D))I_{n,p}\,,$$

where exp is the matrix exponential, $M_\perp\in\mathcal{V}_{n,n-p}$ is such that $M^\top M_\perp = 0$ and $K_M(D)$ is skew-symmetric: $K_M(D)^\top = -K_M(D)$. Therefore, the von Mises-Fisher log-density along X_t writes as:

$$\begin{aligned}\text{Tr}(F^\top X_t) &= \text{Tr}(\Lambda M^\top (M, M_\perp)\exp(tK_M(D))I_{n,p}) \\ &= \text{Tr}(\Lambda I_{p,n}\exp(tK_M(D))I_{n,p}) \\ &= \text{Tr}(I_{n,p}^\top \exp(tK_M(D))^\top I_{p,n}^\top \Lambda^\top) \\ &= \text{Tr}(I_{p,n}\exp(tK_M(D)^\top)I_{n,p}\Lambda) \\ &= \text{Tr}(\Lambda I_{p,n}\exp(-tK_M(D))I_{n,p}) \\ &= \text{Tr}(F^\top X_{-t})\end{aligned}$$

Therefore, the von Mises-Fisher density is symmetric around its mode. □

5.E Additional Details on the UK Biobank Experiment

5.E.1 Impact of the Number p of Patterns

We perform the same experiment as in Section 5.5.2 with different numbers of patterns, $p\in\{2,10\}$, always running the MCMC-SAEM for 1000 iterations with 20 MCMC steps per SAEM iteration. We call the related models M2, M5, and M10. The normalized patterns of M2 and M10 are reproduced in Figures 5.E.1 and 5.E.2. The patterns of M2 correspond to patterns 1 and 2 of M5 and M10. Similarly, the patterns of M5 correspond to patterns 1 to 5 of M10. This result confirms

Algorithm 5.C.1: Maximum Likelihood Estimation of $\theta = (F, \mu, \sigma_\varepsilon, \sigma_\lambda, \pi)$ for the mixture model

Initialize θ_0 and S_0.
Initialize X_0, λ_0 and z_0 using the K-Means algorithm.
for $t = 1$ *to* T **do**
  if $(t \bmod 5) = 0$ **then**
    Align together the parameters $(F^c, \mu^c)_{c=1}^K$ of each cluster using Algorithm 5.4.2.
  end
  if $t \leq T/3$ *and* $(t \bmod 5) = 0$ **then**
    for $k = 1$ *to* N **do**
      Use Algorithm 5.4.2 to align $X_t^{(k)}$ with $\pi_V\left(F_t^{z_t^{(k)}}\right)$.
      Permute $\lambda_t^{(k)}$ accordingly.
    end
  end
  Set $\widetilde{X}_0^{(k)} = X_t^{(k)}$, $\widetilde{\lambda}_0^{(k)} = \lambda_t^{(k)}$, $\widetilde{z}_0^{(k)} = z_t^{(k)}$
  for $\ell = 1$ *to* n_{MCMC} **do**
    for $k = 1$ *to* N **do**
      Sample $\widetilde{X}_\ell^{(k)}$ from the Metropolis kernel $q_X(\cdot \mid \widetilde{X}_{\ell-1}^{(k)}, \widetilde{\lambda}_{\ell-1}^{(k)}, \widetilde{z}_{\ell-1}^{(k)}; \theta_t)$ targetting $p(X^{(k)} \mid A^{(k)}, \widetilde{\lambda}_{\ell-1}^{(k)}, \widetilde{z}_{\ell-1}^{(k)}; \theta_t)$.
      Sample $\widetilde{\lambda}_\ell^{(k)}$ from the Metropolis kernel $q_\lambda(\cdot \mid \widetilde{X}_\ell^{(k)}, \widetilde{\lambda}_{\ell-1}^{(k)}, \widetilde{z}_{\ell-1}^{(k)}; \theta_t)$ targetting $p(\lambda^{(k)} \mid A^{(k)}, \widetilde{X}_\ell^{(k)}, \widetilde{z}_{\ell-1}^{(k)}; \theta_t)$.
      Sample $\widetilde{z}_\ell^{(k)}$ from the distribution $p(z^{(k)} \mid A^{(k)}, \widetilde{X}_\ell^{(k)}, \widetilde{\lambda}_\ell^{(k)}; \theta_t)$.
    end
  end
  Set $X_{t+1}^{(k)} = \widetilde{X}_{n_{\text{MCMC}}}^{(k)}$, $\lambda_{t+1}^{(k)} = \widetilde{\lambda}_{n_{\text{MCMC}}}^{(k)}$ and $z_{t+1}^{(k)} = \widetilde{z}_{n_{\text{MCMC}}}^{(k)}$.
  Update the sufficient statistics $\overline{S}_{t+1} = (1 - \alpha_t)\overline{S}_t + \alpha_t S(A, X_{t+1}, \lambda_{t+1})$.
  Compute π_{t+1} using the proportion of samples $z_{t+1}^{(k)}$ belonging to each cluster.
  for $c = 1$ *to* K **do**
    Compute μ_{t+1}^c, $(\sigma_\varepsilon^c)_{t+1}$ and $(\sigma_\lambda^c)_{t+1}$ with Equation (5.5) using only the k such that $z_{t+1}^{(k)} = c$.
    Compute F_{t+1}^c by solving problem (5.6), using only the k such that $z_{t+1}^{(k)} = c$.
  end
end
return θ_T, $(X_t, \lambda_t, z_t)_{t=1}^T$

that our model acts in a way comparable to PCA, selecting first the dominant patterns with the largest eigenvalues. Figure 5.E.3 compares the posterior means of $\lambda \cdot X$ given by M2, M5 and M10 for 5 subjects. Coherently, the approximation $\lambda^{(k)} \cdot X^{(k)}$ refines and gets closer to $A^{(k)}$ as p increases. Over the 1000 subjects, these posterior means achieve, respectively, 57% (±7%), 47% (±5%) and 35% (±4%) relative RMSE.

However, this observation does not assess whether higher values of p provide additional relevant features to represent the network structure. The following result illustrates this idea. We repeat the experiment of missing link MAP imputation on models M2 and M10. We find that both M2 and M10 yield a worse prediction than M5 on this task: model M2 gets 70% (±16%) rRMSE and M10 gets 76% (±16%) rRMSE, whereas model M5 gets 65% (±15%) rRMSE. While the prediction performance of M2 is expected to be worse than M5's, observing a worse prediction performance in M10 means that the information captured by the additional components does not help infer the network structure. As with PCA, the components with lesser amplitude are less relevant to perform regression tasks; this idea is at the core of Partial Least Square Regression Wold et al. [2001].

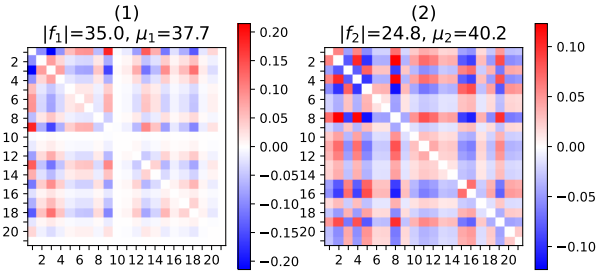

Figure 5.E.1: Normalized connectivity patterns when $p = 2$, computed as in Figure 5.10.

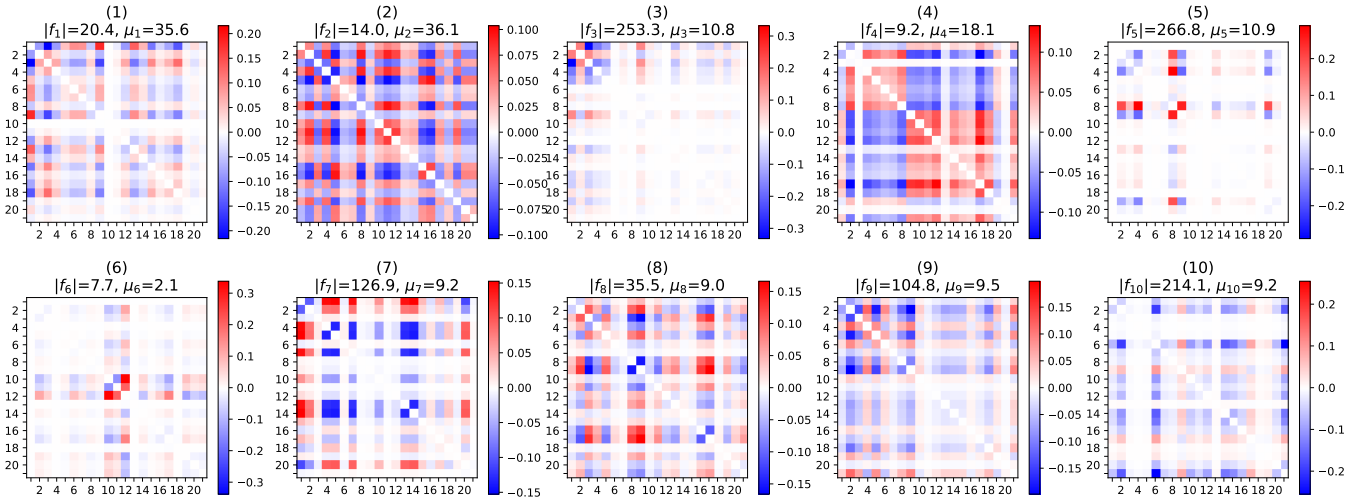

Figure 5.E.2: Normalized connectivity patterns when $p = 10$, computed as in Figure 5.10.

Therefore, the parameter p should be chosen with care when using our model for predictive purposes. The experiment presented above can be used to quantify the relevance of the obtained representation, but other methods could be explored. Future work could investigate the question of parameter selection by adapting Bayesian model selection methods to our method, as well as likelihood ratio tests or criteria like BIC and AIC.

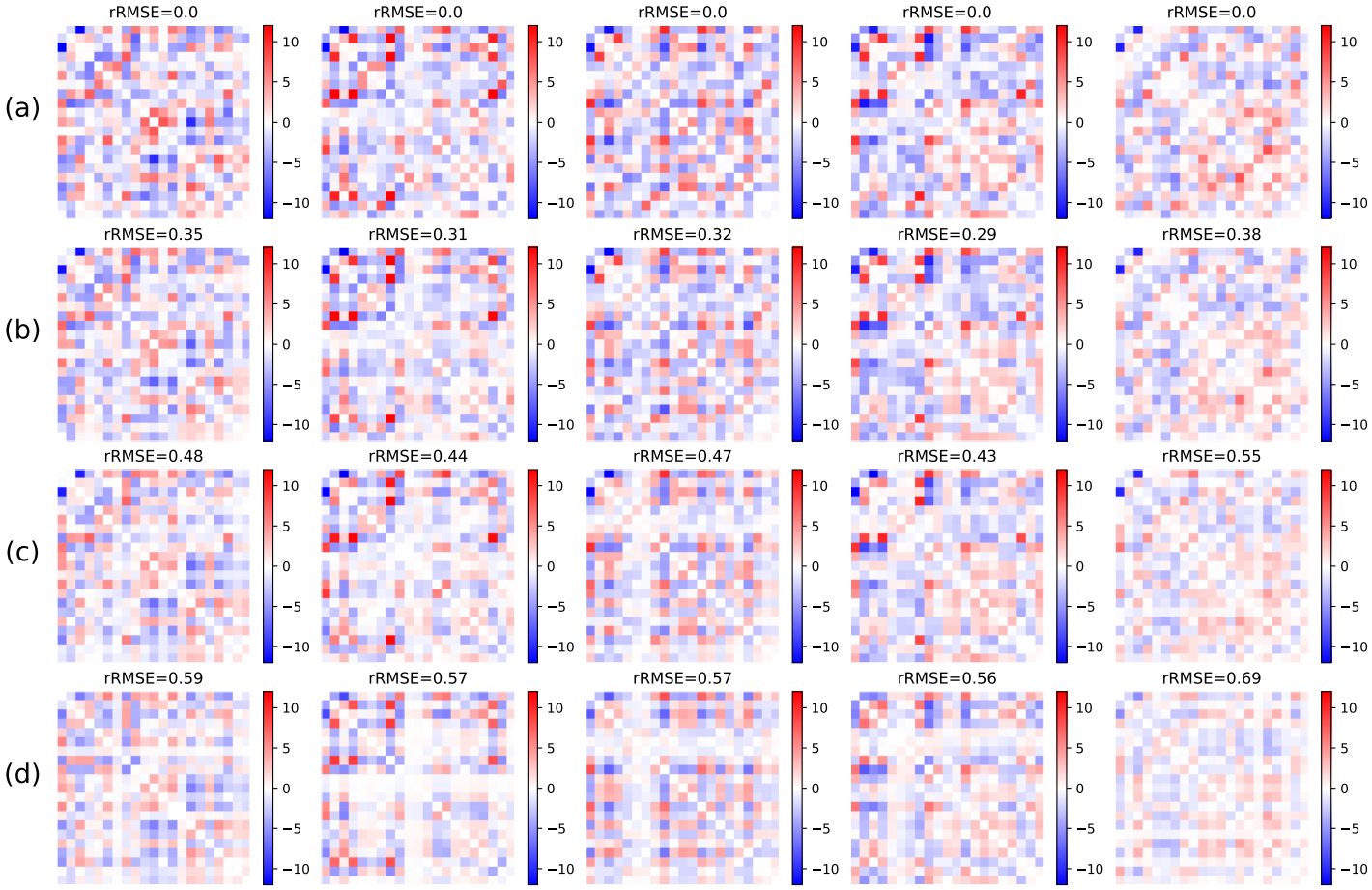

Figure 5.E.3: (**a**) UK Biobank connectivity matrices for 5 subjects. (**b**) M10 posterior mean value of $\lambda \cdot X$. (**c**) M5 posterior mean value of $\lambda \cdot X$. (**d**) M2 posterior mean value of $\lambda \cdot X$. The rRMSE coherently increases as p decreases.

5.E.2 Brain Regions of the UK Biobank fMRI Correlation Networks

As explained in Section 5.5.2, the Regions Of Interest (ROIs) that define the correlation networks are detected automatically using a spatial ICA Kiviniemi et al. [2003]. Each component of the ICA attributes a weight to each brain voxel. The brain regions are visualized by selecting the voxels with weight above a certain threshold. The obtained level set may be scattered over the brain, which sometimes makes their interpretation difficult. In Figure 5.E.4, we show the brain regions mentioned in the interpretation of the patterns identified by our model, namely regions 1, 2, 3, 4, 8, 9, 10, 11, 12, 17, 19. In this figure, as well as online, the ICA weight threshold value is set to 5.

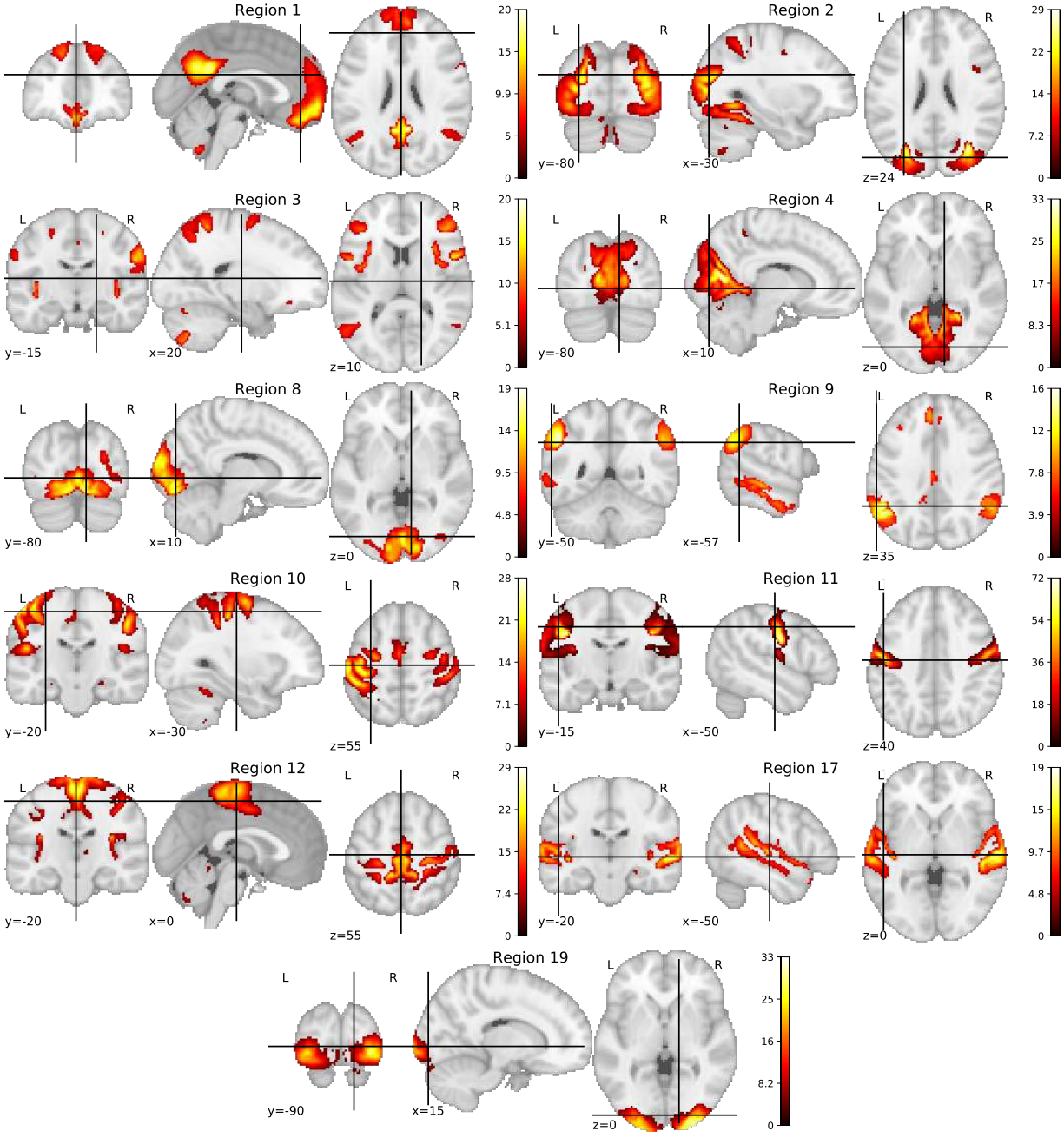

Figure 5.E.4: Frontal, sagittal, and transverse cuts of the brain for the UK Biobank fMRI brain regions analyzed in this chapter. As explained in Section 5.5.2, region 1 comprises part of the Default Mode Network of the brain, which characterizes its activity at rest. Region 3, which is anti-correlated to region 1, is related to sensory functions. Regions 2, 4, 8, 9, and 19 are involved in the visual functions. Regions 10, 11, 12 correspond to motor control. Region 17 is involved in memory and spatial navigation. The L/R letters distinguish the left and right hemispheres. The black axes on each view give the three-dimensional position of the cut. The color strength corresponds to the truncated ICA weight.

Chapter 6

Asymptotic Analysis of the Spectral Model

Matrix data sets arise in network analysis for medical applications, where each network belongs to a subject and represents a measurable phenotype. These large dimensional data are often modeled using lower-dimensional latent variables, which explain most of the observed variability and can be used for predictive purposes. In this chapter, we provide asymptotic convergence guarantees for the estimation of the hierarchical statistical model of Chapter 5. This model captures the variability of matrices by modeling a truncation of their eigendecomposition. We show that this model is identifiable, and that consistent Maximum A Posteriori (MAP) estimation can be performed to estimate the distribution of eigenvalues and eigenvectors. The MAP estimator is shown to be asymptotically normal for a restricted version of the model.

Contents

6.1 Introduction

Latent variable models are powerful tools to capture the complexity of high-dimensional data. Their hierarchical structure decouples this complexity into a low-dimensional distribution of latent variables and a mechanism to generate observations from latent variables. Over the last decades, they have proven relevant to perform regression and classification tasks as well as to provide interpretable representations of the data. In this chapter, we are interested more specifically in the analysis of matrix data sets: in this context, an observation is a matrix which represents the interactions between a given number of entities. The main case of interest is network data

set analysis, where matrices represent the evolution of a given network across time, or the same network structure measured in different situations.

Recently, the analysis of network data sets has received increased attention in the literature, in particular for medical applications, where each network represents a different patient, typically its brain connectivity network. The need to understand the complex structure of the interactions within networks has brought the development of low-dimensional representation of these networks, with methods like sparse dictionary learning or graph auto-encoders [D'Souza et al., 2018, Li et al., 2019]. In many cases, the core modeling assumption relies on the low rank of the observed matrices [Chen et al., 2020]. In that regard, such models can be interpreted as constraints on the distribution of the eigenvalues and the eigenvectors. However, although these recent works have achieved great performance on practical tasks, little has been done in the literature so far to analyze their theoretical soundness.

In this chapter, we provide an asymptotic analysis for the network data set analysis model of Chapter 5 which, in terms of generative modeling, can be considered a generalization of several current similar models relying on graph auto-encoders [Kipf and Welling, 2017] and dictionary learning [D'Souza et al., 2019b]. The model quantifies the variability in the spectral decomposition of network adjacency matrices: the leading eigenvectors, taking values in the Stiefel manifold, and the related eigenvalues are considered as latent variables in a hierarchical generative model. It relies on the classical assumption that the relevant information in a matrix of interaction coefficients can be captured by a low-rank approximation [Shabalin and Nobel, 2013]. The model structure introduced in Chapter 5 was shown to be able to account for the complex variability of functional brain networks using a restricted number of parameters, and provides an interpretable representation of this variability.

We first show that the model is identifiable, and consider the parameter estimation problem. We show that, although the Maximum Likelihood Estimator may not be defined, the Maximum A Posteriori estimator exists for wide classes of prior distributions. Finally, we show the almost sure consistency of the estimator and its asymptotic normality as the number of samples goes to infinity. The technical difficulties arise from the hierarchical structure of the model: only a few specific such cases have received attention in the literature. For instance, the identifiability of latent variable models remains an open question for most latent variable network analysis models. Although our results take stock on the model structure, we believe that they can be transposed without hurdle to many similar models.

Notations

In the next sections, we use the following notations:

- $A^\top$ denotes matrix transposition, $\mathrm{Tr}(A)$ the trace and $\det(A)$ the determinant,
- $\|x\|$ denotes the canonical Euclidean norm for vectors, and the related operator norm for matrices,
- $\|A\|_F$ denotes the Frobenius norm and $\langle A, B\rangle_F = \mathrm{Tr}(A^\top B)$ the related inner product for matrices,
- If X is an $n \times p$ matrix, $x_i \in \mathbb{R}^n$ denotes its i-th column, so that $X = (x_1, \dots, x_p)$,
- $\mathcal{V}_{np}$ is the Stiefel manifold of $n \times p$ matrices X such that $X^\top X = I_p$.
- $O_n(\mathbb{R})$ is the orthogonal group $\mathcal{V}_{nn}$,
- For λ a vector and X a matrix, we define $\lambda \cdot X = X^\top \mathrm{Diag}(\lambda) X$,
- For A an $n \times n$ matrix and X an $n \times p$ matrix, we define $A * X = (x_i^\top A x_i)_{i=1}^p$.

Appendix 6.A provides a comprehensive table describing the variables used throughout the chapter.

6.2 A Statistical Model for Spectral Decomposition

6.2.1 Model Definition

Observations distribution

We study the generative model for sets of weighted graph adjacency matrices $A_1, \dots, A_N \in \mathbb{R}^{n \times n}$ proposed in Chapter 5. It draws symmetric low rank adjacency matrices A by generating their eigenvectors $X = (x_1, \dots, x_p) \in \mathbb{R}^{n \times p}$ and eigenvalues $\lambda = (\lambda_1, \dots, \lambda_p) \in \mathbb{R}^p$, and combining them with an additive noise $\varepsilon \in \mathbb{R}^{n \times n}$.

$$A = X \mathrm{Diag}(\lambda) X^\top + \varepsilon \tag{6.1}$$

In practice, the adjacency matrix A represents a network. n corresponds to the number of nodes (e.g., in the case of brain connectivity, the number of brain regions), and $p \ll n$ is chosen such that the residual term ε is small. The eigenvectors take values in the Stiefel manifold $\mathcal{V}_{np}$ of matrices such that $X^\top X = I_p$. Their probability distribution will be described in the next section. The eigenvalues follow a multivariate Gaussian distribution $\lambda \sim \mathcal{N}(\mu, \sigma_\lambda^2 I_p)$. The noise ε is a symmetric matrix whose coefficients above the diagonal also follow a Gaussian distribution $\mathcal{N}(0, \sigma_\varepsilon^2 I_{n \times (n+1)/2})$. We assume that the variables λ, X, ε are independent. This assumption is strong: it might not be satisfied in practice, as the variation of a pattern x_i should be naturally correlated to a variation of the related λ_i. However, it also allows keeping a restricted number of parameters, which allows for robust estimation in practice when the number of observed matrices is low. Their interpretations will be given in Section 6.2.2 on simpler alternative models.

Eigenvectors distribution

As an element of the Stiefel manifold $\mathcal{V}_{np}$, the eigenvector matrix X is described by a probability distribution over $\mathcal{V}_{np}$. The canonical framework for these distributions is exposed in Chikuse [2003c], and consists in taking a measure with density with respect to the invariant measure over the Stiefel manifold. The invariant measure $[\mathrm{d}X]$ is defined, up to a constant, as the only measure invariant to orthogonal transformations, i.e., for $S \subset \mathcal{V}_{np}$ and $O \in O_n(\mathbb{R}), O' \in O_p(\mathbb{R})$:

$$\int_S [\mathrm{d}X] = \int_{OS} [\mathrm{d}X] = \int_{SO'} [\mathrm{d}X] \,.$$

It can be rescaled by a constant factor to correspond to the Hausdorff measure over $\mathcal{V}_{np}$ [Jauch et al., 2020b].

The distribution considered for X is the von Mises-Fisher (vMF) distribution, also called Matrix Langevin distribution in the literature. It was first introduced by Khatri and Mardia [1977], who derived basic properties of the distribution and its Maximum Likelihood Estimator (MLE), and was further studied for both theoretical and algorithmic purposes [Jupp and Mardia, 1979, Chikuse, 2003a, Kume et al., 2013, Pal et al., 2020]. The von Mises-Fisher distribution over $\mathcal{V}_{np}$ is defined by its probability density function (p.d.f.) with respect to the invariant measure:

$$p(X) = \frac{1}{\mathcal{C}(F)} \exp(\mathrm{Tr}(X^\top F)) = \frac{1}{\mathcal{C}(F)} \exp\left(s_1 \langle x_1, m_1 \rangle + \dots + s_p \langle x_p, m_p \rangle\right) , \tag{6.2}$$

with $\mathcal{C}(F)$ the normalizing constant and

$$F = (f_1, \dots, f_p) = M \mathrm{Diag}(s) = (m_1, \dots, m_p) \mathrm{Diag}(s_1, \dots, s_p)$$

the parameter of the distribution ($F \in \mathbb{R}^{n \times p}$). In the model considered here, $M \in \mathcal{V}_{np}$ and the s_i's are non-negative to ensure identifiability. By definition, the modal point M has maximal probability. The s_i's control the spread around the modal point, and are called the *concentration parameters* of the distribution.

The vMF distribution has a simple interpretation and requires few parameters. It imposes no dependency between the columns of X, except the orthogonality constraint. It forms an exponential family of distributions, and as such lends itself to efficient numerical estimation procedures. The normalizing constant $\mathcal{C}(F)$ has an analytic expression relying on the hypergeometric function of a

matrix argument, and represents the main difficulty when analyzing the distribution, as it prevents from getting an explicit expression of its moments.

With this definition, we can write the full density of the model defined in the previous section. The likelihood of an observed matrix A writes:

$$\begin{aligned} p(A \mid \theta) &= \iint_{\mathcal{V}_{np}\times\mathbb{R}^p} p(A \mid X, \lambda, \theta) p(X \mid \theta) p(\lambda \mid \theta) \, [\mathrm{d}X] \mathrm{d}\lambda \\ &= \iint_{\mathcal{V}_{np}\times\mathbb{R}^p} \frac{1}{\mathcal{C}(F)(2\pi)^{(n^2+p)/2}\sigma_\varepsilon^{n^2}\sigma_\lambda^p} \exp\left(\langle X, F\rangle_F - \frac{1}{2\sigma_\lambda^2}\left\|\lambda - \mu\right\|^2 - \frac{1}{2\sigma_\varepsilon^2}\left\|A - \lambda\cdot X\right\|_F^2 \right) [\mathrm{d}X] \mathrm{d}\lambda \,, \end{aligned}$$

where we denote $\lambda \cdot X = X\mathrm{Diag}(\lambda)X^\top$ to lighten the formula, and $\theta = (F, \mu, \sigma_\lambda, \sigma_\varepsilon)$ regroups the model parameters.

Remark. The overall model structure (6.1) can be compared with Equation (1.11) in Forrester [2010], which states that, for any continuous probability distribution $p(A)$ over the space of symmetric matrices: for any bounded continuous function h,

$$\int_{\mathbb{R}^{n\times n}} h(A)p(A)\,\mathrm{d}A = \iint_{O_n(\mathbb{R})\times\mathbb{R}^n} h(\lambda\cdot X)p(\lambda\cdot X)\prod_{i<j}(\lambda_i - \lambda_j)\,[\mathrm{d}X]\mathrm{d}\lambda\,,$$

with $[\mathrm{d}X]$ the normalized invariant measure over the group of orthogonal matrices $O_n(\mathbb{R})$. In other words, any matrix distribution is equivalently characterized by the joint distribution of its eigenvalues and eigenvectors. In that regard, our main hypotheses consist in constraining on the number of non-zero eigenvalues and imposing that the distributions of X and λ can be decoupled.

6.2.2 Motivation: Network Modeling

Beyond the graphon model

The graphon [Lovász, 2012] is the standard reference model used in network theory to analyze large graphs from a probabilistic perspective. Many pieces of work in both the theoretical [Khetan and Mj, 2018, Janson, 2013b, Xu, 2018] and applied [Latouche and Robin, 2016, Sischka and Kauermann, 2022, Mukherjee and Chakrabarti, 2019] literatures focus on the properties of the model it describes and its statistical estimation.

A graphon is a symmetric function $w : [0,1]^2 \to [0,1]$, which is to be understood as a continuous adjacency matrix with an infinite number of nodes. The graphon defines a distribution over $n \times n$ symmetric adjacency matrices by drawing n uniform numbers $U_1, \dots, U_n \sim \mathcal{U}([0,1])$, and forming the matrix $A_{ij} = w(U_i, U_j)$, or $A_{ij} \sim \mathcal{B}(w(U_i, U_j))$ in the case of binary networks. The graphon inference problem thus consists, given one or several matrices A, in determining both the function w and the positions (U_i) of the nodes.

The main application of the graphon model is the Stochastic Block-Model (SBM), which assumes that w is block-wise constant. It amounts to dividing the set of nodes into clusters with given probabilities, and determining the connection between the nodes with the connection between their clusters. The SBM provides a well-studied [Olhede and Wolfe, 2014, Aicher et al., 2015, Peixoto, 2020] framework which is particularly relevant for a clustering analysis of networks, i.e., finding the most relevant partition among the nodes.

Both the graphon model and the SBM were conceived to analyze networks where nodes are drawn randomly and play interchangeable roles. They mostly focus on understanding the structure of the hidden graphon dynamic, which requires identifying the U_i's or the cluster labels.

Given a data set of matrices, both graphon and SBM would either 1) assume that the U_i's are drawn independently for each matrix, or 2) take the same U_i's for each matrix in the data set. The first case yields a distribution whose expectation has constant coefficients: $\mathbb{E}[A_{ij}] = \mathbb{E}[w(U, U')]$ with $U, U' \sim \mathcal{U}([0,1])$. The second case results in a constant distribution with $A_{ij} = w(U_i, U_j)$ for every sample matrix A, or a matrix of independent Bernoulli variables $A_{ij} \sim \mathcal{B}(w(U_i, U_j))$ in the case of binary networks. Both options lead to simplistic distributions which are not relevant from a practical perspective.

In the context considered here, the nodes remain the same from one matrix to another (e.g., brain regions), and cannot be permuted. This allows easily estimating the average interactions, which is the main difficulty for the graphon and the SBM. Modeling the matrices' spectral decomposition goes one step further than the SBM, and induces a dependency between the coefficients. It allows for instance computing the distribution of a set of matrix coefficients given other observed matrix coefficients.

Accounting for the full network variability

Two similar approaches currently co-exist in the literature to analyze sets of networks. On the one hand, Variational Graph Auto-Encoders (VGAE) [Kipf and Welling, 2017] assume that each node i is represented by a low-dimensional vector $z_i \in \mathbb{R}^p$, and models the adjacency matrix as $A_{ij} = h(z_i^\top z_j)$, with h a non-linear function. The model thus characterizes A by a low-dimensional representation $Z \in \mathbb{R}^{n\times p}$, and retrieves $A = h(Z^\top Z) = h(\mathbf{1}_p \cdot Z)$. The matrices $\mathbf{1}_p \cdot Z$ are constrained to having positive eigenvalues. Additionally, the VGAE model considers all variables z_i as independent and identically distributed.

On the other hand, a dictionary model was proposed by [D'Souza et al., 2019b], and writes each adjacency matrix A in the data set as a weighted combination of fixed rank-one matrices: $A = \lambda_1 x_1 x_1^\top + \cdots + \lambda_p x_p x_p^\top$, which rewrites as $\lambda \cdot X$. Here, the goal is to find the best λ for each matrix A, while the matrix X is the same for all networks. This model thus imposes a strong dimension constraint on the adjacency matrices.

Each of these two approaches capture one aspect of the variability: for the VGAE, only the "eigenvectors" vary, and for the dictionary model, only the "eigenvalues" depend on the network. The model we study here simultaneously accounts for these two sources of variability, and thus allows for a richer representation, while keeping a latent space dimension comparable to that of VGAE. From the VGAE perspective, the rows $(M_{ki})_{i=1}^p$ shape the distribution of z_k, and the parameters (μ, σ_λ^2) determine the (possibly non-positive) inner products between the z_i's. From the dictionary model perspective, the column $m_i = (M_{ki})_{k=1}^n$ gives the i-th dictionary element and s_i its concentration; the coefficient M_{ki} gives the strength of the contribution of pattern i to the interactions of node k in the network. The parameters (μ, σ_λ^2) give the distribution of the dictionary weights.

6.2.3 Conditional Distribution

Summarizing the model definition in Section 6.2.1, we assume that an observed adjacency matrix A writes as $A = \lambda \cdot X + \varepsilon$, with $(\lambda, X) \in \mathbb{R}^p \times \mathcal{V}_{np}$ being independent latent variables and ε a symmetric matrix of Gaussian distributed noise coefficients. The full model p.d.f. writes:

$$p(A, X, \lambda \mid \theta) = \frac{1}{\mathcal{C}(F)(2\pi)^{(n^2+p)/2}\sigma_\varepsilon^{n^2}\sigma_\lambda^p} \exp\left(\langle X, F\rangle_F - \frac{1}{2\sigma_\lambda^2}\left\|\lambda - \mu\right\|^2 - \frac{1}{2\sigma_\varepsilon^2}\left\|A - \lambda \cdot X\right\|_F^2\right).$$

From this expression, we can express the conditional distribution of the latent variables (X, λ) given A as follows. In the remainder of the chapter, we will denote

$$A * X = (x_k^\top A x_k)_{k=1}^p, \quad \frac{1}{\sigma_p^2} = \frac{1}{\sigma_\varepsilon^2} + \frac{1}{\sigma_\lambda^2} \quad \text{and} \quad \mu_{AX} = \sigma_p^2\left[\frac{1}{\sigma_\varepsilon^2} A * X + \frac{1}{\sigma_\lambda^2}\mu\right]. \tag{6.3}$$

The expression of the conditional density $p(X, \lambda \mid A, \theta)$ of the latent variables given the observed variable A writes as:

$$\begin{cases} p(X \mid A, \theta) \propto \exp\left(\langle X, F\rangle_F + \frac{1}{2\sigma_p^2}\left\|\mu_{AX}\right\|^2\right) \\ p(\lambda \mid X, A, \theta) = \mathcal{N}(\mu_{AX}, \sigma_p^2)\,. \end{cases}$$

The proof of this equation follows the same lines as in Lemma 6 in Appendix 6.D. We will be using this expression of the conditional distribution in Section 6.5 on asymptotic normality. The $\frac{1}{2\sigma_p^2}\left\|\mu_{AX}\right\|^2$ term in the distribution of $(X \mid A)$ is typically much larger than $\langle X, F\rangle_F$ as long as $n \gg p$, and it thus determines the shape of the distribution. As shown in the following proposition, it is maximized by the eigenvectors of A.

Proposition 4. *For $A \in \mathbb{R}^{n\times n}$, $\|\mu_{AX}\|^2$ is maximized by taking X among the eigenvectors of A. Furthermore, if the eigenvalues of A all have multiplicity one, this maximization is strict.*

Proof. Let $A = U^\top DU$ be the eigendecomposition of A, with $U^\top U = I_n$. Without loss of generality, we take $\sigma_\lambda = \sigma_\varepsilon = 1$. We have:

$$\begin{aligned}
\max_{X\in\mathcal{V}_{np}} 2\|\mu_{AX}\|^2 &= \max_{X\in\mathcal{V}_{np}} \sum_{i=1}^{p} (x_i^\top A x_i + \mu_i)^2 \\
&= \max_{Y\in\mathcal{V}_{np}} \sum_{i=1}^{p} (y_i^\top (D + \mu_i I_n) y_i)^2 \quad \text{(setting } Y = UX\text{)} \\
&= \max_{Y\in\mathcal{V}_{np}} \sum_{i=1}^{p}\sum_{k=1}^{n} [(d_k + \mu_i) y_{ik}^2]^2 \\
&\leq \max_{Y\in\mathcal{V}_{np}} \sum_{i=1}^{p}\sum_{k=1}^{n} (d_k + \mu_i)^2 y_{ik}^2 \quad \text{(Jensen's inequality)} \\
&= \max_{Y\in\mathcal{V}_{np}} \langle K, Y \odot Y\rangle_F \,.
\end{aligned}$$

With $K \in \mathbb{R}^{n\times p}$ defined by $K_{ij} = d_k + \mu_i$ and $Y \odot Y$ the Hadamard (entry-wise) product. If we extend K to a $n \times n$ matrix K' by padding zeros, and extend Y to an orthogonal matrix Q by completing Y into a basis, the objective function remains unchanged: $\langle K, Y\odot Y\rangle_F = \langle K', Q\odot Q\rangle_F$.

Since Q is orthogonal, the matrix $S = Q \odot Q$ is doubly stochastic. Furthermore, the Birkhoff-von Neumann theorem states that the set of doubly stochastic matrices is the convex hull of the set of permutation matrices. As a consequence, the linear function $\langle K', S\rangle_F$ is maximized by taking for S a permutation matrix. Such matrices are orthogonal and verify $S \odot S = S$, and their only square roots for the Hadamard product are permutation matrices with negative coefficients allowed. Therefore, the optimal choice for Y has its columns in the canonical basis. Hence, the optimal choice for $X = U^\top Y$ is to take its columns among the eigenvectors of A. [1]

When Y is a permutation matrix, Jensen's inequality becomes an equality, so that taking the related $X = U^\top Y$ is also an optimal choice for the original objective $\|\mu_{AX}\|^2$. Furthermore, if A has n distinct eigenvalues, Jensen's inequality is strict except when y_i is a vector of the canonical basis. Therefore, in that case, the optimal subset of eigenvectors of A (up to permutation and change of sign) is the only maximizer of $\|\mu_{AX}\|^2$. □

Remark. When taking $\mu = 0$, the result can be proved more simply by using Ky Fan's principle on eigenvectors [Fan, 1949]. A closely related, yet different result, was recently obtained by Liang et al. [2021]. We believe that obtaining a closed-form formula for maximizing the complete conditional density $p(X \mid A, \theta)$ would require significantly more work. The eigenvectors of A are no longer optimal: the best value of X is obtained as a trade-off between M and the closest optimal eigenvalue combination of A, with the concentration and variance parameters determining the balance between both.

6.3 Model Identifiability

Identifiability of statistical model $p(x \mid \theta)$ refers to the property that, if $\theta_1 \neq \theta_2$, then the distributions $p(\cdot \mid \theta_1)$ and $p(\cdot \mid \theta_2)$ must differ. It is a generally desirable property, as it ensures that the model is well-defined and behaves in an intuitive way. It is also of immediate theoretical interest, since it enables to prove that Maximum Likelihood Estimators converge to the correct value when the data is generated according to the model. For instance, it can be proved by retrieving the parameter θ from a set of moments of $p(\cdot \mid \theta)$.

The identifiability of latent variable models is a general, long-standing question, which has been studied and proved for only few specific models. It relates to the question of identifying the

[1] The authors thank the `math.stackexchange.com` community member `user1551` for his helpful answer on the Birkhoff-von Neumann theorem.

parameters of graphical models where only a fraction of the variables is observed. Much work has been devoted to the identifiability of finite mixture models [Teicher, 1963, Yakowitz and Spragins, 1968, Holzmann et al., 2004, Tabrizi et al., 2020]. In a similar spirit, classes of statistical models with discrete latent variables have also recently been proved to be identifiable [Allman et al., 2009, Gu and Xu, 2021]. Partial results have been shown for mixed-effects models, in particular in a longitudinal setting [Tabrizi et al., 2020, Lavielle and Aarons, 2016]. In a less closely related domain, identifiability results exist on time series model with latent variables [Douc et al., 2021]. Finally, general identifiability results are available for (possibly infinite) mixtures of exponential models [Barndorff-Nielsen, 1965, 1978]. Although the latter result is related to the model we consider here, its necessary theoretical conditions turn out to be hard to verify in practice.

The main difficulty with identifying latent variable models comes from the expression of the observations' likelihood:

$$p(A \mid \theta) = \iint_{\mathcal{V}_{np} \times \mathbb{R}^p} p(A, X, \lambda \mid \theta) \, [\mathrm{d}X] \mathrm{d}\lambda \, .$$

Even though our full model $p(A, X, \lambda \mid \theta)$ is identifiable, the marginalized model $p(A \mid \theta)$ may not be: permuting two eigenvalues μ_i, μ_j and the related eigenvector parameters f_i, f_j, or changing the sign of f_i does not change the distribution of A. This first obvious source of non-identifiability is easily overcome, by imposing that the normalized columns $(m_1, \dots, m_p)$ (denoting $m_i = f_i/|f_i|$) are sorted according to the lexicographical order and that each column has its first non-zero element positive. An additional constraint allows getting a provably identifiable marginal model: we shall assume that the f_i's are non-zero, i.e., that the concentration parameters $s_i = \|f_i\|$ are positive. These two constraints form the set of identifiable parameters Θ^{id}:

$$\Theta^{\mathrm{id}} = \{\theta \mid m_1 \prec \cdots \prec m_p \text{ and } \min_i s_i > 0\} \, .$$

With this definition, we have the following result:

Theorem 2. *If $p < n$, over Θ^{id}, different parameters $\theta_1 \neq \theta_2$ yield different marginal probability distributions $p(A \mid \theta_1)$ and $p(A \mid \theta_2)$.*

Proof. Given $\theta \in \Theta^{\mathrm{id}}$, we show that all parameters $(F, \mu, \sigma_\lambda, \sigma_\varepsilon)$ can be retrieved from the distribution $p(A \mid \theta)$. We first identify the noise variance. This allows identifying the eigenvalue parameters, and finally the eigenvector parameters.

Identifying σ_λ and σ_ε. Using Lemma 6 and $\alpha I_n * X = \alpha \mathbf{1}_p$, we have, for all $\alpha \in \mathbb{R}$:

$$p(A = \alpha I_n \mid \theta) = \frac{1}{\sqrt{2\pi}^{n^2} \sigma_\varepsilon^{n^2}} \frac{\sigma_p^p}{\sigma_\lambda^p} \exp\left(-\frac{1}{2\sigma_\varepsilon^2} n^2 \alpha^2 + \frac{\sigma_p^2}{2\sigma_\varepsilon^4} p^2 \alpha^2 + \|\mu\|^2 \left(\frac{\sigma_p^2}{2\sigma_\lambda^4} - \frac{1}{2\sigma_\lambda^2} \right) + \alpha \frac{\sigma_p^2}{\sigma_\lambda^2 \sigma_\varepsilon^2} \langle \mu, \mathbf{1}_p \rangle \right) ,$$

with $\sigma_p^{-2} = \sigma_\varepsilon^{-2} + \sigma_\lambda^{-2}$. The function $\alpha \mapsto \log p(A = \alpha I_n \mid \theta)$ is a second-order polynomial, its coefficients (a_0, a_1, a_2) can thus be identified. In particular, the degree two coefficient gives the value of

$$a_2 = -\frac{n^2}{2\sigma_\varepsilon^2} + \frac{p^2 \sigma_p^2}{2\sigma_\varepsilon^4} \, . \tag{6.4}$$

Similarly, the computation in Lemma 6 can be used to derive the gradient $\nabla_A p(A \mid \theta)$. It writes, for $A = \alpha I$:

$$\nabla_A p(A \mid \theta) = \frac{1}{\sigma_\varepsilon^2} p(A = \alpha I \mid \theta) \left(-\alpha I + \mathbb{E}[B] \right) ,$$

where B is the random variable given by $B = \lambda_p \cdot X$, with $\lambda_p \sim \mathcal{N}\left(\frac{\sigma_p^2}{\sigma_\lambda^2} \mu + \frac{\sigma_p^2}{\sigma_\varepsilon^2} \alpha \mathbf{1}_p, \sigma_p^2 \right)$. Furthermore, since we have

$$\mathbb{E}[B] = \mathbb{E}[X^\top \mathrm{Diag}(\lambda_p) X] = \sum_{i=1}^{p} \mathbb{E}[\lambda_{p,i}] \mathbb{E}[x_i x_i^\top] \, ,$$

we deduce:

$$\frac{\nabla_A p(A=\alpha I \mid \theta)}{p(A=\alpha I \mid \theta)} = \frac{1}{\sigma_\varepsilon^2}\left(-\alpha I + \sum_{i=1}^{p}\left[\frac{\sigma_p^2}{\sigma_\lambda^2}\mu_i + \frac{\sigma_p^2}{\sigma_\varepsilon^2}\alpha\right]\mathbb{E}[x_i x_i^\top]\right).$$

Finally, since $\operatorname{Tr}\mathbb{E}[x_i x_i^\top] = \langle \mathbb{E}[x_i x_i^\top], I\rangle_F = \mathbb{E}[x_i^\top I x_i] = 1$, we have:

$$\operatorname{Tr}\left(\frac{\nabla_A p(A=\alpha I \mid \theta)}{p(A=\alpha I \mid \theta)}\right) = \frac{1}{\sigma_\varepsilon^2}\left(-\alpha n + \frac{\sigma_p^2}{\sigma_\lambda^2}\langle \mu, \mathbf{1}_p\rangle + p\frac{\sigma_p^2}{\sigma_\varepsilon^2}\alpha\right).$$

As a consequence, the α-linear function above can be deduced from the distribution of A, hence we know its coefficients. In particular, the leading coefficient a_3 writes:

$$a_3 = -\frac{n}{\sigma_\varepsilon^2} + p\frac{\sigma_p^2}{\sigma_\varepsilon^4}.$$

The formulas of a_3 and a_2 in Equation (6.4) can be combined to obtain $-\frac{1}{2\sigma_\varepsilon^2}(n^2 - np) = a_2 - pa_3/2$. Therefore, since $p \neq n$, σ_ε can be identified, along with σ_λ.

Identifying μ. The moment generating function of A writes as:

$$G_A(T) = \mathbb{E}[e^{\langle T, A\rangle_F}] = \mathbb{E}[e^{\langle T, \lambda\cdot X + \varepsilon\rangle_F}] = G_{\lambda\cdot X}(T) \times G_\varepsilon(T).$$

Since the distribution of ε has been characterized, $G_\varepsilon(T)$ is known, and hence $G_{\lambda\cdot X}$ can be deduced as $G_A(T)/G_\varepsilon(T)$. As the moment generating function characterizes the probability distribution, if the distribution $\lambda\cdot X$ is identifiable then the distribution of A is identifiable. We thus turn on the problem of identifying μ given the distribution of $\lambda\cdot X$ (and proceed similarly for the eigenvector parameters in the next paragraph). We have for $t \in \mathbb{R}$:

$$\begin{aligned}\mathbb{E}[e^{t\lambda\cdot X}] &= \mathbb{E}\left[\sum_{k=0}^{\infty}\frac{1}{k!}t^k X^\top \operatorname{Diag}(\lambda)^k X\right] = \mathbb{E}\left[X^\top \operatorname{Diag}((e^{t\lambda_i})_{i=1}^p) X\right] \\ &= \sum_{i=1}^{p}\mathbb{E}[e^{t\lambda_i} x_i x_i^\top] = \sum_{i=1}^{p}\mathbb{E}[e^{t\lambda_i}]\mathbb{E}[x_i x_i^\top] = \sum_{i=1}^{p} e^{t\mu_i + \frac{1}{2}\sigma_\lambda^2 t^2}\mathbb{E}[x_i x_i^\top],\end{aligned}$$

which in particular gives $\operatorname{Tr}(\mathbb{E}[e^{t\lambda\cdot X}]) = \sum_{i=1}^{p} e^{t\mu_i + \frac{1}{2}\sigma_\lambda^2 t^2}$.

The functions of the form $t \mapsto e^{t\mu_i + \frac{1}{2}\sigma_\lambda^2 t^2}$ are linearly independent for distinct μ_i's: this allows retrieving both the μ_i's and the multiplicity count of each eigenvalue.

Identifying F. From there, we could use the matrices $\mathbb{E}[x_i x_i^\top]$ to identify the modal directions m_i. Indeed, as shown in Khatri and Mardia [1977] (Equations 2.9 to 2.11), each m_k is an eigenvector of each $\mathbb{E}[x_i x_i^\top]$. However, the related eigenvalues and remaining $n-p$ eigenvectors are unknown, and the relevant eigenvectors cannot be identified easily. In the limit of large concentration parameters, $\mathbb{E}[x_i x_i^\top] \simeq m_i m_i^\top$, so that the largest eigenvalue is the one corresponding to m_i. Yet this argument cannot be quantified, as the eigenvalues involve partial derivatives of $\log \mathcal{C}(F)$ which are hard to manipulate.

Instead, we get a better result by expressing the density of the distribution of the random variable $B = \lambda\cdot X$, which we also denote as $B = \mathscr{D}(\lambda, X)$, with support on the set $\operatorname{Im}(\mathscr{D})$ of $n\times n$ square matrices with rank p. The distribution of B is characterized by the expectations $\mathbb{E}[h(B)]$ with h continuous bounded. We have:

$$\mathbb{E}[h(B)] = \iint h(\mathscr{D}(\lambda, X)) \cdot p(\lambda \mid \theta) p(X \mid \theta)\,[\mathrm{d}X]\mathrm{d}\lambda. \tag{6.5}$$

We want to perform a change of variable to express the expectation as an integral over $\operatorname{Im}(\mathscr{D})$. However, this cannot be performed directly. First, the mapping $\mathscr{D}$ is not injective. Next, the most relevant change of variable formula for this problem is, to the best of our knowledge, the main

result of Traynor [1994], which gives a formula for mappings taking inputs in vector spaces (which is not the case here as $X \in \mathcal{V}_{np}$).

The first problem can be solved by splitting the integral over domains where $\mathscr{D}$ is injective, which means preventing permutation and change of signs in the columns of X. To that end, for $\pi \in S_p$ a permutation and $f \in \{\pm 1\}^p$, we denote $X_{\pi,f} = (f_1 x_{\pi(1)}, \dots, f_p x_{\pi(p)})$, and by $\lambda_\pi = (\lambda_{\pi(1)}, \dots, \lambda_{\pi(p)})$. We also define the sets

$$\begin{cases} \Delta_0 = \{X \in \mathcal{V}_{np} \mid x_1 \prec \cdots \prec x_p \text{ and the first non-zero coefficient of each column is } > 0\} \\ \Delta_{\pi,f} = \{X_{\pi,f} \mid X \in \Delta_0\}\,, \end{cases} \tag{6.6}$$

where $\prec$ denotes the lexicographical order over $\mathbb{R}^n$. By construction, we have $\mathcal{V}_{np} = \cup_{\pi,f} \Delta_{\pi,f}$: for each $X \in \mathcal{V}_{np}$, we can change the sign of its columns and sort the result in the lexicographic order to obtain a permuted version $X_{\pi f} \in \Delta_0$. Using this decomposition, we get

$$\mathbb{E}[h(B)] = \sum_{\pi,f} \iint_{\mathbb{R}^p \times \Delta_{\pi,f}} h(\mathscr{D}(\lambda, X)) \cdot p(\lambda \mid \theta)\, [\mathrm{d}X]\mathrm{d}\lambda\,.$$

Furthermore, the map $X \mapsto X_{\pi,f}$ corresponds to multiplying X by an orthogonal matrix. By construction, the invariant measure over $\mathcal{V}_{np}$ is invariant to this transformation [Chikuse, 2003c]. Moreover, the map $\lambda \mapsto \lambda_\pi$ is also a linear orthogonal transformation, and as such has Jacobian determinant one. Hence, we can perform the change of variable $(\lambda, X) \mapsto (\lambda_\pi, X_{\pi,f})$, and we get:

$$\begin{aligned} \mathbb{E}[h(B)] &= \sum_{\pi,f} \iint_{\mathbb{R}^p \times \Delta_0} h(\mathscr{D}(\lambda_\pi, X_{\pi,f})) \cdot p(\lambda_\pi \mid \theta) p(X_{\pi,f} \mid \theta)\, [\mathrm{d}X]\mathrm{d}\lambda \\ &= \iint_{\mathbb{R}^p \times \Delta_0} h(\mathscr{D}(\lambda, X)) \cdot \sum_{\pi,f} p(\lambda_\pi \mid \theta) p(X_{\pi,f} \mid \theta)\, [\mathrm{d}X]\mathrm{d}\lambda \\ &= \iint_{\mathbb{R}^p \times \Delta_0} h(\mathscr{D}(\lambda, X)) \cdot \sum_{\pi,f} p(\lambda \mid \theta_{\pi,f}) p(X \mid \theta_{\pi,f})\, [\mathrm{d}X]\mathrm{d}\lambda\,, \end{aligned}$$

with $\theta_{\pi,f} = (F_{\pi,f}, \mu_\pi, \sigma_\lambda, \sigma_\varepsilon)$.

The first problem is now solved, as $\mathscr{D}$ is injective over $\mathbb{R}^p \times \Delta_0$. We now need to get to an integral formulation over a vector space. To that end, we consider the inverse of the Cayley transform of X: $D = C^{-1}(X)$. We refer the reader to Appendix 6.B for a definition of the Cayley transform C. It is a smooth injective map from the tangent space at identity $T_{I_{np}}\mathcal{V}_{np} = \{\binom{A}{B} \mid A^\top = -A\}$ to the manifold $\mathcal{V}_{np}$, which covers the entire manifold apart from a set with measure zero. As explained in Jauch et al. [2020b] (Theorem 4.1), a change of variable from D to X can be performed, and amounts to adding a multiplicative factor $J_1(D)$, with J_1 is a generalized Jacobian determinant. It follows that we can rewrite:

$$\mathbb{E}[h(B)] = \iint_{\mathbb{R}^p \times C^{-1}(\Delta_0)} h(\mathscr{D}(\lambda, C(D))) \cdot \sum_{\pi,f} p(\lambda \mid \theta_{\pi,f}) p(X \mid \theta_{\pi,f}) \cdot J_1(D)\, \mathrm{d}D\mathrm{d}\lambda\,.$$

Since the map $D \mapsto C(D)$ is injective on $T_{I_{np}}\mathcal{V}_{np}$ (Equations (1-3) in Jauch et al. [2020b]), the map $(\lambda, D) \mapsto B = \mathscr{D}(\lambda, C(D))$ is injective over $\mathbb{R}^p \times C^{-1}(\Delta_0)$. Given B, we denote by λ_B, X_B and D_B its pre-images by $\mathscr{D}$ and C. Since the considered mapping is smooth, the main theorem of Traynor [1994] applies. Letting $J_2(\lambda, D)$ be the generalized Jacobian determinant involved in the formula, it writes as:

$$\mathbb{E}[h(B)] = \int_{\mathrm{Im}(\mathscr{D})} h(B) \cdot \sum_{\pi,f} p(\lambda_B \mid \theta_{\pi,f}) p(X_B \mid \theta_{\pi,f}) \cdot \frac{J_1(D_B)}{J_2(\lambda_B, D_B)}\, \mathrm{d}B\,.$$

where $\mathrm{d}B$ denotes the Hausdorff measure over $\mathrm{Im}(\mathscr{D})$. Since both maps C and $\mathscr{D}$ are diffeomorphic, the generalized Jacobian determinants involved are non-zero.

As a consequence, the random variable B has density

$$\sum_{\pi,f} p(\lambda_B \mid \theta_{\pi,f}) p(X_B \mid \theta_{\pi,f}) \cdot \frac{J_1(D_B)}{J_2(\lambda_B, D_B)}$$

over its support w.r.t. the Hausdorff measure. Therefore, if the distribution of B is known, we can deduce the value of the function $B \mapsto \sum_{\pi,f} p(\lambda_B \mid \theta_{\pi,f}) p(X_B \mid \theta_{\pi,f})$. For $X \in \Delta_0$ and $\lambda \in \mathbb{R}^p$, it comes that we know the value of

$$f_\lambda(X) = \sum_{\pi,f} p(\lambda \mid \theta_{\pi,f}) p(X \mid \theta_{\pi,f}) \,.$$

Since the sum above is invariant by any permutation π and change of sign f, it follows that the value of this expression is known not only for $X \in \Delta_0$, but over the whole manifold $\mathcal{V}_{np}$. Now, we consider the specific case $\lambda = \mu$. Up to a normalizing constant, $f_\mu(X)$ is a probability distribution over $\mathcal{V}_{np}$: it is a mixture of von Mises-Fisher distributions with parameters $(F_{\pi,f})$ and mixture weights proportional to $p(\mu \mid \theta_{\pi,f})$. This structure allows using the main result of Kent [1983], which grants that the von Mises-Fisher densities given by the $F_{\pi,f}$ are linearly independent. This result can be combined with the main theorem of Yakowitz and Spragins [1968], which states that a family of finite mixtures is identifiable if and only if the mixture components form a linearly independent set.

As a consequence, we identify the parameter F up to a column permutation and change of sign. Moreover, in the sum above, the probabilities $p(\mu \mid \theta_{\pi,f})$ with maximal amplitude are given by $\pi = \mathrm{Id}$, and all the other permutations such that for all i, $\mu_{\sigma(i)} = \mu_i$ (which encompasses eigenvalue multiplicity). Since we assumed that all concentration parameters are positive, all $(F_{\pi,f})$ are distinct and hence the maximal mixture weights correspond to the matrices $(F_{\pi,f})$ with π as just described. This finally allows matching eigenvalues with eigenvectors, completing the identification of θ.

□

6.4 Existence and Consistency of the MAP Estimator

6.4.1 Maximum A Posteriori *versus* Maximum Likelihood

We turn to the problem of estimating θ from samples $A_1, \ldots, A_N$ when the number of samples N grows large. In this section, we assume that the samples are distributed according to a distribution P, which may not be of the form $p(A \mid \theta)$.

However, the MLE may not be defined: the optimal value for F may theoretically be infinite, as the model likelihood does not necessarily decrease at infinity. For instance, if the samples $A_1, \ldots, A_N$ are drawn from a Gaussian distribution with i.i.d. coefficients and mean equal to a rank p matrix $\lambda_0 \cdot X_0$, the parameters σ_λ and s_i tend to take extreme values (σ_λ being very small and s_i being very large), and the distribution of latent variables is highly concentrated around (λ_0, X_0). This phenomenon occurs because the estimated data distribution asymptotically converges to the true data distribution, which lies at the boundary of the model family (in the sense that taking very large s_i's and a very small σ_λ yields a distribution close to the true data distribution).

This problem is overcome numerically by adding a prior distribution $p(\theta)$ and considering the Maximum A Posteriori (MAP) estimator over the set Θ of all parameters:

$$\hat{\theta}_N \in \mathrm{argmax}_\Theta \, p(\theta \mid A_1, \ldots A_N) = \mathrm{argmax}_\Theta \, p(A_1, \ldots A_N \mid \theta) p(\theta) \,.$$

In this section, we want to account for the possible convergence of latent variable distributions to constant values. For this purpose, instead of the parameterization $\theta = (F, \mu, \sigma_\lambda, \sigma_\varepsilon)$, we will be defining the parameter set by $\Theta = \{\theta = (M, s, \mu, \sigma_\lambda, \sigma_\varepsilon) \mid \sigma_\lambda > 0, s_i < +\infty\}$, with the equivalence given by $F = M\mathrm{Diag}(s)$. In the next section, this representation will allow us to formally consider an extension of the set Θ accounting for the case where $s_i = +\infty$ and $\sigma_\lambda = 0$.

We consider inverse Gamma distributions for the prior $p(\sigma_\lambda, \sigma_\varepsilon)$, the uniform distribution over $\mathcal{V}_{np}$ for M, and any p.d.f. decreasing at infinity for $p(s)$ and $p(\mu)$. Unlike the MLE, with this prior specification the MAP estimator is guaranteed to exist.

Theorem 3. *Given the proposed model, with parameters following the prior distribution described above, for any set of matrices $(A_i)_{i=1}^N$, there exists $\hat{\theta}_N \in \mathrm{argmax}_{\theta \in \Theta} \, p(\theta \mid A_1, ..., A_N)$.*

Proof. The bound obtained in Lemma 7 gives with Bayes' formula:

$$\log p(\theta \mid A) \leq -\frac{n^2}{2}\log(2\pi) - (n^2 - p)\log\sigma_\varepsilon - p\log\sigma_\lambda + \log p(\theta) - \log p(A)\,.$$

Since

$$\begin{cases} p(\sigma_\lambda) = \frac{\beta_\lambda^{\alpha_\lambda}}{\Gamma(\alpha_\lambda)}(1/\sigma_\lambda)^{\alpha_\lambda+1}\exp(-\beta_\lambda/\sigma_\lambda) \\ p(\sigma_\varepsilon) = \frac{\beta_\varepsilon^{\alpha_\varepsilon}}{\Gamma(\alpha_\varepsilon)}(1/\sigma_\varepsilon)^{\alpha_\varepsilon+1}\exp(-\beta_\varepsilon/\sigma_\varepsilon)\,, \end{cases}$$

and given the other assumptions on the prior distribution, we have $\log p(\theta \mid A) \to -\infty$ as any of the model variables reaches an open boundary of its domain. Furthermore, the function $\log p(\theta \mid A)$ is smooth: the integral representation given by Lemma 6 writes as

$$p(A \mid \theta) = \frac{1}{(2\pi)^{n^2/2}\sigma_\varepsilon^{n^2}}\frac{1}{C(F)}\frac{\sigma_p^p}{\sigma_\lambda^p}\exp\left[-\frac{1}{2\sigma_\varepsilon^2}\|A\|_F^2 - \frac{1}{2\sigma_\lambda^2}\|\mu\|^2\right]\int_{\mathcal{V}_{np}}\exp\left[\langle F, X\rangle + \frac{1}{2\sigma_p^2}\|\mu_{AX}\|^2\right]\mathrm{d}X\,.$$

Since the manifold $\mathcal{V}_{np}$ is compact and the integrand $f(\theta, X) = \exp(\langle F, X\rangle + \|\mu_{AX}\|^2/2\sigma_p^2)$ is smooth on $\Theta \times \mathcal{V}_{np}$, classical integration theorems grant that $\log p(\theta \mid A)$ is smooth over every compact subset of Θ: given a compact set K, the domination function $g(X) = \max_{\theta\in K} f(X, \theta)$ is smooth over K. Hence, $\log p(\theta \mid A)$ is smooth over Θ. In particular, the function $\log p(A \mid \theta)$ is coercive and continuous, and it thus admits a maximizer over Θ. □

6.4.2 MAP Consistency

The above result motivates the study of the MAP estimator over the MLE. However, although adding a prior distribution grants the existence of a maximizer within Θ, the weight of the prior term decreases as the number of samples grows large, and we should expect the MAP estimator to diverge to the boundary of Θ for some empirical data distributions P. This phenomenon is accounted for by considering an extended set of parameters Θ^∞ allowing null eigenvector variance (i.e., λ constant) and infinite von Mises-Fisher concentrations (i.e., x_i constant for some i's):

$$\Theta^\infty = \{(M, s, \mu, \sigma_\lambda, \sigma_\varepsilon) \mid \sigma_\lambda \in [0, +\infty), s_i \in [0, +\infty]\}\,.$$

We prove in Lemma 8 that the likelihood $p(A \mid \theta)$ extends continuously to Θ^∞. The extension essentially amounts to considering eigenvalue and eigenvector distributions restricted to a conditional subspace. With this convention, the objective function ℓ to be asymptotically maximized can be defined over Θ^∞ as the almost sure (a.s.) limit of the empirical objective function $\frac{1}{N}\sum_{i=1}^N \log p(A_i \mid \theta) + \frac{1}{N}p(\theta)$ defined over Θ:

$$\ell(\theta) = \mathbb{E}_{P(\mathrm{d}A)}[\log p(A \mid \theta)]\,.$$

If P has a density with respect to the Lebesgue measure, the function ℓ is equal, up to a constant term which depends only on P, to the opposite of the Kullback-Leibler divergence between P and $p(A \mid \theta)$. The MAP estimator is said to be *consistent* if it converges to the set Θ_* of maximizers of $\ell(\theta)$. In the case where P corresponds to some $p(A \mid \theta^*)$ for $\theta^* \in \Theta^{\mathrm{id}}$, ℓ only has one maximizer, which is the true model parameter θ^*. For large classes of sufficiently regular families of statistical models, the MLE and the MAP can be proved to be consistent and, in probability, to minimize the KL divergence to the optimal point [van der Vaart, 1998].

The consistency of MLE for latent variable models has been studied for several classes of models, like Hidden Markov Models [Douc et al., 2011], Independent Component Analysis [Bonhomme and Robin, 2009] or longitudinal mixed effects models [Chevallier et al., 2021, Allassonnière et al., 2007]. Along these results, we obtain the almost sure (a.s.) consistency of the MAP estimator. We study two particular cases: in the first case, we assume that the parameters which may diverge stay bounded, and obtain a.s. convergence to the set of maximizers over the constrained set. In the second case, we show that the unconstrained MAP estimator converges a.s. to the set of maximizers over Θ^∞.

The convergence to the set of maximizers of $\ell(\theta)$ is quantified by the distance $d(\hat{\theta}_N, \Theta_*)$. However, the set Θ_*^∞ of maximizers of ℓ over Θ^∞ may have some elements with infinite coordinates,

which prevents from quantifying distances. To overcome this issue, we consider the reparameterization $\xi(\theta) = (M, h(s), \mu, \sigma_\lambda, \sigma_\varepsilon)$, with $h : [0, +\infty]^p \to [0, 1]^p$ applying the same continuous increasing transformation to each s_i, for instance $h(s)_i = \operatorname{atanh}(s_i)$. Over the new parameter space $\Xi^\infty = \xi(\Theta^\infty)$, we also obtain the almost sure consistency of the MAP $\hat{\xi}_N = \xi(\hat{\theta}_N)$.

Theorem 4. *Let Θ^η be the set of parameters with each s_i and σ_λ^{-1} upper bounded by η, and let Θ_*^η be the set of maximizers of ℓ over Θ^η. Consider the following hypotheses:*

H1 *The number of latent patterns is strictly lower than the number of nodes: $p < n$.*

H2 *The samples $(A_i)_{i=1}^N$ are independent and identically distributed.*

H3 *The true data distribution $P(\mathrm{d}A)$ has a density w.r.t. the Lebesgue measure and exponentially decaying tails beyond a compact set: there exist $a, b > 0$, such that for x large enough, $\sup_{\|A\|_F \geq x} P(A) \leq a \exp(-bx)$.*

*Then, assuming **H1**, **H2** and **H3**:*

1. *For all $\eta > 0$, $\Theta_*^\eta \neq \varnothing$ and the MAP estimator $\hat{\theta}_N^\eta$ on Θ^η is consistent: for every continuous metric δ, almost surely,*
$$\delta(\hat{\theta}_N^\eta, \Theta_*^\eta) \underset{N \to +\infty}{\longrightarrow} 0\,.$$

2. *The extended set of maximizers is non-empty: $\Theta_*^\infty \neq \varnothing$. Denoting $\Xi_*^\infty = \xi(\Theta_*^\infty)$, for every continuous metric δ, almost surely,*
$$\delta(\hat{\xi}_N, \Xi_*^\infty) \underset{N \to +\infty}{\longrightarrow} 0\,.$$

Remark. As a consequence, if all the elements of Θ^∞ are equal on a coordinate, the corresponding coordinate of $\hat{\theta}_N$ converges to this value. In particular, for some distributions P we may have $s_i \to +\infty$ or $\sigma_\lambda \to 0$ almost surely. This explains the phenomenon observed in the previous section on Gaussian empirical data distributions.

The proof follows the architecture of van der Vaart [1998], Chevallier et al. [2021]. The main difficulties and specificities lie in the proofs of the required lemmas which are specific to the model, and the possibility of having partially constant latent variable distributions. We thus only present here the structure of the main proof, and refer the reader to Appendix 6.C for the detailed argument. As the proof for the first assertion is a strictly simpler version of the proof of the second assertion, we omit it for the sake of brevity.

Sketch of the proof. The proof is divided into four parts. We define

$$\mathbb{E}^* = \sup_{\theta \in \Theta^\infty} \mathbb{E}_{P(\mathrm{d}A)}[\log p(A \mid \theta)] \quad \text{and} \quad K_\varepsilon = \{\theta \in \overline{\Theta^\infty} \mid \delta(\xi(\theta), \Xi_*^\infty) \geq \varepsilon\}\,,$$

with $\overline{\Theta^\infty}$ the Alexandrov compactification of Θ^∞, as detailed in Appendix 6.C.

A) We prove that, for all $\theta_\infty \in \overline{\Theta^\infty}$ such that $\delta(\xi(\theta_\infty), \Xi_*^\infty) \geq \varepsilon$, there exists an open neighborhood $\mathcal{U} \subset \overline{\Theta^\infty}$ of θ_∞ such that

$$\mathbb{E}_{P(\mathrm{d}A)}\left[\sup_{\theta \in \mathcal{U} \cap \Theta^\infty} \log p(A \mid \theta)\right] < \mathbb{E}^*\,.$$

B) The set K_ε described above is compact, and therefore among all the sets $\mathcal{U}$ defined in part A we can extract a finite cover of K_ε. This allows proving that

$$\limsup_{N \to +\infty} \sup_{\theta \in K_\varepsilon \cap \Theta^\infty} \frac{1}{N} \sum_{i=1}^N \log p(A_i \mid \theta) < \mathbb{E}^*\,.$$

C) Using the definition of $\hat{\theta}_N$ and the law of large numbers, we show that

$$\liminf_{N\to+\infty} \frac{1}{N}\sum_{i=1}^{N} \log p(A_i \mid \hat{\theta}_N) \geq \mathbb{E}^* \,.$$

D) Finally, combining the two arguments above allows getting a contradiction if $\hat{\theta}_N \in K_\varepsilon$ for an infinite number of N. As a consequence, for all $\varepsilon > 0$, $\hat{\theta}_N \notin K_\varepsilon$ almost surely as $N \to +\infty$, which gives precisely $\delta(\hat{\xi}_N, \Xi_*^\infty) \to 0$.

□

6.5 Asymptotic Normality of the MAP Estimator

A consequence of Theorem 2 is that, if the empirical data distribution P corresponds to $p(A \mid \theta_0)$ for some $\theta_0 \in \Theta^{\mathrm{id}}$, we have $\Theta_*^{\mathrm{id}} = \{\theta_0\}$: thus, by Theorem 4, the MAP estimator over Θ^{id} converges almost surely to θ_0. A classical question is then to establish the rate of convergence of $\hat{\theta}_N$ toward θ_0, as well as the limiting asymptotic distribution. An answer for the more general case of M and Z-estimators is provided in Chapter 5 of van der Vaart [1998], which we restate with adapted notations:

Theorem 5 (Theorem 5.23 in van der Vaart [1998])**.** *Let $m_\theta(A) = \log p(A \mid \theta)$. Assume that m_θ is a measurable function such that $\theta \mapsto m_\theta(A)$ is differentiable at θ_0 for P-almost every A with derivative $\nabla_A m_{\theta_0}(A)$. Assume that there exists a function $\overline{m}$ with $\mathbb{E}_{P(\mathrm{d}A)}[\overline{m}(A)^2] < +\infty$, such that, for every θ_1 and θ_2 in a neighborhood of θ_0:*

$$|m_{\theta_1}(A) - m_{\theta_2}(A)| \leq \overline{m}(A)\,\|\theta_1 - \theta_2\| \,. \tag{6.7}$$

Furthermore, assume that the map $\ell(\theta) = \mathbb{E}_{P(\mathrm{d}A)}[m_\theta(A)]$ admits a second-order Taylor expansion at a point of maximum θ_0 with nonsingular symmetric second derivative $V = \nabla^2\ell(\theta_0)$. If

$$\frac{1}{N}\sum_{i=1}^{N} m_{\hat{\theta}_N}(A_i) \geq \sup_{\theta\in\Theta} \frac{1}{N}\sum_{i=1}^{N} m_\theta(A_i) - o_{\mathbb{P}}(1/N) \tag{6.8}$$

and $\hat{\theta}_N \to \theta_0$ in probability, then

$$\sqrt{N}(\hat{\theta}_N - \theta_0) = -V^{-1}\frac{1}{\sqrt{N}}\sum_{i=1}^{N} \nabla_\theta m_{\theta_0}(A_i) + o_{\mathbb{P}}(1)\,.$$

In particular, the sequence $\sqrt{N}(\hat{\theta}_N - \theta_0)$ is asymptotically normal with mean zero and covariance matrix $V^{-1}\mathbb{E}_{P(\mathrm{d}A)}[\nabla_\theta m_\theta(A)\nabla_\theta m_\theta(A)^\top]V^{-1}$.

Remark. The notation $o_{\mathbb{P}}(1/N)$ designates a random variable Z_N such that $NZ_N \to 0$ in probability.

The most important condition in the theorem above is the non singularity of the Hessian matrix at θ_0. In general, $\nabla^2_\theta\ell(\theta)$ is impossible to compute for latent variable models, as it involves the Hessian of $\log p(A \mid \theta)$. However, the problem gets more tractable when the data distribution P corresponds to $p(A \mid \theta_0)$ for some $\theta_0 \in \Theta$. The Hessian matrix at θ_0 then classically rewrites as the Fisher information matrix $I(\theta_0)$ (see for instance Lemma 5.3 in Lehmann and Casella [2003]):

$$\nabla^2_\theta\ell(\theta_0) = \mathbb{E}_{p(A|\theta_0)}[\nabla^2_\theta \log p(A \mid \theta_0)] = -\mathbb{E}_{p(A|\theta_0)}\left[(\nabla_\theta \log p(A \mid \theta_0))(\nabla_\theta \log p(A \mid \theta_0))^\top\right] = -I(\theta_0)\,.$$

The non-singularity of the Fisher information matrix remains difficult to prove for general latent variable models. Some papers consider it as a base hypothesis to obtain the asymptotic normality, e.g., for Factor Analysis [Anderson and Amemiya, 1988] or Hidden Markov Models [Bickel et al., 1998]. In the latter case, the more recent work of Douc [2005] provided a condition to obtain the non-singularity of the Fisher information matrix. A recent result was obtained by Ren et al. [2015]

on the asymptotic normality of MLE for Gaussian graphical models and apply it to estimation from partial observations. In this specific case, the Fisher information has a simple closed form expression.

For the model considered here, no closed form expression can be expected, as the gradient of the log-likelihood writes with integrals on $\mathcal{V}_{np}$. Instead, we notice that, since the density $p(A \mid \theta_0)$ is continuous and $I(\theta_0)$ writes as the expectation of $(\nabla_\theta \log p(A \mid \theta_0))(\nabla_\theta \log p(A \mid \theta_0))^\top$, the matrix will be non-singular if we can find $\dim(\theta_0)$ matrices A_i such that the related gradients $\nabla_\theta \log p(A_i \mid \theta_0)$ are linearly independent. This is formalized in the following lemma:

Lemma 4. *Let $d = \dim(\theta_0)$. If $A_1, \dots, A_d$ matrices can be found such that the related gradients $\nabla_\theta \log p(A_i \mid \theta_0)$ are independent, then $I(\theta_0)$ is positive definite.*

Proof. Let $x \in \mathbb{R}^d$. We have:

$$x^\top I(\theta_0) x = \mathbb{E}_{p(A|\theta_0)} \left[\langle x, \nabla_\theta \log p(A \mid \theta_0) \rangle^2 \right] \geq 0 \,.$$

If $x^\top I(\theta_0) x = 0$, then $\langle x, \nabla_\theta \log p(A \mid \theta_0) \rangle^2$ must be zero everywhere. Therefore, since the map $\theta \mapsto \log p(A \mid \theta)$ is infinitely smooth, x is orthogonal to all the gradients $\nabla_\theta \log p(A_i \mid \theta_0)$, and thus to their linear span, which covers the full space, which implies $x = 0$. As a consequence, $I(\theta_0)$ is positive definite. □

In the case of our model, it turns out that, although the expression of $\nabla_\theta \log p(A \mid \theta)$ is intractable, it simplifies as $\|A\|_F$ grows large. This simplification comes from the so-called Fisher identity:

$$\begin{aligned}
\nabla_\theta \log p(A \mid \theta) &= \frac{1}{p(A \mid \theta)} \nabla_\theta p(A \mid \theta) \\
&= \frac{1}{p(A \mid \theta)} \iint_{\mathcal{V}_{np} \times \mathbb{R}^p} \nabla_\theta p(A, X, \lambda \mid \theta) \, [\mathrm{d}X] \mathrm{d}\lambda \\
&= \iint_{\mathcal{V}_{np} \times \mathbb{R}^p} \nabla_\theta \log p(A, X, \lambda \mid \theta) \cdot p(X, \lambda \mid A, \theta) \, [\mathrm{d}X] \mathrm{d}\lambda \\
&= \mathbb{E}[\nabla_\theta \log p(A, X, \lambda) \mid A] \,,
\end{aligned}$$

and the gradient rewrites as an expectation of the complete log-likelihood over the latent variables. Given the complete expression

$$p(A, X, \lambda \mid \theta) = \frac{1}{(2\pi)^{n^2/2} \sigma_\varepsilon^{n^2}} \frac{1}{(2\pi)^{p/2} \sigma_\lambda^p} \frac{1}{\mathcal{C}(F)} \exp\left[\langle F, X \rangle_F - \frac{1}{2\sigma_\varepsilon^2} \|A - \lambda \cdot X\|_F^2 - \frac{1}{2\sigma_\lambda^2} \|\lambda - \mu\|^2 \right] ,$$

this expectation yields for instance:

$$\nabla_F \log p(A \mid \theta) = -\nabla_F \log \mathcal{C}(F) + \mathbb{E}[X \mid A] \,.$$

As $\|A\|_F \to +\infty$, we show in the upcoming Proposition 5 that the eigenvector distribution of $(X \mid A)$ concentrates around the permutations of the p eigenvectors of A related to the p largest eigenvalues. As a consequence, $\nabla_F \log p(A \mid \theta)$ writes as the sum of $\nabla_F \log \mathcal{C}(F)$ and a linear combination of all $(X_A)_{\pi,f}$, with X_A the $n \times p$ eigenvector matrix of A and $\pi \in S_p, f \in \{\pm 1\}^p$. However, although X_A can be chosen freely, the subsequent linear combination turns out to be hard to compute and manipulate, which ultimately prevents from getting an explicit expression for the gradient in F. The same phenomenon happens with the other gradients, which all rely on an expectation given A.

This observation motivates the main hypothesis for our normality result. We shall consider a **restricted variant of the main model** $\tilde{p}(A, X, \lambda)$, where the X variable is constrained to the set Δ_0 defined in Equation (6.6): the density of X writes as

$$\tilde{p}(X \mid \theta) = \frac{\mathbf{1}_{X \in \Delta_0}}{\mathcal{C}'(F)} \exp\left(\langle X, F \rangle_F \right) , \tag{6.9}$$

with $\mathcal{C}'(F) = \int_{\Delta_0} \exp\left(\langle X, F \rangle_F \right) [\mathrm{d}X]$. This constraint does not fundamentally change the model in the limits where $s_i \to 0$ and $s_i \to +\infty$. For intermediate values, it truncates the other sections $\Delta_{\pi,f}$

of the vMF distribution, but does not change the support of the distribution of $\lambda \cdot X$, as it still covers the set of rank p matrices. The resemblance between p and $\tilde{p}$ is optimized when the maximum of $\langle X, F \rangle_F$ is reached in Δ_0, i.e., when choosing the normalized columns of F to be in Δ_0. We adopt this convention in the remainder of the section, as it also facilitates proving the identifiability of the restricted model.

In the remainder of this section, the notations $\ell(\theta)$, $\hat{\theta}_N$, ... refer to densities and estimators obtained for the restricted model. We also assume that the empirical data distribution is given by $\tilde{p}(A \mid \theta_0)$ rather than $p(A \mid \theta_0)$. With this restricted model, we have the following result:

Proposition 5. *Let $A \in \mathbb{R}^{n \times n}$ with rank at least p and distinct eigenvalues, and let $A_t = tA$ for $t \in \mathbb{R}$. On the restricted model with $X \in \Delta_0$, the distribution $(X \mid A = A_t)$ converges to the constant value X_A, with $X_A \in \Delta_0$ the list of eigenvectors of A corresponding to the p largest eigenvalues. In particular, $\mathbb{E}[X \mid A = A_t]$ converges to X_A.*

Proof. By definition, $\mathbb{E}[X \mid A = A_t]$ is the expectation of X w.r.t. the probability density proportional to

$$\mathbf{1}_{X \in \Delta_0} \exp\left(\langle X, F \rangle + \frac{1}{2\sigma_p^2} \left\| \mu_{tA,X} \right\|^2 \right).$$

As $t \to +\infty$, the function $g_t(X) = \frac{1}{2\sigma_p^2} \left\| \mu_{tA,X} \right\|^2$ reaches its maximum to a point which converges to X_A. We have indeed:

$$g_t(X) = \left\| \mu_{tA,X} \right\|^2 = t^2 \frac{\sigma_p^2}{2\sigma_\varepsilon^4} \left\| A * X \right\|^2 + t \frac{\sigma_p^2}{\sigma_\lambda^2 \sigma_\varepsilon^2} \langle A * X, \mu \rangle + \frac{\sigma_p^2}{\sigma_\lambda^4} \left\| \mu^2 \right\|.$$

By Proposition 4, $\left\| A * X \right\|^2$ is only at $X = X_A$ on Δ_0 (unicity is guaranteed as the eigenvalues of A are distinct). Let D a region of $\mathcal{V}_{np}$ with non-zero invariant measure such that $X_A \notin \overline{D}$ and let $\eta > 0$ such that if $X \in D$ then $\left\| A * X \right\|^2 \leq \left\| A * X_A \right\|^2 - 2\eta$. Let B_η be a neighborhood of X_A such that $\left\| A * X \right\| \geq \left\| A * X_A \right\| - \eta$. We have:

$$\begin{aligned}
\mathbb{P}(X \in D \mid A = A_t, \theta) &= \frac{\int_D \exp\left(\langle X, F \rangle + g_t(X) \right) [\mathrm{d}X]}{\int_{\mathcal{V}_{np}} \exp\left(\langle X, F \rangle + g_t(X) \right) [\mathrm{d}X]} \\
&\leq \frac{\int_D \exp\left(\langle X, F \rangle + g_t(X) \right) [\mathrm{d}X]}{\int_{B_\eta} \exp\left(\langle X, F \rangle + g_t(X) \right) [\mathrm{d}X]} \\
&\leq \frac{\int_D \exp\left(\langle X, F \rangle + t^2 \frac{\sigma_p^2}{2\sigma_\varepsilon^4} (\left\| A * X_A \right\|^2 - 2\eta) + t \frac{\sigma_p^2}{\sigma_\lambda^2 \sigma_\varepsilon^2} \left\| A * X_A \right\| \left\| \mu \right\| \right) [\mathrm{d}X]}{\int_{B_\eta} \exp\left(\langle X, F \rangle + t^2 \frac{\sigma_p^2}{2\sigma_\varepsilon^4} (\left\| A * X_A \right\|^2 - \eta) - t \frac{\sigma_p^2}{\sigma_\lambda^2 \sigma_\varepsilon^2} \left\| A * X_A \right\| \left\| \mu \right\| \right) [\mathrm{d}X]} \\
&\leq \exp\left(2 \left\| F \right\|_* - 2t^2 \eta \frac{\sigma_p^2}{2\sigma_\varepsilon^4} + 2t \frac{\sigma_p^2}{\sigma_\lambda^2 \sigma_\varepsilon^2} \left\| A * X_A \right\| \left\| \mu \right\| \right) \frac{|D|_{\mathcal{V}_{np}}}{|B_\eta|_{\mathcal{V}_{np}}} \\
&\underset{t \to +\infty}{\longrightarrow} 0.
\end{aligned}$$

Hence, by the Portmanteau theorem, the sequence of probability distributions $(X \mid A = A_t)$ converges in distribution to the constant X_A. □

Remark. Proposition 5 can be compared to the decreasing uncertainty on the normalized position $x / \left\| x \right\|$ of a point x going to infinity. If we used the complete model, the distribution of X would instead converge to the sum of Diracs at $(X_A)_{\pi,f}$ weighted by $p((X_A)_{\pi,f} \mid \theta)$.

With the result above, we can prove that $\dim \Theta$ linearly independent gradients $\nabla_\theta \log p(A \mid \theta_0)$ can be obtained.

Lemma 5. *For all $\theta \in \Theta$, the log-likelihood gradient map $A \mapsto \nabla_\theta \log \tilde{p}(A \mid \theta)$ of the restricted model takes $\dim \Theta = np + p + 2$ linearly independent values.*

Proof. As explained above, the Fisher identity reminded here allows computing gradients as A grows large:

$$\nabla_\theta \log \tilde{p}(A \mid \theta) = \mathbb{E}[\nabla_\theta \log \tilde{p}(A, X, \lambda \mid \theta) \mid A].$$

In order to alleviate the notations, the expectations $\mathbb{E}$ below refer to the distribution $\tilde{p}(A, X, \lambda \mid \theta_0)$. Since we have:

$$\tilde{p}(A, X, \lambda \mid \theta) = \frac{1}{(2\pi)^{n^2/2}\sigma_\varepsilon^{n^2}} \frac{1}{(2\pi)^{p/2}\sigma_\lambda^p} \frac{\mathbf{1}_{X \in \Delta_0}}{\mathcal{C}'(F)} \exp\left[\langle F, X\rangle - \frac{1}{2\sigma_\varepsilon^2}\left\|A - \lambda \cdot X\right\|_F^2 - \frac{1}{2\sigma_\lambda^2}\left\|\lambda - \mu\right\|^2\right],$$

we thus get for $(F, \mu, \sigma_\lambda, \sigma_\varepsilon)$:

1. $\nabla_F \log \tilde{p}(A \mid \theta) = -\nabla_F \log C(F) + \mathbb{E}[X \mid A]$,
2. $\nabla_\mu \log \tilde{p}(A \mid \theta) = \frac{1}{\sigma_\lambda^2}\left[\mathbb{E}[\lambda \mid A] - \mu\right]$,
3. $\nabla_{\sigma_\varepsilon^2} \log \tilde{p}(A \mid \theta) = -\frac{n^2}{2\sigma_\varepsilon^2} + \frac{1}{2\sigma_\varepsilon^4}\mathbb{E}[\left\|A - \lambda \cdot X\right\|_F^2 \mid A]$,
4. $\nabla_{\sigma_p^2} \log \tilde{p}(A \mid \theta) = -\frac{p}{2\sigma_\lambda^2} + \frac{1}{2\sigma_\lambda^4}\mathbb{E}[\left\|\lambda - \mu\right\|^2 \mid A]$.

Let $t \in \mathbb{R}$, consider the matrix $A_t = tA$ and denote $X_A \in \Delta_0$ the matrix of eigenvectors of A for the p largest eigenvalues. The expressions above simplify as $t \to +\infty$:

1. *For F*: Proposition 5 gives for A with p distinct non-zero leading eigenvalues:

$$\nabla_F \log \tilde{p}(A_t \mid \theta) \to -\nabla_F \log C(F) + X_A\,.$$

2. *For μ*: as seen in Section 6.2.3, $(\lambda \mid X, A_t) \sim \mathcal{N}(\mu_{A_t X}, \sigma_p^2)$, so that we have

$$\begin{aligned}
\frac{1}{t}\nabla_\mu \log \tilde{p}(A_t \mid \theta) &= \frac{1}{t}\frac{1}{\sigma_\lambda^2}\left[\mathbb{E}[\mathbb{E}[\lambda \mid X, A_t] \mid A] - \mu\right] \\
&= \frac{1}{t}\frac{1}{\sigma_\lambda^2}\left[\mathbb{E}[\mu_{A_t X} \mid A_t] - \mu\right] \\
&= \frac{1}{t}\frac{1}{\sigma_\lambda^2}\left[\mathbb{E}\left[\frac{\sigma_p^2}{\sigma_\varepsilon^2} tA * X + \frac{\sigma_p^2}{\sigma_\lambda^2}\mu \mid A_t\right] - \mu\right] \\
&\underset{t\to+\infty}{\longrightarrow} \frac{\sigma_p^2}{\sigma_\lambda^2\sigma_\varepsilon^2} A * X_A\,.
\end{aligned}$$

3. *For σ_ε^2*: similarly, we get

$$\begin{aligned}
\frac{1}{t^2}\nabla_{\sigma_\varepsilon^2} \log \tilde{p}(A_t \mid \theta) &= -\frac{n^2}{2t^2\sigma_\varepsilon^2} + \frac{1}{2t^2\sigma_\varepsilon^4}\mathbb{E}[\mathbb{E}[\left\|A_t - \lambda \cdot X\right\|_F^2 \mid X, A_t] \mid A_t] \\
&= -\frac{n^2}{2t^2\sigma_\varepsilon^2} + \frac{1}{2t^2\sigma_\varepsilon^4}\mathbb{E}[\mathbb{E}[\left\|tA\right\|_F^2 - 2\langle \lambda, tA * X\rangle + \left\|\lambda\right\|^2 \mid X, A_t] \mid A_t] \\
&= -\frac{n^2}{2t^2\sigma_\varepsilon^2} + \frac{1}{2t^2\sigma_\varepsilon^4}\mathbb{E}[\left\|tA\right\|_F^2 - 2\langle \mu_{A_t X}, tA * X\rangle + \left\|\mu_{A_t X}\right\|^2 + p\sigma_p^2 \mid A_t] \\
&\underset{t\to+\infty}{\longrightarrow} \frac{1}{2\sigma_\varepsilon^4}\left[\left\|A\right\|_F^2 - 2\frac{\sigma_p^2}{\sigma_\varepsilon^2}\left\|A * X_A\right\|^2 + \frac{\sigma_p^4}{\sigma_\varepsilon^4}\left\|A * X_A\right\|^2\right].
\end{aligned}$$

4. *For σ_λ^2*:

$$\begin{aligned}
\frac{1}{t^2}\nabla_{\sigma_p^2} \log \tilde{p}(A_t \mid \theta) &= -\frac{p}{2t^2\sigma_\lambda^2} + \frac{1}{2t^2\sigma_\lambda^4}\mathbb{E}[\left\|\lambda - \mu\right\|^2 \mid A_t] \\
&= -\frac{p}{2t^2\sigma_\lambda^2} + \frac{1}{2t^2\sigma_\lambda^4}\mathbb{E}[\mathbb{E}[\left\|\lambda - \mu\right\|^2 \mid X, A_t] \mid A_t] \\
&= -\frac{p}{2t^2\sigma_\lambda^2} + \frac{1}{2t^2\sigma_\lambda^4}\mathbb{E}[\left\|\mu_{A_t X} - \mu\right\|^2 + p\sigma_p^2 \mid A_t] \\
&\underset{t\to+\infty}{\longrightarrow} \frac{\sigma_p^4}{2\sigma_\lambda^4\sigma_\varepsilon^4}\left\|A * X_A\right\|^2\,.
\end{aligned}$$

In these expressions, $A * X_A = (\lambda_1, \dots, \lambda_p)$ is the p leading eigenvalues of A. Furthermore, we have $\|A * X_A\|_F^2 = \lambda_1^2 + \dots + \lambda_p^2$ and $\|A\|_F^2 = \lambda_1^2 + \dots + \lambda_n^2$. In the remainder of the proof, we call these asymptotic rescaled values *limit gradients*. Using the formulas above, we derive the following limit gradients.

- Taking $X \in \mathcal{V}_{np}$, we consider the limit gradient for $\mu \cdot X^i$. Up to factors t and t^2 which do not affect the linear independence, the result is:

$$\begin{pmatrix} -\nabla_F \log C(F) + X \\ \frac{\sigma_p^2}{\sigma_\lambda^2 \sigma_\varepsilon^2} \mu \\ \frac{1}{2\sigma_\varepsilon^4} \left[1 - 2\frac{\sigma_p^2}{\sigma_\varepsilon^2} + \frac{\sigma_p^4}{\sigma_\varepsilon^4}\right] \|\mu\|^2 \\ \frac{\sigma_p^4}{2\sigma_\lambda^4 \sigma_\varepsilon^4} \|\mu\|^2 \end{pmatrix}.$$

- Taking a vector $\lambda \in \mathbb{R}^p$ such that $\|\lambda\| = \|\mu\|$, we consider matrices of the form $\lambda \cdot I_{np}$. The resulting limit gradient at $t \to +\infty$ is:

$$\begin{pmatrix} -\nabla_F \log C(F) + I_{np} \\ \frac{\sigma_p^2}{\sigma_\lambda^2 \sigma_\varepsilon^2} \lambda \\ \frac{1}{2\sigma_\varepsilon^4} \left[1 - 2\frac{\sigma_p^2}{\sigma_\varepsilon^2} + \frac{\sigma_p^4}{\sigma_\varepsilon^4}\right] \|\mu\|^2 \\ \frac{\sigma_p^4}{2\sigma_\lambda^4 \sigma_\varepsilon^4} \|\mu\|^2 \end{pmatrix}.$$

- We consider the matrices $A = \mathrm{Diag}(\mu_1, \dots, \mu_p, \alpha, \dots, \alpha)$ with $0 < \alpha < \min_i |\mu_i|$. The resulting limit gradient is:

$$\begin{pmatrix} -\nabla_F \log C(F) + I_{np} \\ \frac{\sigma_p^2}{\sigma_\lambda^2 \sigma_\varepsilon^2} \mu \\ \frac{1}{2\sigma_\varepsilon^4} \left[1 - 2\frac{\sigma_p^2}{\sigma_\varepsilon^2} + \frac{\sigma_p^4}{\sigma_\varepsilon^4}\right] \|\mu\|^2 + \frac{1}{2\sigma_\varepsilon^4} \alpha^2 (n-p)^2 \\ \frac{\sigma_p^4}{2\sigma_\lambda^4 \sigma_\varepsilon^4} \|\mu\|^2 \end{pmatrix}.$$

- Finally, we take the matrix $\mu \cdot I_{np}$. The resulting limit gradient is:

$$\begin{pmatrix} -\nabla_F \log C(F) + I_{np} \\ \frac{\sigma_p^2}{\sigma_\lambda^2 \sigma_\varepsilon^2} \mu \\ \frac{1}{2\sigma_\varepsilon^4} \left[1 - 2\frac{\sigma_p^2}{\sigma_\varepsilon^2} + \frac{\sigma_p^4}{\sigma_\varepsilon^4}\right] \|\mu\|^2 \\ \frac{\sigma_p^4}{2\sigma_\lambda^4 \sigma_\varepsilon^4} \|\mu\|^2 \end{pmatrix}.$$

Subtracting the limit gradient at $\mu \cdot I_{np}$, we can get linear combinations of gradients arbitrarily close to any vector of the forms:

$$\begin{pmatrix} X - I_{np} \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \frac{\sigma_p^2}{\sigma_\lambda^2 \sigma_\varepsilon^2} (\lambda - \mu) \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ \frac{1}{2\sigma_\varepsilon^4} \alpha^2 (n-p)^2 \\ 0 \end{pmatrix}.$$

With $X \in \mathcal{V}_{np}$ and $\|\lambda\| = \|\mu\|$. We can now use Lemma 13, which states that the elements of the form $X - I_{np}$ span $\mathbb{R}^{np}$. Similarly, elements of the form $\lambda - \mu$ span $\mathbb{R}^p$: taking $\lambda = -\mu$, the space contains -2μ and thus μ. Hence, it also contains all elements λ with norm $\|\mu\|$, which can be rescaled to get the entire space. As a consequence, the three vector families above span the entire linear hyperplan $\mathbb{R}^{np} \times \mathbb{R}^p \times \mathbb{R} \times \{0\}$. Furthermore, the limit gradient at $\mu \cdot I_{np}$ has a non-zero last coordinate and does not belong to this linear hyperplan. As a consequence, the set of all limit gradients spans the entire gradient space. Therefore, the set of all gradients, which gets arbitrarily close to limit gradients, also spans the entire gradient space. Finally, we can thus find $d = np+p+2$ matrices $(A_i)_{i=1}^d$ such that the vectors $(\nabla_\theta \log \tilde{p}(A_i \mid \theta))_{i=1}^d$ are linearly independent. □

With the result of Lemma 5, we can now obtain the asymptotic normality result.

Theorem 6. *Assume that the empirical data distribution is given by the restricted model for some parameter $\theta_0 \in \Theta^{\mathrm{id}}$. Then the MAP estimator $\hat{\theta}_N$ over Θ^{id} for the restricted model converges almost surely to θ_0, and $\hat{\theta}_N$ is asymptotically normal:*

$$\sqrt{N}(\hat{\theta}_N - \theta_0) \xrightarrow[\mathscr{D}]{} \mathcal{N}(0, I(\theta_0)^{-1}) .$$

Proof. As verified in Lemma 14, the restricted model is identifiable on Θ^{id}, so that the only maximizer of $\ell(\theta)$ over Θ^{id} is θ_0. The proof of the consistency Theorem 4 adapts without hurdle to the restricted model, proving that $\hat{\theta}_N$ converges to θ_0 almost surely.

We can now check the conditions to apply Theorem 5. Since $\theta \mapsto \ell(\theta)$ is smooth over Θ, it admits a second-order Taylor expansion at θ_0, and Lemma 4 combined with Lemma 5 ensures that the Hessian matrix at this point is nonsingular. Lemma 15 shows that the Lipschitz condition (6.7) is satisfied by $\log \tilde{p}(A \mid \theta)$. Finally, condition (6.8) is satisfied, as the MAP estimator is such that:

$$\begin{aligned}
\frac{1}{N}\sum_{i=1}^{N} \log p(A_i \mid \hat{\theta}_N) &= \frac{1}{N}\sum_{i=1}^{N} \log \tilde{p}(A_i \mid \hat{\theta}_N) + \frac{1}{N}\log p(\hat{\theta}_N) - \frac{1}{N}\log p(\hat{\theta}_N) \\
&\geq \sup_{\theta \in \Theta} \left(\frac{1}{N}\sum_{i=1}^{N} \log \tilde{p}(A_i \mid \theta) + \frac{1}{N}\log p(\theta) \right) - \frac{1}{N}\log p(\hat{\theta}_N) \\
&\geq \sup_{\theta \in \Theta} \left(\frac{1}{N}\sum_{i=1}^{N} \log \tilde{p}(A_i \mid \theta) \right) - \underbrace{\frac{1}{N}(\sup_{\theta \in \Theta} \log p(\theta) + \log p(\hat{\theta}_N))}_{o_{\mathbb{P}}(1/N)} .
\end{aligned}$$

Theorem 5 thus applies, and grants the convergence in distribution of $\sqrt{N}(\hat{\theta}_N - \theta_0)$ to the centered Gaussian with covariance

$$[\nabla^2_\theta \ell(\theta_0)]^{-1} \mathbb{E}[(\nabla_\theta \log p(A \mid \theta_0))(\nabla_\theta \log p(A \mid \theta_0))^\top][\nabla^2_\theta \ell(\theta_0)]^{-1} = I(\theta_0)^{-1} .$$

□

6.6 Conclusion

This chapter provides theoretical guarantees for the estimation of the eigenvalue and eigenvector distributions of the adjacency matrix decomposition model introduced in Chapter 5. The considered model is identifiable, its MAP estimator exists and converges almost surely to the points minimizing the Kullback-Leibler divergence to the empirical data distribution. By considering an alternate restricted model, we obtain the usual $1/\sqrt{N}$ convergence rate and the asymptotic normality of the MAP estimator using the theory of van der Vaart [1998]. Our results show that asymptotic statistical analysis can be performed on manifold-valued latent variable models to obtain classical guarantees. Arguments similar to those we presented should allow obtaining results in related models where little theoretical work has been done. State-space models on Stiefel and Grassman manifolds [Chikuse, 2006], eigendecomposition models for a single network matrix [Hoff, 2009b] or mixture models [Ali and Gao, 2018] could lend themselves to such an analysis.

The model considered here, as most of the literature on statistics for Stiefel manifolds, is estimated with MLE or MAP. Recently, Pal et al. [2020] proposed a Bayesian framework for von Mises-Fisher distributions which allows computing the posterior distribution of F given observations of X. An interesting question would be to analyze the behavior of this posterior distribution in a hierarchical model where X is a latent variable, in a direction similar to the works of Lin et al. [2017] and Duan et al. [2020].

Finally, another important question on the model we studied is the analysis of its estimation error. In practice, in Chapter 5 we rely on a variant of the EM algorithm to estimate the model parameters. EM-based methods are known to produce local maxima of the likelihood, which prevents from getting a rigorous theoretical analysis of the estimation error. However, even assuming that

no local maximum is found, the E-step of the EM algorithm behaves in an undesirable way, as the conditional distribution of (X, λ) given A is multimodal (one mode per permutation and change of sign for the columns of X). This conditional distribution yields a very low vMF concentration far from the real one, as the samples X are spread over the manifold. A heuristic thus has to be employed in practice to ensure that X stays close to Δ_0, and get a better estimate of the MAP. Future work could investigate this question, either formally justifying this approach or proposing alternate estimation procedures.

6.A Notations

Table 6.A.1 summarizes the main notations used throughout the chapter and the proofs.

6.B Reminders on the Stiefel manifold

The Stiefel manifold is the space of $n \times p$ matrices X such that $X^\top X = I_p$. It inherits a Riemannian manifold structure either as a submanifold of $\mathbb{R}^{n \times p}$ or as a quotient of $O_n(\mathbb{R})$ by $O_{n-p}(\mathbb{R})$. The equivalence between both corresponds to mapping X to the set of orthogonal matrices $(X, X_\perp)$, with $X_\perp$ completing X into an orthonormal basis. The induced metrics are called respectively the Euclidean metric and the canonical metric. The notions exposed here are introduced with great detail and clarity in Edelman et al. [1998].

Tangent space. Let $X \in \mathcal{V}_{np}$. The relation satisfied by matrices H in the tangent space $T_X\mathcal{V}_{np}$ is obtained by differentiating the relation $X^\top X = I_p$: this yields $H^\top X + X^\top H = 0$. This definition of the tangent space can be made more explicit by writing H under the form

$$H = (X, X_\perp)\begin{pmatrix} A \\ B \end{pmatrix} = XA + X_\perp B\,,$$

with $A \in \mathbb{R}^{p \times p}$ and $B \in \mathbb{R}^{(n-p) \times p}$. Such a decomposition is always possible, as $(X, X_\perp)$ is an orthogonal matrix. Using this expression in the equation of the tangent space yields $A^\top = -A$. As a consequence, $T_X\mathcal{V}_{np}$ can be defined as the set of $XA + X_\perp B$, with A a skew-symmetric matrix.

Function gradients. Given a function $f : \mathcal{V}_{np} \to \mathbb{R}$, the manifold gradient of f at X is the matrix-valued function $\nabla_{\mathcal{V}} f$. It is defined by the property that, if X_t is a smooth curve on $\mathcal{V}_{np}$ with $X_0 = X$ and $\dot{X}_0 = H \in T_X\mathcal{V}_{np}$, then $\frac{\mathrm{d}f(X_t)}{\mathrm{d}t}(0) = \langle \nabla_{\mathcal{V}} f(X), H \rangle_X$. Here, $\langle \cdot, \cdot \rangle_X$ denotes the inner product on $T_X\mathcal{V}_{np}$ of the Riemannian manifold structure of $\mathcal{V}_{np}$. Note that the definition of the gradient depends on the metric choice, which is worth mentioning as this choice varies from one paper to another.

An important case is the situation where f can be extended to the whole matrix space. This allows computing the Euclidean gradient of f. Then, depending on the metric choice, explicit formulas are available for the manifold gradient. With respect to the canonical metric, we have [Edelman et al., 1998]:

$$\nabla_{\mathcal{V}} f(X) = \nabla f(X) - X \nabla f(X)^\top X\,.$$

Cayley transform. In Riemannian geometry, the standard way of mapping elements of $T_X\mathcal{V}_{np}$ to the base manifold $\mathcal{V}_{np}$ is the Riemannian exponential map, defined with geodesic equations. Although explicit formulas are available for the exponential map on $\mathcal{V}_{np}$ (see again Edelman et al. [1998]), they rely on matrix exponential and little is known on the properties of the inverse mapping.

In contrast, the Cayley transform C_X behaves better in that regard. It also sends elements from $T_X\mathcal{V}_{np}$ to $\mathcal{V}_{np}$ and behaves similarly to the exponential map close to X, in the sense that

$$C_X(H) = X + H + o(\|H\|_X)\,.$$

Denoting $K = HX^\top - XH^\top$, the Cayley transform at X is defined by:

$$C_X(H) = (I_n + K/2)(I_n - K/2)^{-1} X \in \mathcal{V}_{np}\,.$$

Variable	Definition
A	Symmetric square matrix representing a network
n	Number of rows of the square matrix A
N	Number of random model samples
p	Number of eigenvalues of A accounted for in the model ($p \leq n$).
λ	Element of $\mathbb{R}^p$
$\mathcal{V}_{np}$	Stiefel manifold of $n \times p$ matrices X such that $X^\top X = I_p$
$O_n(\mathbb{R})$	Orthogonal group $\mathcal{V}_{nn}$
X	Element of $\mathcal{V}_{np}$
$X_\perp$	Element of $\mathcal{V}_{n(n-p)}$ completing X into an orthonormal basis
x_i	Column i of matrix X
X_I	Element of $\mathcal{V}_{n\lvert I\rvert}$ formed by $(x_i)_{i\in I}$
$[\mathrm{d}X]$	Invariant measure over $\mathcal{V}_{np}$
$X_{\pi,f}$	For π a permutation and $f \in \{\pm 1\}^p$, we define $X_{\pi,f} = (f_1 x_{\pi(1)}, \dots, f_p x_{\pi(p)})$
Δ_0	Subset of $\mathcal{V}_{np}$ defined in Equation (6.6)
$\Delta_{\pi,f}$	$\Delta_{\pi,f} = \{X_{\pi,f} \mid X \in \Delta_0\}$
$\mathrm{vMF}(F)$	von Mises-Fisher (vMF) distribution on $\mathcal{V}_{np}$ with parameter $F \in \mathbb{R}^{n\times p}$
$\mathcal{C}(F)$	Normalizing constant of the distribution $\mathrm{vMF}(F)$
$\mathcal{C}'(F)$	Normalizing constant of the distribution $\mathrm{vMF}(F)$ restricted to Δ_0
$C_X(D)$	Cayley transform of a tangent vector D at point X
$A_1, \dots, A_N$	Samples of the model considered in the chapter
$X_1, \dots, X_N$	vMF samples representing the eigenvectors of $A_1, \dots, A_N$ (up to noise)
$\lambda_1, \dots, \lambda_N$	Gaussian samples representing the eigenvalues of $A_1, \dots, A_N$ (up to noise)
θ	List of the model parameters: $\theta = F, \mu, \sigma_\lambda, \sigma_\varepsilon$
$p(A \mid \theta)$	Model marginal likelihood of matrix A
$p(A, X, \lambda \mid \theta)$	Complete model likelihood
$p(\theta)$	Prior distribution on θ
$P(\mathrm{d}A)$	Empirical data distribution
F	Parameter of a vMF distribution
(M, s)	Alternate parameterization of vMF distributions: $F = M\mathrm{Diag}(s)$
σ_λ	Variance of the eigenvalues λ_i
σ_ε	Variance of the noise in the observed matrices A_i
$\lambda \cdot X$	Shorthand for $X\mathrm{Diag}(\lambda)X^\top$
$A * X$	Shorthand for $(x_i^\top A x_i)_{i=1}^p$
σ_p^2	Posterior variance of λ given X and A: $\sigma_p^2 = (1/\sigma_\varepsilon^2 + 1/\sigma_\lambda^2)^{-1}$
μ_{AX}	Posterior mean of λ given X and A: $\mu_{AX} = \sigma_p^2(A * X/\sigma_\varepsilon^2 + \mu/\sigma_\lambda^2)$
$\ell(\theta)$	Almost-sure limit of the MAP objective function as $N \to +\infty$
$\hat{\theta}_N$	Maximum A Posteriori estimator of θ
Θ	Space of possible values for θ
Θ^η	For $\eta > 0$, Θ^η represents a truncation of Θ which ensures MAP convergence
Θ^∞	Extension of Θ allowing $s_i = +\infty$ and $\sigma_\lambda = 0$
$\Theta_*, \Theta_*^\eta, \Theta_*^\infty$	Set of minimizers of the KL divergence to the data distribution
$\xi(\theta), \hat{\xi}_N, \Xi_*, \Xi_*^\infty$	Alternate parameterization of $\theta, \hat{\theta}_N, \Theta_*, \Theta_*^\infty$.
$\overline{\Theta^\infty}$	Alexandrov compactification of Θ^∞

Table 6.A.1: Index of notations used throughout the chapter.

C_X was studied in more detailed for $X = I_{np}$ in Jauch et al. [2020b]. In practice, the Cayley transform is used in optimization to perform gradient descent [Fraikin et al., 2007], as it allows projecting the descent direction $\nabla_{\mathcal{V}} f(X)$ onto the manifold and requires only simple linear algebra computations. We prefer it to the exponential map because it has a simple expression, is invertible, and covers the entire manifold apart from a set with measure zero.

Remark. Note that the definition of the Cayley transform used here chapter slightly differs from the standard definition introduced in Chapter 3. In the current chapter, the Cayley transform formula is only used to show that the elements of $\mathcal{V}_{np}$ span $\mathbb{R}^{n\times p}$, and using a simplified formula is sufficient to that purpose.

6.C Proof of the consistency of the MAP estimator

We define

$$\mathbb{E}^* = \sup_{\theta\in\Theta^\infty} \mathbb{E}_{P(\mathrm{d}A)}[\log p(A \mid \theta)]\,.$$

The proof relies on the Alexandrov compactification $\overline{\Theta^\infty}$ of Θ^∞, which adds an infinity point for the coordinates σ_ε (for the cases $\sigma_\varepsilon \in \{0, +\infty\}$), σ_λ (for the case $\sigma_\lambda = +\infty$) and μ (for all the cases where $\|\mu\| = +\infty$).

Part A. We prove that, for all $\theta_\infty \in \overline{\Theta^\infty}$ such that $\delta(\xi(\theta_\infty), \Xi_*^\infty) \geq \varepsilon$, there exists an open neighborhood $\mathcal{U} \subset \overline{\Theta^\infty}$ of θ_∞ such that

$$\mathbb{E}_{P(\mathrm{d}A)}\left[\sup_{\theta\in\mathcal{U}\cap\Theta^\infty} \log p(A \mid \theta)\right] < \mathbb{E}^*\,. \tag{6.10}$$

Let $\mathcal{U}_h$ be a decreasing sequence of open sets such that $\cap_{h\geq 0}\mathcal{U}_h = \{\theta_\infty\}$, and let

$$f_h(A) = \sup_{\theta\in\mathcal{U}_h\cap\Theta^\infty} \log p(A \mid \theta)\,.$$

Two cases arise:

1. If $\theta_\infty \in \Theta^\infty$. Since $\theta \mapsto \log p(A \mid \theta)$ is continuous, we have:

   $$f_h(A) \underset{h\to+\infty}{\longrightarrow} \log p(A \mid \theta_\infty)\,.$$

   And the sequence $f_h(A)$ is decreasing for every A. Furthermore, Lemma 11 ensures that the sequence is bounded from above (with the upper bound obtained by taking the whole space for $\mathcal{U}$). Hence, the monotone convergence theorem applies, and we get:

   $$\lim_{h\to+\infty} \mathbb{E}_{P(\mathrm{d}A)}[f_h(A)] = \mathbb{E}_{P(\mathrm{d}A)}[\log p(A \mid \theta_\infty)] < \mathbb{E}^*$$

   since $\theta_\infty \notin \Theta_*^\infty$. Therefore, it is sufficient to take h large enough to have Equation (6.10) satisfied.

2. If $\theta_\infty \notin \Theta^\infty$, i.e., the variance parameters $(\sigma_\lambda, \sigma_\varepsilon)$ take extreme values, we prove by contradiction that $\lim_{h\to\infty} f_h(A) = -\infty$ a.s. Let us assume that there exists a measurable set $E \in \mathscr{B}(\mathbb{R}^{n\times n})$ such that $\mathbb{P}(A \in E) > 0$ and, for all $A \in E$, $\inf_h f_h(A) > -\infty$. Since $f_h(A)$ is decreasing for every A in E, the infimum is reached at infinity.

   For each h, let $(\theta_{h,m}) \in (\mathcal{U}_h \cap \Theta^\infty)^{\mathbb{N}}$ be a sequence such that:

   $$\lim_{m\to+\infty} \log p(A \mid \theta_{h,m}) = \sup_{\theta\in\mathcal{U}_h\cap\Theta^\infty} \log p(A \mid \theta) = f_h(A) \geq \inf_h f_h(A)\,.$$

   By taking for each h a value of $\theta_{h,m}$ h^{-1}-close to the function's limit, we obtain a sequence $\theta_h \in (\Theta^\eta)^{\mathbb{N}}$ such that $\theta_h \to \theta_\infty$ and

   $$\liminf_{h\to+\infty} \log p(A \mid \theta_h) \geq \inf_h f_h(A) > -\infty\,.$$

Since $\theta_\infty \notin \Theta^\infty$, we have $\sigma_\lambda^\infty = +\infty$, $\sigma_\varepsilon^\infty = 0$ or $\sigma_\varepsilon^\infty = +\infty$. Hence, this contradicts Lemma 12. Therefore, $P(\mathrm{d}A)$-almost surely, $f_h(A) \to -\infty$. We can again apply Lemma 11 and use the monotone convergence theorem, which grants

$$\lim_{h\to+\infty} \mathbb{E}_{P(\mathrm{d}A)}[f_h(A)] = -\infty < \mathbb{E}^* .$$

Therefore, whether θ_∞ is in Θ^∞ or not, there exists an open neighborhood $\mathcal{U}$ of θ_∞ such that

$$\mathbb{E}_{P(\mathrm{d}A)}\left[\sup_{\theta\in\mathcal{U}\cap\Theta^\infty} \log p(A \mid \theta)\right] < \mathbb{E}^* .$$

Part B. Define K_ε as:

$$K_\varepsilon = \{\theta \in \overline{\Theta^\infty} \mid \delta(\xi(\theta), \Xi_*^\infty) \geq \varepsilon\} .$$

By definition of the Alexandrov compactification and by the continuity of δ, K_ε is a compact set, hence we can find a finite open cover $(\mathcal{U}_{h\leq H})$ of it, where each $\mathcal{U}_h$ satisfies Equation (6.10). Let $N \in \mathbb{N}$. For all $\theta \in K_\varepsilon$:

$$\sup_{\theta\in K_\varepsilon\cap\Theta^\infty} \sum_{i=1}^N \log p(A_i \mid \theta) \leq \sup_{1\leq h\leq H} \sum_{i=1}^N \sup_{\theta\in\mathcal{U}_h\cap\Theta^\infty} \log p(A_i \mid \theta) .$$

Since the observations A_i are independent (**H2**), by the law of large numbers and by the definition of $\mathcal{U}_h$:

$$\lim_{N\to+\infty} \frac{1}{N} \sum_{i=1}^N \sup_{\theta\in\mathcal{U}_h\cap\Theta^\infty} \log p(A_i \mid \theta) < \mathbb{E}^* .$$

Hence,

$$\limsup_{N\to+\infty} \sup_{\theta\in K_\varepsilon\cap\Theta^\infty} \frac{1}{N} \sum_{i=1}^N \log p(A_i \mid \theta) < \mathbb{E}^* .$$

Part C. For each $\theta^* \in \Theta_*^\infty$, the law of large numbers gives

$$\lim_{N\to+\infty} \frac{1}{N} \sum_{i=1}^N \log p(A_i \mid \theta^*) = \mathbb{E}^* .$$

Let θ^k be a sequence of parameters with finite values such that $\theta^k \to \theta^*$. Then we have, for all k:

$$p(A^N \mid \hat{\theta}_N) = \frac{p(\hat{\theta}_N \mid A^N)p(A^N)}{p(\hat{\theta}_N)} \geq \frac{p(\theta^k \mid A^N)p(A^N)}{p(\hat{\theta}_N)} = \frac{p(A^N \mid \theta^k)p(\theta^k)}{p(\hat{\theta}_N)}$$

$$\sum_{i=1}^N \log p(A_i \mid \hat{\theta}_N) \geq \sum_{i=1}^N \log p(A_i \mid \theta^k) + (\log p(\theta^k) - \log p(\hat{\theta}_N)) .$$

And, since $\log p(\theta)$ is upper bounded by M, this leads to:

$$\frac{1}{N}(\log p(\theta^k) - \log p(\hat{\theta}_N)) \geq \frac{1}{N} \log \frac{p(\theta^k)}{M} .$$

Hence, $\liminf_{N\to+\infty} \frac{1}{N}(\log p(\theta^k) - \log p(\hat{\theta}_N)) \geq 0$ and, almost surely, for all k:

$$\liminf_{N\to+\infty} \frac{1}{N} \sum_{i=1}^N \log p(A_i \mid \hat{\theta}_N) \geq \mathbb{E}_{P(\mathrm{d}A)}[\log p(A \mid \theta^k)] .$$

And, from the continuity granted by Lemma 12, $\lim_{k\to+\infty} \mathbb{E}_{P(\mathrm{d}A)}[\log p(A \mid \theta^k)] = \mathbb{E}^*$, so that almost surely:

$$\liminf_{N\to+\infty} \frac{1}{N} \sum_{i=1}^N \log p(A_i \mid \hat{\theta}_N) \geq \mathbb{E}^* . \tag{6.11}$$

Part D. Finally, if $\hat{\theta}_N \in K_\varepsilon$ for all $N \in \mathbb{N}$, then:

$$\sum_{i=1}^{N} \log p(A_i \mid \hat{\theta}_N) \leq \sup_{\theta \in K_\varepsilon \cap \Theta^\infty} \sum_{i=1}^{N} \log p(A_i \mid \theta)\,.$$

Which implies almost surely:

$$\limsup_{N\to+\infty} \frac{1}{N} \sum_{i=1}^{N} \log p(A_i \mid \hat{\theta}_N) \leq \limsup_{N\to+\infty} \sup_{\theta \in K_\varepsilon \cap \Theta^\infty} \frac{1}{N} \sum_{i=1}^{N} \log p(A_i \mid \theta) < \mathbb{E}^*\,. \tag{6.12}$$

Which directly contradicts the point of part C. Furthermore, if $\hat{\theta}_N \in K_\varepsilon$ is only true up to a subsequence, the argument remains valid, as all the limits in this proof as $N \to +\infty$ can be taken with respect to any extracted subsequence chosen a *priori*. Therefore, and since we proved in Theorem 3 that $\hat{\theta}_N$ is finite and $\{\theta \in \Theta^\infty \mid \delta(\xi(\theta), \Xi_*^\infty) \geq \varepsilon\} \subset K_\varepsilon$, $\delta(\hat{\xi}_N, \Xi_*^\infty) \geq \varepsilon$ as $N \to +\infty$ almost surely, for all $\varepsilon > 0$. As a consequence, $\delta(\hat{\xi}_N, \Xi_*^\infty) \to 0$ almost surely.

6.D Lemmas

In order to state the required lemmas, let us denote

$$A * X = (x_k^\top A x_k)_{k=1}^p, \quad \frac{1}{\sigma_p^2} = \frac{1}{\sigma_\varepsilon^2} + \frac{1}{\sigma_\lambda^2} \quad \text{and} \quad \mu_{AX} = \sigma_p^2 \left[\frac{1}{\sigma_\varepsilon^2} A * X + \frac{1}{\sigma_\lambda^2} \mu \right]. \tag{6.13}$$

We have the following lemma.

Lemma 6. *The model likelihood rewrites as*

$$p(A \mid \theta) = \frac{1}{(2\pi)^{n^2/2} \sigma_\varepsilon^{n^2}} \frac{\sigma_p^p}{\sigma_\lambda^p} \exp\left(-\frac{1}{2\sigma_\varepsilon^2} \|A\|_F^2 - \frac{1}{2\sigma_\lambda^2} \|\mu\|^2 \right) \mathbb{E}_X \left[\exp\left(\frac{1}{2\sigma_p^2} \|\mu_{AX}\|^2 \right) \right], \tag{6.14}$$

where $\mathbb{E}_X$ denotes the expectation taken with respect to X only.

Proof. From the definition of our model,

$$\begin{aligned} p(A \mid \theta) &= \iint_{\mathcal{V}_{np} \times \mathbb{R}^p} p(A \mid X, \lambda, \theta) p(X \mid \theta) p(\lambda \mid \theta)\, [\mathrm{d}X]\, \mathrm{d}\lambda \\ &= \iint_{\mathcal{V}_{np} \times \mathbb{R}^p} \frac{1}{(2\pi)^{n^2/2} \sigma_\varepsilon^{n^2}} \frac{1}{(2\pi)^{p/2} \sigma_\lambda^p} \frac{1}{C(F)} \exp\left[\langle F, X \rangle - \frac{1}{2\sigma_\varepsilon^2} \|A - \lambda \cdot X\|_F^2 - \frac{1}{2\sigma_\lambda^2} \|\lambda - \mu\|^2 \right] [\mathrm{d}X]\, \mathrm{d}\lambda\,. \end{aligned}$$

Furthermore:

$$\begin{aligned} \|A - \lambda \cdot X\|_F^2 &= \|A\|_F^2 - 2 \sum_{k=1}^{p} \lambda_k \langle A, x_k^\top x_k \rangle_F + \sum_{k,l=1}^{p} \lambda_k \lambda_l \langle x_k^\top x_k, x_l^\top x_l \rangle_F \\ &= \|A\|_F^2 - 2 \sum_{k=1}^{p} \lambda_k (x_k^\top A x_k) + \sum_{k,l=1}^{p} \lambda_k \lambda_l \delta_{kl} \\ &= \|A\|_F^2 - 2 \langle \lambda, A * X \rangle + \|\lambda\|^2\,. \end{aligned}$$

So that, using $\frac{1}{\sigma_p^2} = \frac{1}{\sigma_\varepsilon^2} + \frac{1}{\sigma_\lambda^2}$:

$$\begin{aligned}
p(A \mid \theta) &= \iint_{\mathcal{V}_{np}\times\mathbb{R}^p} \frac{1}{(2\pi)^{n^2/2}\sigma_\varepsilon^{n^2}} \frac{1}{(2\pi)^{p/2}\sigma_\lambda^p} \frac{1}{C(F)} \\
&\qquad \exp\left[\langle F, X\rangle - \frac{1}{2\sigma_\varepsilon^2}\left(\|A\|_F^2 - 2\langle\lambda, A*X\rangle + \|\lambda\|^2\right) - \frac{1}{2\sigma_\lambda^2}\left(\|\lambda\|^2 - \langle\lambda,\mu\rangle + \|\mu\|^2\right)\right] [\mathrm{d}X]\,\mathrm{d}\lambda \\
&= \iint_{\mathcal{V}_{np}\times\mathbb{R}^p} \frac{1}{(2\pi)^{n^2/2}\sigma_\varepsilon^{n^2}} \frac{1}{(2\pi)^{p/2}\sigma_\lambda^p} \frac{1}{C(F)} \\
&\qquad \exp\left[\langle F, X\rangle - \frac{1}{2\sigma_\varepsilon^2}\|A\|_F^2 - \frac{1}{2\sigma_\lambda^2}\|\mu\|^2 + \langle\lambda, \frac{1}{\sigma_\varepsilon^2}A*X + \frac{1}{\sigma_\lambda^2}\mu\rangle - \frac{1}{2}\left(\frac{1}{\sigma_\varepsilon^2} + \frac{1}{\sigma_\lambda^2}\right)\|\lambda\|^2\right] [\mathrm{d}X]\,\mathrm{d}\lambda \\
&= \iint_{\mathcal{V}_{np}\times\mathbb{R}^p} \frac{1}{(2\pi)^{n^2/2}\sigma_\varepsilon^{n^2}} \frac{1}{(2\pi)^{p/2}\sigma_\lambda^p} \frac{1}{C(F)} \\
&\qquad \exp\left[\langle F, X\rangle - \frac{1}{2\sigma_\varepsilon^2}\|A\|_F^2 - \frac{1}{2\sigma_\lambda^2}\|\mu\|^2 + \frac{1}{\sigma_p^2}\langle\lambda, \sigma_p^2\left[\frac{1}{\sigma_\varepsilon^2}A*X + \frac{1}{\sigma_\lambda^2}\mu\right]\rangle - \frac{1}{2\sigma_p^2}\|\lambda\|^2\right] [\mathrm{d}X]\,\mathrm{d}\lambda\,.
\end{aligned}$$

Let $\mu_{AX} = \sigma_p^2\left[\frac{1}{\sigma_\varepsilon^2}A*X + \frac{1}{\sigma_\lambda^2}\mu\right]$. We get:

$$\begin{aligned}
p(A \mid \theta) &= \iint_{\mathcal{V}_{np}\times\mathbb{R}^p} \frac{1}{(2\pi)^{n^2/2}\sigma_\varepsilon^{n^2}} \frac{1}{(2\pi)^{p/2}\sigma_\lambda^p} \frac{1}{C(F)} \\
&\qquad \exp\left[\langle F, X\rangle - \frac{1}{2\sigma_\varepsilon^2}\|A\|_F^2 - \frac{1}{2\sigma_\lambda^2}\|\mu\|^2 + \frac{1}{\sigma_p^2}\langle\lambda,\mu_{AX}\rangle - \frac{1}{2\sigma_p^2}\|\lambda\|^2 \pm \frac{1}{2\sigma_p^2}\|\mu_{AX}\|^2\right] [\mathrm{d}X]\,\mathrm{d}\lambda \\
&= \int_{\mathcal{V}_{np}} \frac{1}{(2\pi)^{n^2/2}\sigma_\varepsilon^{n^2}} \frac{1}{C(F)} \frac{\sigma_p^p}{\sigma_\lambda^p} \exp\left[\langle F, X\rangle - \frac{1}{2\sigma_\varepsilon^2}\|A\|_F^2 - \frac{1}{2\sigma_\lambda^2}\|\mu\|^2 + \frac{1}{2\sigma_p^2}\|\mu_{AX}\|^2\right] \\
&\qquad \underbrace{\int_{\mathbb{R}^p} \frac{1}{(2\pi)^{p/2}\sigma_p^p} \exp\left[-\frac{1}{2\sigma_p^2}\|\lambda - \mu_{AX}\|^2\right] \mathrm{d}\lambda}_{=1} [\mathrm{d}X]\,.
\end{aligned}$$

Thus, we obtain the result:

$$\begin{aligned}
p(A \mid \theta) &= \frac{1}{(2\pi)^{n^2/2}\sigma_\varepsilon^{n^2}} \frac{1}{C(F)} \frac{\sigma_p^p}{\sigma_\lambda^p} \exp\left[-\frac{1}{2\sigma_\varepsilon^2}\|A\|_F^2 - \frac{1}{2\sigma_\lambda^2}\|\mu\|^2\right] \int_{\mathcal{V}_{np}} \exp\left[\langle F, X\rangle + \frac{1}{2\sigma_p^2}\|\mu_{AX}\|^2\right] [\mathrm{d}X] \\
&= \frac{1}{(2\pi)^{n^2/2}\sigma_\varepsilon^{n^2}} \frac{\sigma_p^p}{\sigma_\lambda^p} \exp\left[-\frac{1}{2\sigma_\varepsilon^2}\|A\|_F^2 - \frac{1}{2\sigma_\lambda^2}\|\mu\|^2\right] \mathbb{E}_X\left[\exp\left[\frac{1}{2\sigma_p^2}\|\mu_{AX}\|^2\right]\right]\,.
\end{aligned}$$

□

Lemma 7 (Bound on the log-likelihood)**.** *For all matrix A and parameters θ,*

$$\log p(A \mid \theta) \leq -\frac{n^2}{2}\log(2\pi) - (n^2 - p)\log\sigma_\varepsilon - p\log\sigma_\lambda\,.$$

Proof. Using $\frac{1}{\sigma_p^2} = \frac{1}{\sigma_\lambda^2} + \frac{1}{\sigma_\varepsilon}$, Jensen's inequality gives $\|\mu_{AX}\|^2 \leq \frac{1}{\sigma_\varepsilon^2}\|A*X\|^2 + \frac{1}{\sigma_\lambda^2}\|\mu\|^2$. Proposition 4 implies, for $\mu = 0$, that $\|A*X\| \leq \|A\|_F$. Hence, for all $X \in \mathcal{V}_{np}$, we obtain the inequality $\|\mu_{AX}\|^2 \leq \frac{1}{\sigma_\varepsilon^2}\|A\|_F^2 + \frac{1}{\sigma_\lambda^2}\|\mu\|^2$. This bound yields in the expression of Lemma 6:

$$\log p(A \mid \theta) \leq -\frac{n^2}{2}\log(2\pi) - n^2\log\sigma_\varepsilon + p\log\sigma_p - p\log\sigma_\lambda\,.$$

Furthermore, from the definition of σ_p we have $\sigma_p \leq \sigma_\varepsilon$, which gives the desired bound:

$$\log p(A \mid \theta) \leq -\frac{n^2}{2}\log(2\pi) - (n^2 - p)\log\sigma_\varepsilon - p\log\sigma_\lambda\,.$$

□

Lemma 8 (Continuity of $p(A \mid \theta)$ over Θ^∞)**.** *The likelihood $p(A \mid \theta)$ extends continuously when $s_i = +\infty$ for a subset I of r indices or when $\sigma_\lambda = 0$. In other words, $\theta \mapsto p(A \mid \theta)$ is continuous over Θ^∞. With the following notations*

- *J is the complementary of I in $\{1, \dots, p\}$,*
- *X_I is the $n \times r$ matrix $(x_{i_1}, \dots, x_{i_r})$,*
- *$M_I^\perp$ denotes an $n \times (n-r)$ matrix such that $M_I^\top M_I^\perp = 0$ and $M_I^\perp \in \mathcal{V}_{n,n-r}$.*
- *$q_{\text{vMF}}(X, F)$ is the von Mises-Fisher density with parameter F and variable X,*
- *$F = M\text{Diag}(s)$ is the parameterization of F described in Section 6.2,*

the extension reads:

$$p(A \mid \theta) = \begin{cases} \int_{\mathcal{V}_{n-r,p-r}} q_{\text{vMF}}(Y; (M_I^\perp)^\top F_J) p(A \mid X = (M_I, M_I^\perp Y), \lambda = \mu, \theta) \, [\mathrm{d}Y] & \text{if } \sigma_\lambda = 0 \\ \iint_{\mathcal{V}_{n-r,p-r} \times \mathbb{R}^p} q_{\text{vMF}}(Y; (M_I^\perp)^\top F_J) p(A \mid X = (M_I, M_I^\perp Y), \lambda = \mu, \theta) p(\lambda \mid \theta) \, [\mathrm{d}Y] \mathrm{d}\lambda & \text{otherwise.} \end{cases}$$

If all latent variables are constant, this yields the Gaussian likelihood $A \sim \mathcal{N}(\mu \cdot M, \sigma_\varepsilon^2 I_{n \times n})$.

Proof. For notational convenience, we suppose that I is composed of the first r indices of $\{1, \dots, p\}$. Let $\{X_I = M_I\} \subset \mathcal{V}_{np}$ be the set of values of X such that X and M match on the columns of I. The continuity at infinity comes from the expression:

$$\begin{aligned} p(A \mid \theta) &= \mathbb{E}_{X,\lambda}[p(A \mid X, \lambda, \theta)] \\ &= \mathbb{E}_{X_I,\lambda}[\mathbb{E}_{X_J}[p(A \mid X, \lambda, \theta) \mid X_I, \lambda]] \, . \end{aligned}$$

The conditional expectation, computed below, is continuous (as the parameters in it remain finite). Furthermore, in distribution, $X_I \to M_I$ and $\lambda \to \mu$, as $s_I \to +\infty$ and $\sigma_\lambda \to 0$. Therefore, in the limit the expression reduces to the conditional expectation taken at the limiting final values:

$$\mathbb{E}[p(A \mid X, \lambda, \theta) \mid X_I] = \frac{1}{\iint_{\{X_I = M_I\} \times \mathbb{R}^p} \exp(\langle F_J, X_J \rangle) p(\lambda \mid \theta) \, [\mathrm{d}X] \mathrm{d}\lambda} \iint_{\{X_I = M_I\} \times \mathbb{R}^p} \exp(\langle F_J, X_J \rangle) f(A, X, \lambda) p(\lambda \mid \theta) \, [\mathrm{d}X] \mathrm{d}\lambda \, ,$$

where the measure for X here corresponds to the Hausdorff measure over $\{X_I = M_I\}$. Furthermore, we have $\{X_I = M_I\} = \{(M_I, M_I^\perp Y) \mid Y \in \mathcal{V}_{n-r,p-r}\}$ and the map $Y \mapsto (M_I, M_I^\perp Y)$ is an isometry with respect to the invariant measures (which is equal to the Hausdorff measure for Stiefel manifolds, as noted in Jauch et al. [2020b]). We thus get:

$$\begin{aligned} &\iint_{\{X_I = M_I\} \times \mathbb{R}^p} \exp(\langle F_J, X_J \rangle) f(A, X, \lambda) p(\lambda \mid \theta) \, [\mathrm{d}X] \mathrm{d}\lambda \\ &\qquad = \iint_{\mathcal{V}_{n-r,p-r} \times \mathbb{R}^p} \exp(\langle (M_I^\perp)^\top F_J, Y \rangle) f(A, (M_I, M_I^\perp Y), \lambda) p(\lambda \mid \theta) \, [\mathrm{d}Y] \mathrm{d}\lambda \, , \end{aligned}$$

and similarly $\iint_{\{X_I = M_I\} \times \mathbb{R}^p} \exp(\langle F_J, X_J \rangle) p(\lambda \mid \theta) \, [\mathrm{d}X] \mathrm{d}\lambda = \mathcal{C}((M_I^\perp)^\top F_J)$. □

Lemma 9 (Better bound on the likelihood)**.** *For all parameters $\theta \in \Theta$ and matrices $A \in \mathbb{R}^{n \times n}$ such that $\|\mu\| > \max(2 \|A\|_F, 2\sigma_\lambda \sqrt{p/2 - 1})$, we have the bound*

$$p(A \mid \theta) \leq \frac{1}{(2\pi\sigma_\varepsilon^2)^{n^2/2}} \left(\frac{2 \|\mu\|^p}{\Gamma(p/2) \sigma_\lambda^p} + 1 \right) \exp\left(-\frac{1}{2\sigma_\varepsilon^2} (\|\mu\| / 2 - \|A\|_F)^2 \right) .$$

Proof. Using Proposition 4, which in particular grants that $\|A * X\| \leq \|A\|_F$, we have

$$\begin{aligned}
p(A \mid \theta) &= \mathbb{E}_{\lambda,X}\left[\frac{1}{(2\pi\sigma_\varepsilon^2)^{n^2/2}} \exp\left(-\frac{1}{2\sigma_\varepsilon^2}\|A - \lambda \cdot X\|_F^2\right)\right] \\
&= \mathbb{E}_{\lambda,X}\left[\frac{1}{(2\pi\sigma_\varepsilon^2)^{n^2/2}} \exp\left(-\frac{1}{2\sigma_\varepsilon^2}\left(\|A\|_F^2 - 2\langle\lambda, A * X\rangle, + \|\lambda\|^2\right)\right)\right] \\
&= \mathbb{E}_{\lambda,X}\left[\frac{1}{(2\pi\sigma_\varepsilon^2)^{n^2/2}} \exp\left(-\frac{1}{2\sigma_\varepsilon^2}\left(\|A\|_F^2 - \|A * X\|^2 + \|A * X - \lambda\|^2\right)\right)\right] \\
&\leq \mathbb{E}_{\lambda,X}\left[\frac{1}{(2\pi\sigma_\varepsilon^2)^{n^2/2}} \exp\left(-\frac{1}{2\sigma_\varepsilon^2}\|A * X - \lambda\|^2\right)\right].
\end{aligned}$$

Since $\|A * X\| \leq \|A\|_F$, we have $\|A * X - \lambda\| \geq d(\lambda, B(0, \|A\|_F)) = \max(0, \|\lambda\| - \|A\|_F)$. And since $\|\mu\| > 2\|A\|_F$, we have:

$$\begin{aligned}
(2\pi\sigma_\varepsilon^2)^{n^2/2} p(A \mid \theta) \leq\ & \mathbb{E}_{\lambda,X}\left[\mathbf{1}_{\|\lambda\| \leq \|\mu\|/2} \exp\left(-\frac{1}{2\sigma_\varepsilon^2}\max(0, \|\lambda\| - \|A\|_F)^2\right)\right] \\
&+ \mathbb{E}_{\lambda,X}\left[\mathbf{1}_{\|\lambda\| > \|\mu\|/2} \exp\left(-\frac{1}{2\sigma_\varepsilon^2}(\|\mu\|/2 - \|A\|_F)^2\right)\right] \\
\leq\ & \mathbb{P}(\|\lambda\| \leq \|\mu\|/2) + \exp\left(-\frac{1}{2\sigma_\varepsilon^2}(\|\mu\|/2 - \|A\|_F)^2\right).
\end{aligned}$$

Furthermore,

$$\begin{aligned}
\mathbb{P}(\|\lambda\| \leq \|\mu\|/2) &\leq \mathbb{P}\left(\|\lambda - \mu\| \in \left[\frac{1}{2}\|\mu\|, \frac{3}{2}\|\mu\|\right]\right) \\
&= \mathbb{P}\left(\frac{1}{\sigma_\lambda^2}\|\lambda - \mu\|^2 \in \left[\frac{1}{4\sigma_\lambda^2}\|\mu\|^2, \frac{9}{4\sigma_\lambda^2}\|\mu\|^2\right]\right).
\end{aligned}$$

Since by definition $\lambda \sim \mathcal{N}(0, \sigma_\lambda^2 I_p)$, $\frac{1}{\sigma_\lambda^2}\|\lambda - \mu\|^2$ follows a chi-squared distribution with degree p. Its CDF is given by:

$$F(x) = \frac{\gamma(p/2, x/2)}{\Gamma(p/2)},$$

with $\gamma(p/2, x/2) = \int_{x/2}^{\infty} t^{p/2-1}e^{-t}\,\mathrm{d}t$. Therefore, we have:

$$\begin{aligned}
\mathbb{P}(\|\lambda\| \leq \|\mu\|/2) &\leq \mathbb{P}\left(\frac{1}{\sigma_\lambda^2}\|\lambda - \mu\|^2 \in \left[\frac{1}{4\sigma_\lambda^2}\|\mu\|^2, \frac{9}{4\sigma_\lambda^2}\|\mu\|^2\right]\right) \\
&= \frac{1}{\Gamma(p/2)} \int_{\frac{1}{4\sigma_\lambda^2}\|\mu\|^2}^{\frac{9}{4\sigma_\lambda^2}\|\mu\|^2} t^{p/2-1}e^{-t}\,\mathrm{d}t.
\end{aligned}$$

Furthermore, the function $t \mapsto t^{p/2-1}e^{-t}$ is decreasing for $t > p/2 - 1$ and $\|\mu\|^2 > 4\sigma_\lambda^2(p/2-1)$, so that:

$$\begin{aligned}
\mathbb{P}(\|\lambda\| \leq \|\mu\|/2) &\leq \frac{1}{\Gamma(p/2)} \frac{9\|\mu\|^2 - \|\mu\|^2}{4\sigma_\lambda^2} \left(\frac{1}{4\sigma_\lambda^2}\|\mu\|^2\right)^{p/2-1} \exp\left(-\frac{1}{4\sigma_\lambda^2}\|\mu\|^2\right) \\
&\leq \frac{2\|\mu\|^p}{\Gamma(p/2)\sigma_\lambda^p} \exp\left(-\frac{1}{4\sigma_\lambda^2}\|\mu\|^2\right).
\end{aligned}$$

This finally yields the claimed result:

$$\begin{aligned}
p(A \mid \theta) &\leq \frac{2\|\mu\|^p}{\Gamma(p/2)\sigma_\lambda^p(2\pi\sigma_\varepsilon^2)^{n^2/2}} \exp\left(-\frac{1}{4\sigma_\lambda^2}\|\mu\|^2\right) + \frac{1}{(2\pi\sigma_\varepsilon^2)^{n^2/2}} \exp\left(-\frac{1}{2\sigma_\varepsilon^2}(\|\mu\|/2 - \|A\|_F)^2\right) \\
&\leq \frac{1}{(2\pi\sigma_\varepsilon^2)^{n^2/2}} \left(\frac{2\|\mu\|^p}{\Gamma(p/2)\sigma_\lambda^p} + 1\right) \exp\left(-\frac{1}{2\sigma_\varepsilon^2}(\|\mu\|/2 - \|A\|_F)^2\right).
\end{aligned}$$

□

Lemma 10 (Chevallier et al. [2021], Lemma 1)**.** *Let $p < q$ be two integers. Then, for any differentiable map $f : \mathbb{R}^p \to \mathbb{R}^q$ and any compact subset K of $\mathbb{R}^p$, there exists a constant λ depending only on p and q such that*

$$\int_{\mathbb{R}^q \setminus f(K)} \log^+ \frac{1}{d(A, f(K))} \mathrm{d}A < \lambda \left(\sup_K \|Df\| + 2 \right)^q \mathrm{Diam}(K) \,.$$

Lemma 11. *Assume hypotheses **H1, H3**. We have*

$$\mathbb{E}_{P(\mathrm{d}A)} \left[\sup_{\theta \in \Theta^\infty} (\log p(A \mid \theta))^+ \right] < +\infty \,.$$

Proof. For an observation A and all $\theta \in \Theta$, we have:

$$\begin{aligned} p(A \mid \theta) &= \mathbb{E}\left[p(A \mid X, \lambda)\right] \\ &= \mathbb{E}\left[\frac{1}{(\sigma_\varepsilon \sqrt{2\pi})^{n^2}} \exp\left(-\frac{1}{2\sigma_\varepsilon^2} \|A - \lambda \cdot X\|_F^2 \right) \right] \\ &\leq \frac{1}{(\sigma_\varepsilon \sqrt{2\pi})^{n^2}} \exp\left(-\frac{1}{2\sigma_\varepsilon^2} d(A, R_{np})^2 \right) . \end{aligned}$$

Where R_{np} denotes the set of $n \times n$ matrices with rank less than p. This inequality remains true for $\theta \in \Theta^\infty$, as both sides extends continuously to Θ^∞. Hence, for all $\theta \in \Theta^\infty$:

$$\log p(A \mid \theta) \leq -n^2 \log(\sigma_\varepsilon \sqrt{2\pi}) - \frac{1}{2\sigma_\varepsilon^2} d(A, R_{np})^2 \tag{6.15}$$

$$\leq -n^2 \log(\sigma_\varepsilon \sqrt{2\pi}) - \frac{1}{2\sigma_\varepsilon^2} d(A, R_{np})^2 \,. \tag{6.16}$$

This quantity is maximized for $\sigma_\varepsilon^2 = \frac{1}{n^2} d(A, R_{np})^2$. Which gives, taking the positive part, up to a finite additive constant α:

$$(\log p(A \mid \theta))^+ \leq \alpha + n^2 \log^+ \left(\frac{1}{d(A, R_{np})} \right) . \tag{6.17}$$

We now want to apply Lemma 10 to integrate over A. To that end, we need to parameterize R_{np} with a map from a lower dimensional space. The naive mapping $\mathbb{R}^p \times \mathbb{R}^{n \times p} \to R_{np}$ mapping (λ, X) to $\lambda \cdot X$ does not work directly, as it is not "coercive", in the sense that (λ, X) can go to infinity with $\lambda \cdot X$ possibly staying bounded. This problem is overcome by restricting the X domain of the map $(\lambda, X) \mapsto \lambda \cdot X$ to a set of points close to $\mathcal{V}_{np}$.

Let $f : \mathbb{R}^{n \times p} \times \mathbb{R}^p \to \mathbb{R}^{n \times n}$, defined by $f(U, v) = v \cdot U = U\mathrm{Diag}(v)U^\top$. Then we have that $R_{np} = f(\mathcal{V}_{np} \times \mathbb{R}^p)$. We have furthermore

$$Df_{U,v}(H, w) = U\mathrm{Diag}(v)H^\top + H\mathrm{Diag}(v)U^\top + U\mathrm{Diag}(w)U^\top \,,$$

so that

$$\|Df_{U,v}(H)\|_2 \leq 2 \|U\|_2 \|v\|_\infty \|H\|_2 + \|U\|_2^2 \|w\|_\infty \,.$$

Hence, the operator norm of the differential (for the matrix operator norm) satisfies the inequality $\|Df_{u,v}\|_2 \leq C_{np} \|(U, v)\|_{\ell^1}^2$ (with C_{np} a generic product of norm equivalence constants, whose definition may implicitly vary depending on the equation).

Let $\beta \in]0, 1]$. Since $\mathcal{V}_{np}$ is a compact subset of $\mathbb{R}^{n \times p}$, There exists $X_1, ..., X_H \in \mathcal{V}_{np}$ such that the union of Frobenius balls $\cup_{h=1}^H B_F(X_h, \beta)$ covers $\mathcal{V}_{np}$. In particular, we have

$$f\left(\left(\cup_{h=1}^H B(X_h, \beta) \right) \times \mathbb{R}^p \right) = R_{np} \,.$$

Let $(h,t) \in [\![1,H]\!] \times \mathbb{Z}^p$: we define B_{ht} as $B(X_h,\beta) \times B_\infty(t,1/2)$. Using hypothesis **H1** gives $np+p<n^2$, hence Lemma 10 applies to f. We get:

$$\begin{aligned}\int_{\mathbb{R}^{n\times n}\setminus f(B_{ht})} \log^+ \frac{1}{d(A,f(B_{ht}))}\mathrm{d}A &< \lambda\left(\sup_{B_{ht}}\|Df\|+2\right)^{n^2}\operatorname{Diam}(B_{ht})\\ &\le \lambda\left(\sup_{(U,v)\in B_{ht}} C_{np}\,\|(U,v)\|_{\ell^1}^2+2\right)^{n^2}(\sqrt{n}+\beta)\\ &\le \lambda\left(C_{np}(\|X_h\|_{\ell^1}+\|t\|_{\ell^1}+C_{np}\beta+p)^2+2\right)^{n^2}(\sqrt{n}+1)\\ &\le (a_{np}\,\|t\|_\infty+b_{np})^{n^2} \quad (\text{as } \|U\|_F \le \|X_h\|_F+\beta \le \sqrt{p}+1)\,.\end{aligned}$$

With a_{np}, b_{np} constants depending only on n and p. Let $D_T = \cup_{h\in[\![1,H]\!],\|t\|_\infty\le T} B_{ht}$. We have:

$$\begin{aligned}1/d(A,f(D_T)) &= \sup_{(U,v)\in D_T} 1/d(A,f(U,v)) \le \sum_{h=1}^{H}\sum_{\|t\|_\infty\le T}\sup_{(U,v)\in B_{ht}} 1/d(A,f(U,v))\\ &\le \sum_{h=1}^{H}\sum_{\|t\|_\infty\le T} 1/d(A,f(B_{ht}))\,.\end{aligned}$$

Hence, since the sets $f(B_t)$ have zero Lebesgue measure in $\mathbb{R}^{n\times n}$(as $np+p<n^2$):

$$\begin{aligned}\int_{\mathbb{R}^{n\times n}} \log^+ \frac{1}{d(A,f(D_T))}P(A)\mathrm{d}A &= \sum_{h=1}^{H}\sum_{j=1}^{T}\sum_{\|t\|_\infty=j}\int_{\mathbb{R}^{n\times n}\setminus f(B_{ht})} \log^+ \frac{1}{d(A,f(B_{ht}))}P(A)\mathrm{d}A\\ &\le \sum_{h=1}^{H}\sum_{j=1}^{T}\sum_{\|t\|_\infty=j}(a_{np}\,\|t\|_\infty+b_{np})^{n^2}\max_{d(A,f(B_{ht}))\le 1}P(A)\,.\end{aligned}$$

Now, if A is such that $d(A,f(B_{ht}))\le 1$, we have $\|A-f(X_h,t)\|_F \le 1+C_{np}/2$. Furthermore, since the columns of X_h are orthonormal we have $\|f(X_h,t)\|_2 = \|t\|_\infty$, and we obtain the implication $d(A,f(B_{ht}))\le 1 \implies \|A\|_F \ge C_{np}\,\|t\|_\infty - 1 - C'_{np}/2 \ge c(\|t\|_\infty - 1)$ for some $c>0$. Hence,

$$\begin{aligned}\int_{\mathbb{R}^{n\times n}} \log^+ \frac{1}{d(A,f(D_T))}P(A)\mathrm{d}A &\le \sum_{j=1}^{T}\sum_{\|t\|_\infty=j} H(a_{np}j+b_{np})^{n^2}\max_{d(A,f(B_{ht}))\le 1}P(A)\\ &\le \sum_{j=1}^{T}(j+1)^{(np+p)}H(a_{np}j+b_{np})^{n^2}\sup_{\|A\|_F\ge c(j-1)}P(A)\,.\end{aligned}$$

Since P has an exponentially decaying tail beyond some compact set (hypothesis **H3**), this sum converges to a finite value. Since the sequence $\left(\log^+\left(d(A,f(D_T))^{-1}\right)\right)_{T\in\mathbb{N}}$ is non-negative non-decreasing with limit $\log^+\left(d(A,R_{np})^{-1}\right)$, Fatou's lemma gives:

$$\begin{aligned}\int_{\mathbb{R}^{n\times n}} \log^+\left(\frac{1}{d(A,R_{np})}\right)P(\mathrm{d}A) &= \int_{\mathbb{R}^{n\times n}} \liminf_{T\to+\infty}\log^+\left(\frac{1}{d(A,f(D_T))}\right)P(\mathrm{d}A)\\ &\le \liminf_{T\to+\infty}\int_{\mathbb{R}^{n\times n}} \log^+\left(\frac{1}{d(A,f(D_T))}\right)P(\mathrm{d}A) < +\infty\,.\end{aligned}$$

Thus, we finally get the desired result with Equation (6.17):

$$\mathbb{E}\left[(\log p(A\mid\theta))^+\right] \le \alpha + n^2\int_{\mathbb{R}^{n\times n}} \log^+\left(\frac{1}{d(A,R_{np})}\right)P(\mathrm{d}A) < +\infty\,.$$

□

Lemma 12. *We have:*

1. *$P(\mathrm{d}A)$ almost-surely, for any sequence $\theta_k \in \Theta^\infty$ such that $\lim_{k\to+\infty} \theta_k \in \overline{\Theta^\infty} \setminus \Theta^\infty$,*

$$\lim_{k\to+\infty} \log p(A \mid \theta_k) = -\infty\,.$$

2. *For any sequence $\theta_k \in \Theta^\infty$ such that $\lim_{k\to+\infty} \theta_k \in \overline{\Theta^\infty} \setminus \Theta^\infty$,*

$$\lim_{k\to+\infty} \mathbb{E}_{P(\mathrm{d}A)}\left[\log p(A \mid \theta_k)\right] = -\infty\,.$$

3. *The mapping $\theta \mapsto \mathbb{E}_{P(\mathrm{d}A)}[\log p(A \mid \theta)]$ is continuous on Θ^∞ and $\Theta^\infty_* \neq \varnothing$.*

Proof. We prove the three points consecutively.

1. Let $(\theta_k) \in \Theta^\infty$ a sequence such that $\theta_\infty = \lim_{k\to+\infty} \theta_k \in \overline{\Theta^\infty} \setminus \Theta^\infty$. By definition,

$$\overline{\Theta^\infty} \setminus \Theta^\infty = \{(M, s, \mu, \sigma_\lambda, \sigma_\varepsilon) \mid s \in [0,+\infty]^p \text{ and } (\sigma_\lambda = +\infty \text{ or } \sigma_\varepsilon \in \{0, +\infty\} \text{ or } \mu = \infty)\}\,.$$

   We treat the cases separately, depending on the limits $\sigma_\lambda, \sigma_\varepsilon \in \{0, c > 0, \infty\}$ and $\mu \in \mathbb{R}^p \cup \{\infty\}$.

   (a) $\sigma_\lambda \to \infty, \sigma_\varepsilon \to c$: then, by Lemma 7, $\log p(A \mid \theta) \to -\infty$

   (b) If $\sigma_\varepsilon \to +\infty$ or $\sigma_\varepsilon \to 0$. We can use Lemma 9: since A has density with respect to the Lebesgue measure, $\|A\|_F \neq \|\mu\|/2$ almost surely, so that $\log p(A \mid \theta) \to -\infty$ as $\sigma_\varepsilon \to +\infty$ or $\sigma_\varepsilon \to 0$.

   (c) If $\mu \to \infty$ and ($\sigma_\lambda \to c, \sigma_\varepsilon \to c$ or $\sigma_\lambda \to 0, \sigma_\varepsilon \to c$): Lemma 9 grants that $\log p(A \mid \theta) \to -\infty$.

2. Let $(\theta_k) \in \Theta^\infty$ a sequence such that $\theta_\infty = \lim_{k\to+\infty} \theta_k \in \overline{\Theta^\infty} \setminus \Theta^\infty$. Let $f_k(A) = p(A \mid \theta_k)$. We proved above that, almost surely, $f_k(A) \to -\infty$.

   Let $m < 0$. We have $\mathbf{1}_{f_k(A)\geq m} \to 0$ almost surely, hence $\mathbb{E}_{P(\mathrm{d}A)}[f_k(A)\mathbf{1}_{f_k(A)\geq m}] \to 0$ as $k \to +\infty$.

$$\mathbb{E}_{P(\mathrm{d}A)}[f_k(A)] = \mathbb{E}_{P(\mathrm{d}A)}[f_k(A)\mathbf{1}_{f_k(A)<m}] + \mathbb{E}[f_k(A)\mathbf{1}_{f_k(A)\geq m}] \underset{k\to+\infty}{\leq} m + o(1)\,.$$

   Therefore, $\limsup_{k\to+\infty} \mathbb{E}_{P(\mathrm{d}A)}[f_k(A)] \leq m$ for all $m < 0$, hence

$$\lim_{k\to+\infty} \mathbb{E}_{P(\mathrm{d}A)}[\log p(A \mid \theta_k)] = -\infty\,.$$

3. Let $x > 0$. Lemma 8 shows that the map $\theta \mapsto \log p(A \mid \theta)$ is continuous over the set of parameters $S_x = \{\theta \in \Theta^\infty \mid \sigma_\varepsilon \in [x, 1/x], \sigma_\lambda \leq 1/x\}$, which is a compact. It is therefore bounded, which implies that $\theta \mapsto \mathbb{E}_{P(\mathrm{d}A)}[\log p(A \mid \theta)]$ is continuous over S_x for every x, hence continuous over Θ^∞. Furthermore, suppose that Θ^∞_* is empty. Then any maximizing sequence θ_k is such that $\lim \sigma_\lambda \to +\infty$ or $\lim \sigma_\varepsilon \in \{0, +\infty\}$, which contradicts the point proved above. Therefore, $\Theta^\infty_* \neq \varnothing$.

□

Lemma 13. *For every neighborhood V of I_{np}, $\mathrm{Span}(V \cap \mathcal{V}_{np}) = \mathbb{R}^{n\times p}$. Furthermore, the set $\{X - I_{np} \mid X \in V\}$ also spans $\mathbb{R}^{n\times p}$.*

Proof. The tangent vectors at I_{np} write $H = \binom{A}{B}$ with $A^\top = -A$. The proof relies on a second-order expansion of the Cayley retraction map at I_{np}. Following Jauch et al. [2020b], we define the Cayley transform on this tangent space as a function of $K = \begin{pmatrix} A & -B^\top \\ B^\top & 0 \end{pmatrix}$:

$$C_I(H) = (I_n + K)(I_n - K)^{-1} I_{np}\,.$$

By definition, the map C_I is such that $C_I(H) \in \mathcal{V}_{np}$ for all $H \in T_I\mathcal{V}_{np}$. Furthermore, if an $n \times n$ matrix K is sufficiently small, we have $(I_n + K)^{-1} = I_n - K + K^2 + O(K^3)$. Taking $B = 0$, we get:

$$C_I \begin{pmatrix} \varepsilon A \\ 0 \end{pmatrix} - I_{np} = \varepsilon \begin{pmatrix} A \\ 0 \end{pmatrix} + O(\varepsilon^2)\,.$$

We can thus get linear combinations of elements of $\mathcal{V}_{np}$ arbitrarily close to elements of the form $\binom{A}{0}$ with $A^\top = -A$. Taking $A = 0$ similarly leads to:

$$C_I \begin{pmatrix} 0 \\ \varepsilon B \end{pmatrix} - I_{np} = \begin{pmatrix} -2\varepsilon^2 B^\top B \\ 2\varepsilon B \end{pmatrix} + O(\varepsilon^3)\,.$$

As with A, we obtain a linear combination $\left(C_I\binom{0}{\varepsilon B} - I_{np}\right)/\varepsilon$ arbitrarily close to matrices of the form $\binom{0}{B}$ with $B \in \mathbb{R}^{(n-p)\times p}$. Furthermore, still taking $A = 0$, we obtain:

$$C_I \begin{pmatrix} 0 \\ \varepsilon B \end{pmatrix} + C_I \begin{pmatrix} 0 \\ -\varepsilon B \end{pmatrix} - 2I_{np} = \begin{pmatrix} -4\varepsilon^2 B^\top B \\ 0 \end{pmatrix} + O(\varepsilon^3)\,.$$

We can thus get linear combinations close to elements of the form $\binom{B^\top B}{0}$. This is sufficient to get all matrices with a symmetric upper part, as any symmetric matrix can be obtained as a weighted sum of rank-one matrices of the form $(x, 0, \ldots, 0)^\top \in \mathbb{R}^{(n-p)\times p}$ $(x \in \mathbb{R}^p)$.

As a consequence, there are linear combinations converging to any matrix $\binom{A}{B}$, by combining symmetric and skew-symmetric components for A, and the term for B. In particular, we obtain linear combinations arbitrarily close to a basis of $\mathbb{R}^{n\times p}$, which thus also span the entire space. □

Lemma 14. *The restricted model $\tilde{p}(A \mid \theta)$ is identifiable on Θ^{id}.*

Proof. The parameters $\sigma_\lambda, \sigma_\varepsilon$ and μ can be identified as in Theorem 2. It thus remains to identify $F = M\mathrm{Diag}(s)$ from the distribution of $\lambda \cdot X$. Here, the argument gets much simpler than for the full model: since X is constrained in Δ_0, the mapping $(\lambda, X) \mapsto \lambda \cdot X$ is injective over the whole support of latent variables. Therefore, the changes of variable using the formula of Traynor [1994] directly give access to the density of X over Δ_0 (with the same argument as the one used to obtain $f_\lambda(X)$ for the full model).

By the hypothesis we made when introducing the restricted model, the maximum of $\langle X, F\rangle_F$ over $\mathcal{V}_{np}$ is reached in Δ_0: this point, which can thus be identified, gives the value of M, the normalized columns of F (we recall that we introduced the decomposition $F = M\mathrm{Diag}(s)$).

We use the gradient of $\tilde{p}(X \mid \theta)$ to identify the concentration parameters (s_i). Since the function is defined over $\mathcal{V}_{np}$, we only have access to the projection of its gradient onto the tangent spaces. If we denote by $G(X)$ the Euclidean gradient, the projected manifold gradient writes: $G_{\mathcal{V}}(X) = G(X) - XG(X)^\top X$ [Edelman et al., 1998]. In the case of the function $\tilde{p}(X \mid \theta)$, the manifold gradient thus is: $G_{\mathcal{V}}(X) = \tilde{p}(X \mid \theta)(F - XF^\top X)$. As a consequence, the function $h(X) = F - XF^\top X$ is known over Δ_0. Coherently, we thus obtain:

$$h(M) = M\mathrm{Diag}(s) - M\mathrm{Diag}(s)M^\top M = 0\,.$$

We will now use the first-order variations of $h(X)$ around M to retrieve s. These variations are retrieved by using a simplified Cayley transform on tangent vectors at M (any other smooth retraction map could be used here). As reminded in appendix 6.B, such tangent vectors $H \in T_M\mathcal{V}_{np}$ write as $H = MA + M_\perp B$, with $A^\top = -A$. Denoting $K = HM^\top - MH^\top$, we define the simplified Cayley transform at M by:

$$C_M(H) = (I_n + K)(I_n - K)^{-1}M \in \mathcal{V}_{np}\,.$$

In particular, as in Lemma 13, it satisfies $C_M(\varepsilon H) = M + \varepsilon H + O(\varepsilon^2)$. This gives:

$$\begin{aligned}
h(C_M(\varepsilon H)) &= F - C_M(H)F^\top C_M(\varepsilon H) \\
&= F - (M+\varepsilon H)F^\top(M+\varepsilon H) + O(\varepsilon^2) \\
&= \underbrace{F - MF^\top M}_{=0} - \varepsilon H F^\top M - \varepsilon M F^\top H + O(\varepsilon^2) \\
&= -\varepsilon(MA + M_\perp B)\mathrm{Diag}(s) + \varepsilon M \mathrm{Diag}(s) M^\top (MA + M_\perp B) + O(\varepsilon^2) \\
&= -\varepsilon M[\mathrm{Diag}(s)A + A\mathrm{Diag}(s)] - \varepsilon M_\perp B \mathrm{Diag}(s) + O(\varepsilon^2)\,.
\end{aligned}$$

Taking $B = 0$ and normalizing by ε, we obtain the value of $M[\mathrm{Diag}(s)A + A\mathrm{Diag}(s)]$ for every $p \times p$ skew-symmetric matrix A, which gives $\mathrm{Diag}(s)A + A\mathrm{Diag}(s)$ when multiplying by $M^\top$. For every i, j, taking for A the matrix with $A_{ij} = -A_{ji} = 1$ and zeros everywhere else gives the value of $s_i + s_j$. This gives an over-determined system of equations which allows identifying the s_i's. □

Lemma 15. *If the empirical data distribution is given by $P(A) = \tilde{p}(A \mid \theta_0)$, then condition* (6.7) *for the asymptotic normality theorem of van der Vaart [1998] is satisfied by the restricted model on a neighborhood of θ_0.*

Proof. We are looking for a function $L : \mathbb{R}^{n\times n} \to \mathbb{R}_+$ with $\mathbb{E}[L(A)^2] < +\infty$ and such that, for θ_1 and θ_2 sufficiently close to θ_0,

$$|\log p(A \mid \theta_1) - \log p(A \mid \theta_2)| \leq L(A)\,\|\theta_1 - \theta_2\|\,.$$

Transposed to the restricted model, Lemma 6 gives the marginalized expression:

$$\log \tilde{p}(A \mid \theta) = \log\left[\frac{1}{(2\pi)^{n^2/2}\sigma_\varepsilon^{n^2}}\frac{\sigma_p^p}{\sigma_\lambda^p}\right] - \frac{1}{2\sigma_\varepsilon^2}\|A\|_F^2 - \frac{1}{2\sigma_\lambda^2}\|\mu\|^2 + \log\int_{\Delta_0}\frac{1}{\mathcal{C}'(F)}\exp\left(\langle X, F\rangle_F + \frac{1}{2\sigma_p^2}\|\mu_{AX}\|^2\right)[\mathrm{d}X]\,.$$

For two parameters θ_1 and θ_2, all terms apart from the integral over Δ_0 can be bounded by a term of the form $(C + \|A\|_F^2)\,\|\theta_1 - \theta_2\|$, with C a constant depending on the neighborhood around θ_0. Now let $h(\theta, A, X) = \exp\left(\langle X, F\rangle_F + \frac{1}{2\sigma_p^2}\left\|\mu_{AX}^2\right\|\right)$. Denoting

$$M_{\theta,A} = \max_X \|\mu_{AX}\|^2 \leq \sigma_p^2(\|A\|_F^2/\sigma_\varepsilon^2 + \|\mu\|^2/\sigma_\lambda^2)\,,$$

we have:

$$\begin{aligned}
h(\theta + \mathrm{d}\theta, A, X) &= \exp\left(\langle X, F + \mathrm{d}F\rangle_F + \frac{1}{2\sigma_p^2 + 2\mathrm{d}\sigma_p^2}\left\|\mu_{AX}^2\right\|\right) \\
&= h(\theta, A, X)\left(1 + \langle X, \mathrm{d}F\rangle_F - \frac{\mathrm{d}\sigma_p^2}{2\sigma_p^4}\|\mu_{AX}\|^2 + O((1 + \|A\|_F)^2\,\|\mathrm{d}\theta\|^2)\right) \\
&= h(\theta, A, X)\left(1 + O((1 + \|A\|_F^2)\,\|\mathrm{d}\theta\|)\right)\,,
\end{aligned}$$

where the O notation contains constants depending on θ_0 and the size of its neighborhood. As a consequence:

$$\log\int_{\Delta_0} h(\theta_2, A, X)\,[\mathrm{d}X] - \log\int_{\Delta_0} h(\theta_1, A, X)\,[\mathrm{d}X] = O((1 + \|A\|_F^2)\,\|\theta_2 - \theta_1\|)\,.$$

Finally, the Lipschitz condition (6.7) is satisfied by $L(A) = C(1 + \|A\|_F^2)$. Furthermore, by Lemma 9, $p(A \mid \theta)$ admits second order moments, so that $\mathbb{E}[L(A)^2] < +\infty$. □

Chapter 7

Estimation and Model Selection for Segmented Longitudinal Trajectories

This chapter studies a longitudinal model for disease progression analysis. It takes stock on previous work modeling the trajectory of subjects affected by Parkinson's disease and Alzheimer's disease; in this chapter, the trajectory of each individual is modeled as a piecewise affine trajectory. We show that the population average trajectory can be robustly estimated for trajectories with more than one piece, and that the number of pieces can be selected robustly even with a very strong noise and a large portion of missing data. We apply our methodology to the cohort of the Parkinson's Progression Markers Initiative. We show how our model can be used to describe the disease evolution, and how it could be extended to better account for the impact of treatments.

Contents

7.1 Introduction

In the growing field of large scale observational studies, longitudinal analyzes (repeated measurements of the same subjects at different time points) play a key role in understanding complex population dynamics. They allow modeling the progression of diseases like Alzheimer's disease or Parkinson's disease and monitor the impact of treatments like chemotherapies at the level of individuals.

An increasingly popular approach to model longitudinal data relies on non-linear mixed-effects models. It defines a hierarchical structure which decouples the average trajectory of the population

from the individual-level variability. This approach has proved relevant for disease progression modeling to handle low-dimensional data like clinical scores as well as very high-dimensional data like shapes or images. Although these models have shown a strong capacity to handle complex dynamics, little has been done to assess their performance and choose the best model among a set of candidates. As we will see in more details in the next section, the scientific literature has been evolving in two distinct directions. On the one hand, much work has been devoted to propose expressive non-linear models and devise tailored inference procedure. On the other hand, several papers from the field of model selection have tackled the issue of selecting longitudinal data models. However, most approaches only consider linear mixed-effects models, where closed-form formulas are available and explicit computations can be carried out. Such models have limited expressiveness when it comes to handling real data sets.

In this chapter, we discuss the problem of selecting the best candidate among a set of competing non-linear mixed effects models. We are interested in piecewise linear longitudinal trajectories, as introduced by Chevallier et al. [2021] and Debavelaere et al. [2020]. This framework describes population dynamics with one or more structural breaks, which may correspond to changes in the disease progression or to the impact of a treatment. We focus on the case of clinical scores possibly involving missing data, and wish to select the number of breaks in the average population trajectory. We perform large simulations on synthetic data, and we show that the average population dynamics comprising more than one break can be robustly estimated, along with the individual variability. We show how classical model selection criteria can be used in this complex setting to reliably select the true number of components. We apply our methodology on a longitudinal data set from the Parkinson's Progression Markers Initiative (PPMI) of patients undergoing various treatments for Parkinson's disease.

Section 2 explores the current state of the literature on longitudinal segmented trajectories, disease progression modeling and model selection for longitudinal data. Section 3 introduces the model and details the algorithmic procedure to estimate its parameters and perform model selection. Section 4 presents the results we obtain on both synthetic data sets and cohorts of the PPMI. Section 5 provides complementary remarks on model selection and concludes the study.

7.2 Related Work

Models for longitudinal data. Mixed-effects models provide a general framework to analyze longitudinal observations. They have been widely in the scientific literature, especially in medical applications [Laird and Ware, 1982, Lavielle, 2014]. Their popularity stems from their interpretability and simple formulation, combined to efficient statistical estimation procedures. Mixed-effects models decompose the variability observed in the population of subjects into average trends defined across the population and individual-level deviations. The parameterization of the population average and the individual deviations characterize the model. A large body of work was devoted to the study of mixed-effect models for longitudinal data [van Montfort et al., 2010]. Over the last two decades, the development of new estimation algorithms [Kuhn and Lavielle, 2005] has led to an increased popularity of non-linear mixed-effects models in applications. In particular, it has allowed for the development of models like the spatio-temporal framework of Schiratti et al. [2015] which we will be relying on in this study. Other works like Raket [2020] also proposed tailored non-linear mixed-effects models to monitor the progression of neurodegenerative diseases.

Applications to Parkinson's disease. Over the last decade, longitudinal modeling for the progression of Parkinson's disease has received increased attention in the literature following the creation of the PPMI [Marek et al., 2011]. For a review on longitudinal studies on Parkinson's disease, we refer the reader to Venuto et al. [2016]. More recently, Couronné [2021] showed that non-linear mixed effects models can be used to compute and extract meaningful information from the population dynamics. Ren et al. [2021] showed that longitudinal modeling allows predicting Parkinson's disease's evolution based on clinical scores in the early development of the disease. The work of Severson et al. [2021] relates to our approach of segmenting the disease progression in distinct phases. The authors studied the PPMI data set with a new approach based on Hidden Markov Models. This model accounts for longitudinal disease progression with different hidden

states; it thus provides an alternate way to describe a segmented disease progression.

Information criteria for model selection. The model selection problem has been widely addressed in the literature. The reference method consists in ranking the candidate models according to *information criteria* penalizing the models' likelihoods by taking into account the models' complexity [Konishi and Kitagawa, 2008]. The most widely known are the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC). Although many variants have been proposed throughout the years [Claeskens and Hjort, 2003, Spiegelhalter et al., 2014, Watanabe, 2013, Owrang and Jansson, 2016], AIC and BIC remain efficient and popular approaches for their simplicity and practical performance. The current research now focuses on testing and adapting classical information criteria for specific models and settings [Chen and Chen, 2008, Matsuda et al., 2021].

Longitudinal model selection. Several papers have considered the specific case of model selection for longitudinal data sets. Azari et al. [2006] considers corrected information criteria to account for the correlation between the errors in a linear mixed-effects model. In a similar linear setting, Jones [2011] shows that the number of observations used to compute the BIC can be replaced by a term using the model's Fisher information matrix to capture the effective number of observations. The question of choosing the number of samples in information criteria (as well as in other problems such as Markov Chain Monte-Carlo), more generally aims at determining the *Effective Sample Size* of correlated inputs, and corresponds to the equivalent number of independent observations. In the more general case of non-linear mixed-effects models, Delattre et al. [2014] showed that a BIC-like approximation of the Fisher information can be used to account for both the number of individuals and the number of observations per individual. More recently, Ariyo et al. [2022] investigated the sensitivity to the choice of prior in the selection of linear mixed-effects models for longitudinal data, comparing several tailored Bayesian information criteria.

Segmented regression. Finally, close to our setup of piecewise linear population trajectories is the field of segmented regression. This variant of the linear regression model assumes that the linear parameters of the models are piecewise constant. It seeks to determine the position of the structural breaks and estimate the model's parameters on each break. This problem has been well studied, and it was shown that the BIC allows estimating the number of breaks, while the AIC yields bad performances (see [Bai and Perron, 2003] and the references therein). Strikholm [2006] later proposed a concurrent approach later to dynamically select the number of breaks while simultaneously fitting the model, by iteratively testing the curvature of each piece and adding structural breaks on curved pieces. Although our setup resembles that of segmented regression, the results and methods cannot be transposed directly to the longitudinal context, where the data points vary with the individuals and are located on trajectories that differ from the population average.

7.3 Model and Method

7.3.1 Longitudinal model

Model description

In this chapter, we adopt the framework proposed by Schiratti et al. [2015]. This framework decomposes the variability of the observed data into an average population trajectory and the deviations of each individual from this average trajectory. More specifically, we use the model proposed by Chevallier et al. [2021], where piecewise geodesic average trajectories are modeled and estimated. In the present study, we will be considering piecewise affine trajectories, which alleviates part of the computational burden and makes tractable some computations related to model selection. This assumption is obviously a simplification: a non-negative clinical score which decreases across the progression of a disease would eventually become negative in this framework. While this behavior is not realistic, we will see in our application on PD that it provides a good description of the evolution of the subjects, within the limits of the available observations.

Formally, we model a population of N subjects. Each subject i has n_i data points $(y_{ij})_{1\leq j\leq n_i}$ measured at times $(t_{ij})_{1\leq j\leq n_i}$. A data point y_{ij} consists of d feature scores, where some coordinates may be missing. The model first defines an average population trajectory $D(t)$. This trajectory is taken as a d-dimensional continuous piecewise linear curve, which allows handling structural breaks in the disease progression or changes in a treatment. It is parameterized by a reference point $(p_0, t_{B,0}) \in \mathbb{R}^d \times \mathbb{R}_+$ such that $D(t_{B,0}) = p_0$, and other breakpoints $t_{B,1}, ..., t_{B,K-1}$, with K the number of breaks. The slopes on each of the $K+1$ pieces of the trajectory are given by speed vectors $v_0, \ldots, v_K \in \mathbb{R}^d$ (where the piece $K = 0$ refers to the piece before $t_{B,0}$). The k-th breakpoint position $p_k = D(t_{B,k})$ is then recursively given by $D(t_{B,k+1}) = D(t_{B,k}) + (t_{B,k+1} - t_{B,k})v_{k+1}$.

The trajectory of each individual is defined by a space shift and a time reparameterization of the population trajectory. The space shift represents the difference between the features of each individual and those of the average trajectory. The time reparameterization allows each individual trajectory to happen at a different time and at a different pace. More precisely, the trajectory $D_i(t)$ of subject i is defined as $D_i(t) = D(\varphi_i(t)) + w_i$. Here, $w_i \in \mathbb{R}^d$ and $\varphi_i : \mathbb{R} \to \mathbb{R}$ is a piecewise linear increasing function. It is parameterized by a time shift τ_i, and, for each trajectory piece k, a factor $\alpha_{i,k} > 0$ which measures the acceleration on piece k compared to the reference trajectory. Since the acceleration factors are positive, they are expressed as $\alpha_{i,k} = \exp(\xi_{i,k})$ $(\xi_{i,k} \in \mathbb{R})$ for convenience purpose. With these notations, the time reparameterization writes as:

$$\varphi_i(t) = \begin{cases} t_{B,0} - e^{\xi_{i,0}}(t_{B,0} - t + \tau_{i,0}) & \text{if } t \leq t_{B,0} + \tau_{i,0} \\ t_{B,k} + e^{\xi_{i,k}}(t - t_{B,k} - \tau_{i,k}) & \text{if } t_{B,k} + \tau_{i,k} \leq t \leq t_{B,k+1} + \tau_{i,k+1} \text{ for some } k\,, \end{cases}$$

with the convention $t_{B,K} = +\infty$. The notation $\tau_{i,k}$ designates the time shift between the position $t_{B,k}$ of the break k and its translation on the individual trajectory. It is recursively given by $\tau_{i,0} = \tau_{i,1} = \tau_i$ and, for $k > 1$:

$$\tau_{i,k} = \tau_{i,k-1} + (t_{B,k} - t_{B,k-1})(e^{-\xi_{i,k-1}} - 1)\,.$$

This formulation gives rise to a hierarchical statistical model by defining the distribution of individual variables:

$$\begin{cases} \tau_i \sim \mathcal{N}(0, \sigma_\tau^2) \\ \alpha_{i,k} = \exp(\xi_{i,k}), \xi_i \sim \mathcal{N}(0, \mathrm{Diag}(\sigma_\xi^2)) \\ w_i \sim \mathcal{N}(0, \mathrm{Diag}(\sigma_w^2)) \\ y_{ij} \sim \mathcal{N}(D_i(t_{ij}), \mathrm{Diag}(\sigma^2))\,. \end{cases}$$

To summarize, the measurements $y_{i,1}, \ldots, y_{i,n_i}$ for each individual are modeled as noised observation of a hidden trajectory $D_i(t)$. This trajectory is defined as a space shift and time reparameterization of an average population trajectory $D(t)$; the parameters (τ_i, ξ_i, w_i) of this transformation are hidden latent variables. The population trajectory $D(t)$ is defined as a piecewise linear curve; it is parameterized by break times, slope vectors and the intercept. In the context of disease progression modeling, breaks may account either for structural changes in the disease progression, or for the impact of a treatment.

Remark. The longitudinal framework of Schiratti et al. [2015] classically requires an orthogonality condition between w_i and the direction v_0 of the population trajectory. This condition ensures that the deviation between $D(t)$ and $D_i(t)$ is characterized by a unique set of latent variables (τ_i, ξ_i, w_i). In the context of piecewise geodesic trajectories, Chevallier et al. [2021] adapted this condition by imposing the w_i to be orthogonal to all the speed vectors v_k. This constraint made sense in the context of very high-dimensional data, e.g., image or shape analysis. In the lower-dimensional setting of clinical scores, it becomes more restrictive: e.g., for a trajectory in dimension 2, a trajectory with one break imposes that $w_i = 0$ for all i. Furthermore, the latent variables of piecewise affine individual trajectories are completely determined by the position in space and time of the breaks, which removes the main identifiability issue of the model with no break. For these reasons, we choose to work on a variant of the model of Chevallier et al. [2021] with no orthogonality condition.

Estimation procedure

Estimating the model amounts to determining its parameters $\theta = (p_0, t_B, v, \sigma_\tau^2, \sigma_\xi^2, \sigma_w^2, \sigma^2)$ as well as the posterior distribution of the latent variables $z_i = (\tau_i, \xi_i, w_i)$ given the observed data. In this chapter, we compute the parameters using the Maximum A Posteriori estimator (MAP)

$$\widehat{\theta} \in \operatorname{argmax}_\theta p(\theta \mid y) = \operatorname{argmax}_\theta p(\theta, y)\,,$$

which allows accounting for *a priori* information and regularizes the Maximum Likelihood Estimator (MLE), while theoretically ensuring that the optimal values are not infinite.

Handling missing data. In practice, the data may involve an important proportion of missing coefficients (around 40% in our application). This problem has been widely studied, in particular in the case of longitudinal data sets [Ibrahim and Molenberghs, 2009]. In this study, we make the simple assumption that the data is missing completely at random, i.e., that the data missingness is independent of the data value, and the missingness of each coordinate is chosen independently of the others'. We follow the approach of Couronné [2021] and maximize the likelihood of the observed data y_{obs}: $\widehat{\theta} \in \operatorname{argmax}_\theta p(\theta \mid y_{\text{obs}})$. The precise implementation of this formulation is detailed in Appendix 7.B. In the remainder of the chapter, with a slight abuse of notations, we designate by y the observed data y_{obs}.

Estimation algorithm. For non-linear mixed effects models belonging to the curved exponential family, the MAP is computed with the MCMC-SAEM algorithm [Kuhn and Lavielle, 2004, Allassonnière et al., 2010], which we briefly describe here. Estimating hierarchical models is classically done with the EM algorithm, which alternates computations of expectations (E-step) with respect to the latent variables and maximization with respect to the parameters (M-step). It is especially interesting in the specific setup of curved exponential models, i.e., models where the likelihood $\log p(y, z \mid \theta)$ writes as $\langle S(y, z), \Phi(\theta)\rangle + \Psi(\theta)$. The EM algorithm then write as:

1. **E-step**: Compute $S_t = \mathbb{E}_{p(z|y,\theta_t)}[S(y, z)]$,
2. **M-step**: Obtain $\theta_{t+1} \in \operatorname{argmax}_\theta \langle S_t, \Phi(\theta)\rangle + \Psi(\theta)$.

In complex latent variable models, the E-step becomes intractable and has to be approximated. In the specific case of curved exponential models, the SAEM (Stochastic Approximation EM, [Delyon et al., 1999]) algorithm overcomes this hurdle by estimating S_t with a stochastic approximation procedure. It consists in sampling from the posterior distribution $(z \mid y, \theta_t)$ and using the samples to iteratively estimate $\mathbb{E}_{p(z|y,\theta_t)}[S(y, z)]$. The samples are aggregates with decreasing gains γ_t sequence of the form $\gamma_t = \gamma_0/t^\alpha$. In cases like ours where the sampling step cannot be performed explicitly, the MCMC-SAEM replaces exact sampling with a single step of Markov Chain Monte-Carlo. The general procedure is summarized in Algorithm 7.3.1.

Algorithm 7.3.1: The MCMC-SAEM Algorithm

Initialize θ_0, z_0 and S_0
repeat
  Sample $z_{t+1} \sim q(\cdot \mid z_t; \theta_t)$ from the MCMC kernel $q(z \mid z', \theta_t)$ targeting $p(z \mid y, \theta_t)$
  Update $S_{t+1} = (1 - \gamma_t)S_t + \gamma_t S(y, z_{t+1})$
  Find $\theta_{t+1} \in \operatorname{argmax}_\theta \langle S_{t+1}, \Phi(\theta)\rangle + \Psi(\theta)$
until *convergence*
return θ_T, $(z_t)_{t=1}^T$

Model exponentialization. In our case, the model cannot be written in exponential form, mainly because of the role of the variables p_0, t_B and v in each $D_i(t)$. Kuhn and Lavielle [2005] showed that this hurdle can be overcome with a simple trick, which consists in treating p_0, t_B and v as latent variables following tight Gaussian distributions centered around "twin" parameters $\mathcal{N}(\bar{p}_0, s_{p_0}^2 I_d)$, $\mathcal{N}(\bar{t}_B, s_{t_B}^2 I_K)$, and $\mathcal{N}(\bar{v}, s_v^2 I_{(K+1)d})$. The variance parameters control how close

the so-called *exponentialized model* is from the base model. This new model can easily be shown to be curved exponential, and thus lends itself to estimation with the MCMC-SAEM. The recent work of Debavelaere and Allassonnière [2021] proved that the estimation error induced by this change of model is bounded by a power of the exponentialization variance parameters. The formulas for the sufficient statistics and the maximization step in the MCMC-SAEM are detailed in Appendix 7.B.

Prior distribution. In this work, as in the work of Chevallier et al. [2021], we consider a Bayesian framework and seek the MAP estimator. High variance Gaussian prior distributions are given for the population variables $\bar{p}_0$, $\bar{t}_B$ and $\bar{v}$. Similarly, high variance Inverse Gamma distributions are used for the variance parameters σ_τ^2, σ_w^2 and σ^2.

However, when experimenting with uninformative priors on σ_ξ^2 when $K > 1$, we noticed that the shape of estimated individual trajectories $D_i(t)$ tends to strongly differ from the population template $D(t)$, while still poorly fitting the data. We observed that the estimated values of the ξ_i's may strongly change the length of the trajectory pieces, in such a way that most observations of each individual are regrouped in one piece. We believe that this behavior might be caused by the strongly non-convex shape of the posterior distribution, which affects the robustness of the estimated individual trajectories. In our application to clinical scores, we thus use a strongly informative prior to regularize σ_ξ^2, inducing an implicit penalty on the posterior distribution of the ξ_i's given the observed data. A detailed specification of the prior distribution is given in Appendix 7.A.

Implementation details. In practice, we initialize the MCMC-SAEM by taking a linear regression of each feature across time, setting all speed vectors v_k equal. In the MCMC-SAEM, we use a Symmetric Random Walk Metropolis Hastings within Gibbs sampler with Gaussian transitions. The variance of the proposal Gaussian transitions are tuned along the SAEM steps to reach a desired Metropolis acceptance rate.

All the algorithms presented in this chapter are implemented in Python 3.9 and compiled at runtime with the Python module `numba`, which allows for fast nested loops. The code for the algorithm, as well as scripts to reproduce the experiments on synthetic data, are available online[1].

7.3.2 Model selection

Information criteria for longitudinal data

In order to select the value of the number of breaks K, we compute an information criterion of the form

$$IC(y, \theta) = -2 \log p(y \mid \theta) + \mathrm{pen}(y, \theta)\,,$$

with $\mathrm{pen}(y, \theta)$ a penalty term favoring simple models. For the AIC, $\mathrm{pen}(y, \theta) = 2 \dim(\theta)$, whereas the BIC uses $\mathrm{pen}(y, \theta) = \log(n_{\mathrm{obs}}) \dim(\theta)$, with n_{obs} the number of observations. Both penalties come with their own justifications and theoretical guarantees. When the models are a good approximation to the true data distribution, the AIC can be seen as a first-order Taylor expansion of the test error on unseen data. Asymptotically, AIC is equivalent to cross-validation and selects the model with the best generalization performance. The BIC is an asymptotic Laplace approximation of the integrated data likelihood $\log p(y) = \log \int p(y \mid \theta) p(\theta) \,\mathrm{d}\theta$. If the data was generated from one of the candidate models, asymptotically the BIC chooses the right model.

As emphasized in Debavelaere et al. [2020], choosing between several non-linear mixed effects model is a difficult question, as the classical model selection criteria require using the marginal likelihood of the observed data $p(y \mid \theta)$, which writes as an integral over the latent variables z_i:

$$p(y \mid \theta) = \prod_{i=1}^{N} \int p(y_i \mid z_i, \theta) p(z_i \mid \theta) \,\mathrm{d}z_i\,.$$

When working with high-dimensional complex longitudinal models, the only option to assess the estimation performance is to measure a reconstruction error; it often amounts to computing

[1] `https://github.com/cmantoux/longitudinal-segmented-regression`

the conditional likelihood $p(y \mid z, \theta)$. This applies in particular to longitudinal models on shapes and images, for which the latent space dimension does not allow computing the marginal likelihood $p(y \mid \theta)$ [Debavelaere et al., 2020]. In such cases, the conditional likelihood is used by default, but one may also wonder to what extent it could also be relevant for models with smaller dimension. The work of Merkle et al. [2019] answers this question: the authors show that using the conditional likelihood in information criteria selects models which generalize best on new observations for the *same latent variables*. In our longitudinal context, it means that such criteria select models that generalize well on new observations of the individuals in the training data set, rather than on new observations on unseen individuals. It is furthermore impossible to obtain new data points in-between the known measurements, as it would mean going back in time to obtain new data. Hence, the conditional likelihood does not seem an appropriate solution in our context, and we thus need to estimate the marginal likelihood $p(y \mid \theta)$. Contrary to the high-dimensional settings mentioned above, our model is sufficiently small that the marginal likelihoods can be estimated. As a last remark on this matter, from a model selection perspective, using the conditional likelihood as a performance measure naturally leads to overfit the data : the conditional likelihood uses the best value for the latent variables, which thus act as supplementary parameters without being counted as such in information criteria.

Another question arising in the BIC penalty is the choice of the number of observations n_{obs}. If each individual had only one time point ($n_i = 1$), the observations points would be independent and we would take $n_{obs} = N$. Since repeated measurements of a single individual are correlated, it is unclear to what extent the natural extension $n_{obs} = n_1 + \cdots + n_N$ produces a relevant BIC approximation, and the choice of the effective sample size requires some consideration. In the case of longitudinal linear mixed effects model, Jones [2011] proposes to use the Fisher information matrix, which in this specific case can be computed explicitly. The Fisher information naturally arises when computing the Laplace approximation which defines the BIC:

$$\log p(y) \simeq \log p(y \mid \hat{\theta}_{\text{MLE}}) - \frac{\dim(\theta)}{2} \log(N) - \log \det \mathcal{I}(\hat{\theta}_{\text{MLE}}) + \log p(\hat{\theta}_{\text{MLE}}) \,,$$

with $\hat{\theta}_{\text{MLE}}$ the Maximum Likelihood Estimator (MLE) of θ and $\mathcal{I}(\theta)$ the Fisher information. From its definition, the Fisher information scales with the number of observations per individual: including its determinant in the BIC hence allows accounting for this additional source of information. However, in non-linear settings, the Fisher information is hard to compute in general. Delattre et al. [2014] tackled this issue and showed that, under mild assumptions on the model regularity, the Fisher determinant can be approximated. They propose a tailored, hybrid expression of the BIC:

$$BIC_h(y, \theta) = -2 \log p(y \mid \theta) + \dim(\theta_R) \log(N) + \dim(\theta_F) \log(n_{\text{tot}}) \,,$$

with $n_{\text{tot}} = n_1 + \cdots + n_N$, θ_R the components of θ involved in the random effects and θ_F the parameters of the fixed effects. In our context, this splitting naturally writes as $\theta_R = (\sigma_\tau^2, \sigma_\xi^2, \sigma_w^2)$ and $\theta_F = (p_0, t_B, v, \sigma^2)$. In this chapter, we rely on BIC_h computed from the marginal observation's likelihood to perform model selection. To summarize, BIC_h gather three decisive advantages over other alternatives: 1) it handles the correlated structure of the observations with a minimal cost compared to classical information criteria, 2) unlike selection tools based on the conditional likelihood, it selects model that best generalize on unseen individuals, and 3) it imposes a penalty stronger than AIC, and it is thus more robust to the parameter uncertainty induced by the estimation procedure.

Marginal likelihood estimation

The estimation of marginal densities, and more generally normalizing constants, is a well-studied problem. The naive method for estimating $p(y_i \mid \theta) = \int p(y, z_i \mid \theta) \mathrm{d}z_i$ consists in computing the Monte-Carlo estimator

$$\widehat{p}_{MC}(y_i \mid \theta) = \frac{1}{M} \sum_{k=1}^{M} p(y_i \mid z_{i,k}, \theta), \text{ with } z_{i,k} \sim p(z \mid \theta) \,.$$

Importance sampling. The above estimator is known for having a very large variance, and better estimators can be computed without increasing the computation time. The variance issue is critical for density estimation, as the quantities of interest may vary by several orders of magnitude from one sample to another. For a thorough review on marginal density estimation techniques, we refer the reader to Llorente et al. [2022]. The most standard method is importance sampling (IS), which, given a so-called importance density $q_{IS,i}(z)$, computes the unbiased Monte-Carlo estimator:

$$\widehat{p}_{IS}(y_i \mid \theta) = \frac{1}{M}\sum_{k=1}^{M} \frac{p(y_i \mid z_{i,k}, \theta)}{q_{IS,i}(z_{i,k})}, \text{ with } z_{i,k} \sim q_{IS,i}(z)\,.$$

It is well known that the optimal choice for the importance density in terms of Monte-Carlo variance is given by $q^{\text{opt}}_{IS,i}(z_i) = p(z_i \mid y_i, \theta)$, the posterior density of the latent variables given the observed data. Unfortunately, this density is only known up to its normalizing constant, which happens to be the quantity of interest $p(y_i \mid \theta)$. The simplest way to overcome this issue is to approximate $q^{\text{opt}}_{IS,i}$ by a simpler distribution with known normalizing constant. In this chapter, we use a Gaussian approximation of the posterior density. In practice, the optimal Gaussian importance density $\mathcal{N}(\mu_{IS,i}, \Sigma_{IS,i})$ can be obtained by sampling from the distribution of $p(z_i \mid y_i, \theta)$ and computing the sample mean and covariance matrix.

Bridge sampling. Although importance sampling improves on the naive Monte-Carlo estimator, its variance remains too high to allow discriminating between different models. In this study, we instead use bridge sampling, an alternate procedure which relies on the importance density but may also leverage MCMC samples [Mira and Nicholls, 2004, Gronau et al., 2017]. Bridge sampling relies on the bridge identity:

$$p(y_i \mid \theta) = \frac{\mathbb{E}_{Z_i \sim q_{IS,i}(z)}[\, p(y_i \mid Z_i, \theta)\, h(Z_i)\,]}{\mathbb{E}_{Z_i \sim p(z_i \mid y_i, \theta)}[\, q_{IS,i}(Z_i)\, h(Z_i)\,]}\,,$$

with $h(z_i)$ a so-called bridge function which can be chosen freely. This identity yields a Monte-Carlo estimator relying on M_1 samples $(z^{(1)}_{i,k})$ from $q_{IS,i}$ and M_2 samples $(z^{(2)}_{i,k})$ from $p(z_i \mid y_i, \theta)$:

$$\widehat{p}_{BS}(y_i \mid \theta) = \frac{\frac{1}{M_1}\sum_{k=1}^{M_1} p(y_i \mid z^{(1)}_{i,k}, \theta) h(z^{(1)}_{i,k})}{\frac{1}{M_2}\sum_{k=1}^{M_2} q_{IS,i}(z^{(2)}_{i,k}) h(z^{(2)}_{i,k})}\,. \tag{7.1}$$

As shown by Meng and Wong [1996], the optimal bridge function (in terms of mean squared error of the estimator $\widehat{p}_{BS}(y_i)$) is given by

$$h(z_i) = \frac{1}{s_1 p(y_i, z_i \mid \theta) + s_2 p(y_i \mid \theta) q_{IS,i}(z_i)}\,, \tag{7.2}$$

with $s_1 = M_1/(M_1 + M_2)$ and $s_2 = M_2/(M_1 + M_2)$. In practice, the term $p(y_i \mid \theta)$ in $h(z_i)$ is replaced with its estimator $\widehat{p}_{BS}(y_i \mid \theta)$. This yields a fixed-point equation on $\widehat{p}_{BS}(y_i \mid \theta)$, which is solved iteratively. The relative mean-squared error (RMSE) $\mathbb{E}[(p - \widehat{p})^2]/p^2$ of the bridge sampling estimator can be approximated by accounting for the sample correlation induced by the MCMC. This approximation, in turn, asymptotically matches the variance of $\log \widehat{p}_{BS}(y_i \mid \theta)$, which thus allows for confidence intervals on the information criteria. We refer the reader to Gronau et al. [2017] for further developments. In our numerical experiments, this variance consistently remains very low compared to most likelihood differences used for model comparison.

Marginalization. Finally, in the specific case of piecewise linear trajectories considered in this chapter, straightforward computations allow factoring the complete density of $p(y_i, z_i \mid \theta)$ as $p(y_i, (\tau_i, \xi_i) \mid \theta) \times p(w_i \mid y_i, (\tau_i, \xi_i), \theta)$, with $p(w_i \mid y_i, (\tau_i, \xi_i), \theta)$ given by a conjugate posterior Gaussian distribution. This factorization allows reducing the dimension of the integrals of interest from $d + K + 2$ to $K + 2$ by sampling only from the time-related latent variables τ_i and ξ_i. The factorization formulas are detailed in Appendix 7.C. This dimensionality reduction helps reduce the number of samples required to obtain a sufficiently small variance.

Remark. As recommended when computing sums, products and differences of small probabilities, in the numerical implementation of bridge sampling we store the logarithm of the probabilities rather than their actual value in order to minimize precision loss.

Practical procedure summary

The complete procedure for the marginal likelihood estimation is summarized in Algorithm 7.3.2. In order to sample from the distributions $(z_i \mid y_i, \theta)$, we use a Metropolis within Gibbs sampler with Gaussian transitions. Two differences are to be noted compared with the MCMC in the SAEM. First, we use an adaptive covariance matrix, estimated along the MCMC iterations from the samples, and rescaled to target a given acceptance rate. Second, the variables z_i are conditionally independent given y and θ, and can thus be sampled in parallel. This could not be done in the SAEM procedure, as the maximization step has to be performed after every MCMC step, which prevents from exploiting the independence. In practice, most of the computation time is spent in evaluating the model likelihood of the MCMC samples and importance density samples.

Algorithm 7.3.2: Bridge Sampling procedure for the marginal likelihood estimation of $p(y_i \mid \theta)$

Run a first Metropolis within Gibbs chain $z_{i,1}^{(0)}, \dots, z_{i,M_0}^{(0)}$ targeting the distribution of $p(z_i \mid y_i, \theta)$

Compute the empirical mean μ_{IS} and covariance Σ_{IS} of $(z_i^{(t)})_{1 \leq t \leq M_0}$

Define the importance distribution $q_{IS,i} = \mathcal{N}(\mu_{IS}, \Sigma_{IS})$

Run a second Metropolis within Gibbs chain to draw M_1 samples $z_{i,1}^{(1)}, \dots, z_{i,M_1}^{(1)}$ targeting $p(z_i \mid y_i, \theta)$

Draw M_2 samples $z_{i,1}^{(2)}, \dots, z_{i,M_2}^{(2)}$ from $q_{IS,i}$

Initialize $\widehat{p}_0 = 0$

repeat

  Define $\widehat{h}_t$, the approximation of h obtained by replacing $p(y_i \mid \theta)$ with $\widehat{p}_t$ in Equation (7.2)

  Compute $\widehat{p}_{t+1}$ using Equation (7.1) by replacing h with $\widehat{h}_t$

until *convergence*

Compute the RMSE estimator for $\widehat{p}_{\text{final}}$

return $\widehat{p}_{\text{final}}$ *and the RMSE estimator*

7.4 Results

7.4.1 Synthetic data sets

First, we rely on synthetic data sets to validate the estimation algorithm and investigate the robustness of the model selection procedure. We perform a large number of simulations on a series of randomly generated data sets to assess the average performance for estimation and model selection depending on the noise level and the number of breaks.

Data generation and estimation. We simulate data with characteristics similar to the real data set studied in the next section. We fix a maximum number of breaks $K_{max} = 5$, which is unlikely to be reached in our practical application, where the observed trajectories have around 15 points on average. Then, for each number of break K, we generate five series of ten data sets with $N = 400$ subjects in dimension $d = 6$ with randomly generated population trajectories. Each series corresponds to an average noise level. This level is measured by the fraction s of the variance unexplained by the latent variables. We consider five noise levels evenly spread between $s = 10\%$ and $s = 80\%$: $s \in \{10\%, 28\%, 45\%, 62\%, 80\%\}$.

The variability of the individual trajectories is chosen to roughly match that of the real data; in particular the time shift variability σ_τ is non-negligible with respect to the distances between the break times. The time points of each individual are evenly spaced in time. The observation count per subject is drawn from a Poisson distribution whose average value increases with K: trajectories at $K = 0$ have 6 points on average, whereas trajectories at $K = 5$ have 16 points on average. Finally, we randomly hide 40% of the coefficients in the observed data as missing values, which approximately corresponds to the missing data proportion in our application. This setup is called experiment (A).

Experiment (A) aims at evaluating the feasibility of the estimation task; for this specific purpose, models with only one break need less observations as models with four breaks. However, this difference in the observation count might induce a bias when comparing the estimation performances across different numbers of breaks. In parallel, we thus perform a second series of tests, called experiment (B), with a similar setup, except that, for every number of breaks K, the average number of observations per individual is set to 16. This second experiment measures the increasing difficulty of estimating more complex models from the same amount of data. Note that experiments (A) and (B) match for $K = 5$, this series of simulations is thus only performed once.

For each of these 300 data sets, we run the MCMC-SAEM algorithm for 100,000 iterations for each value of $K \in \{0, \dots, K_{max}\}$. We estimate the observations' marginal likelihood with the procedure described in Algorithm 7.3.2, running each MCMC for 100,000 iterations and drawing 100,000 samples from the importance density. For both MCMCs, the first 10% samples are discarded for burn-in. Finally, we repeat experiments (A) and (B), considering only $N/2 = 200$ subjects to investigate the impact of smaller data on estimation and model selection. We denote these experiments (A') and (B').

Remark. When generating the synthetic data, we initially used the same noise variances for various values of K. However, it turned out that this process made the estimation harder in terms of unexplained variance for small values K, mainly because of the data generation procedure (and more specifically the break time positions). Setting the noise level for each set of parameters based on the explained variance allows for a fair comparison between models, agnostic to the data generation choices.

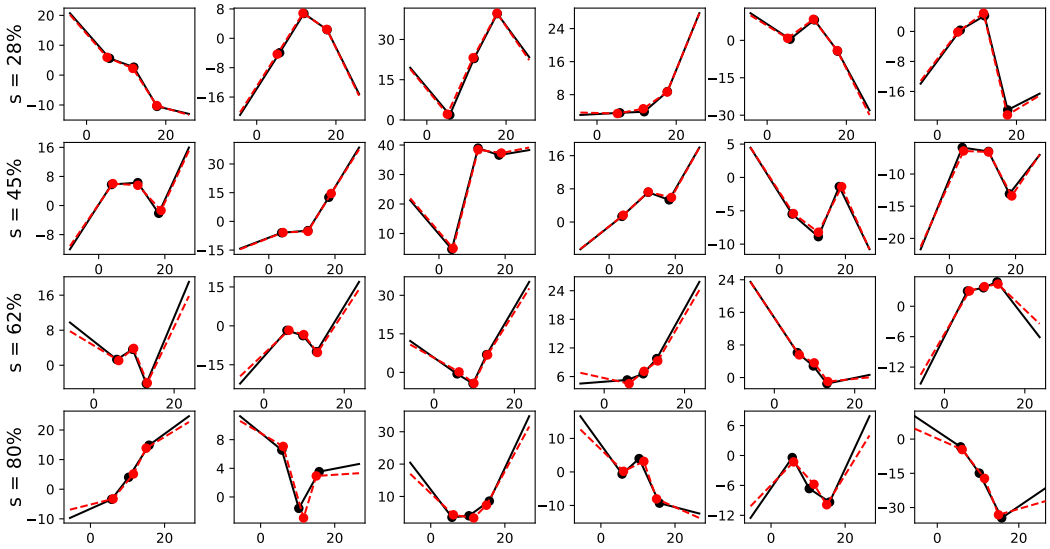

Figure 7.1: True population trajectories (black) and their estimation (dotted red) for data sets with $K = 3$ breaks in experiment (A), with 400 subjects, 12 observations per subject on average and 40% missing data. Each row shows the $d = 6$ components of a population trajectory, with observations sampled with noise level s. In all the similar figures presented in this chapter, the x-axis represents time and the y-axis the feature value.

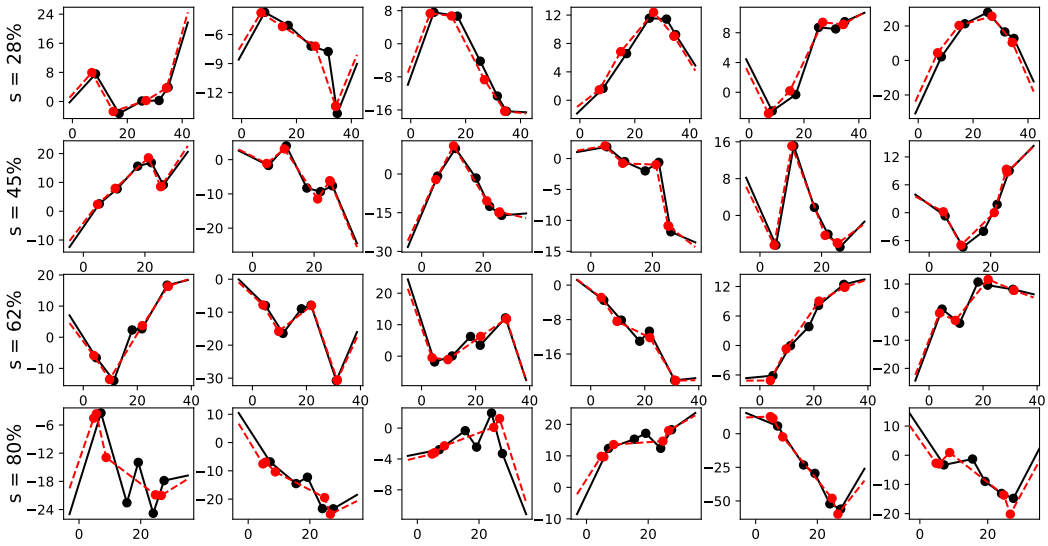

Figure 7.2: True population trajectories (black) and their estimation (red) for data sets with $K = 5$ breaks in experiment (A), with 400 subjects, 16 observations per subject on average and 40% missing data.

Population trajectory estimation. We first investigate the estimation of the parameters and the shape of the population trajectory. Figures 7.1 and 7.2 show some of the generated population trajectories and their estimator for experiment (A). Additional figures can be found for the other experiments in Appendix 7.D. It can be seen that, even for high noise levels and $N = 200$ samples, the models with $K = 3$ breaks are robustly estimated: the breakpoints are correctly identified, and the estimated trajectory matches the ground truth. For higher numbers of breaks ($K = 5$ in the figures), the overall shape of the trajectory is recovered, but some breakpoints are estimated very close to each others, which creates "phantom" trajectory pieces containing very few observations. The mismatch between the estimated breakpoints and their true position prevents from retrieving the true trajectory.

Two criteria can be used to measure estimation performance and the overall fit to the data. First, we compute the estimation error for each of the trajectory parameters (p_0, v, t_B). For each noise level and each value of K, average these errors over the ten i.i.d. generated data sets. The result is shown in Figure 7.3 for experiments (A) and (B). The error coherently increases with the noise level and the number of breaks. It degrades most significantly beyond $K = 4$ breaks. The inaccuracy on the breakpoint position causes the speed vectors to be estimated on time intervals mixing different pieces of the true trajectory, which only makes the estimation more difficult. However, as we saw on the trajectory plots, the estimation error remains relatively low for $K \leq 3$ and $s \leq 62\%$. In the case of experiment (B), increasing the number of observations per subjects yields a performance improvement, and in particular allows for a better estimation at the largest noise level $s = 80\%$.

Second, the performance of the optimization procedure can also be assessed by considering the observation's log-likelihood, which are obtained by Bridge Sampling. By comparing the marginal log-likelihoods $\log p(y \mid \theta^*)$ and $\log p(y \mid \widehat{\theta})$ for the true parameters θ^* and the estimated parameters $\widehat{\theta}$, we can assess the extent to which the MCMC-SAEM algorithm recovers the global maximum of the MAP optimization problem: if the global optimum were reached, we would have $\log p(y \mid \theta^*) \leq \log p(y \mid \widehat{\theta})$. Figure 7.4 shows the difference between these two quantities for each experiment. Higher number of breaks cause a negative gap, meaning that the global optimum is not reached. This is especially true for small noise levels, in experiments (A) and (A') where small values of K are generated with fewer observations. Higher noise levels increase the estimated log-likelihood w.r.t. the true log-likelihood. This can be understood as the loss of signal related to a large portion of noise: a higher noise in the data means that the observation's true distribution is very diffuse, and can thus be well approximated by a wider range of population trajectories.

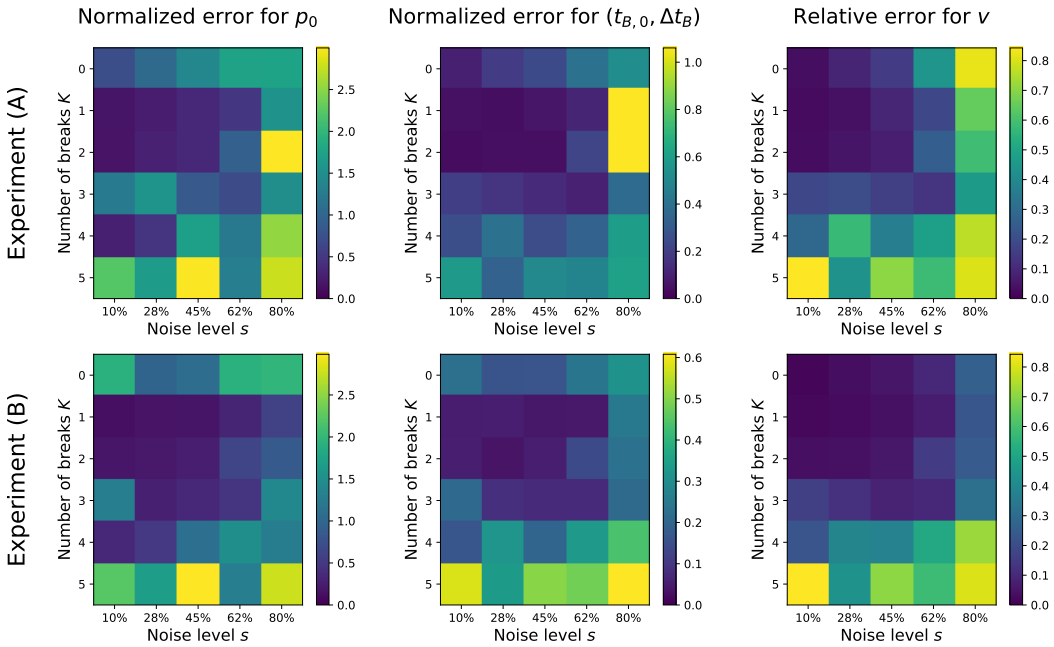

Figure 7.3: Estimation error for the model parameters, averaged over the ten data sets of each noise level s and number of breaks K. The parameters p_0 and t_B being shift parameters, comparing their relative errors makes little sense. Instead, we compute for the normalized error $\|\widehat{p}_0 - p_0\|_{\sigma_w}$, where each coordinate k is rescaled by $\sigma_{w,k}$. Similarly, we compute the errors on $t_{B,0}$ and the time increments $\Delta t_B = (t_{B,1} - t_{B,0}, \ldots, t_{B,K-1} - t_{B,K-2})$, normalized by σ_τ. The error shown for v is the relative error.

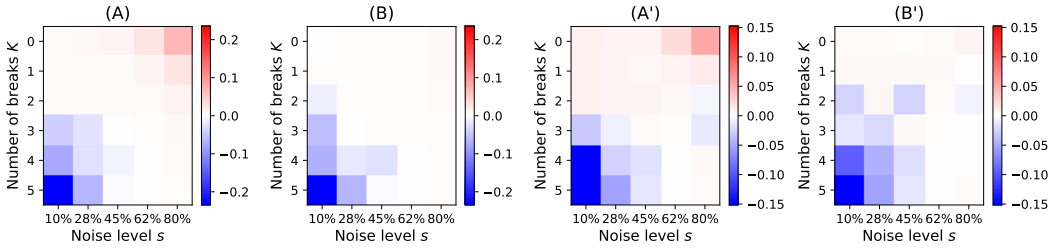

Figure 7.4: Average difference between the log-likelihoods of the estimated parameters and the true parameters, for each experiment, number of breaks K and noise level s. The log-likelihood differences are normalized by the average number of observations to allow for comparisons between different values of K in experiments (A) and (A').

Individual trajectories. We now examine the reconstruction of the individual trajectories. We show the trajectories obtained by using the posterior mean of the latent variables given the data: $\mathbb{E}[z_i \mid \theta, y_i]$. It is computed by running a MCMC for 10,000 iterations on the densities $(\xi_i, \tau_i \mid y, \theta)$ given in appendix 7.C, which also give the posterior mean of w_i through the expression $\mathbb{E}[w_i \mid y_i, \theta] = \mathbb{E}[\mathbb{E}[w_i \mid \xi_i, \tau_i, y_i, \theta] \mid y_i, \theta]$. Note that, from a Bayesian perspective, the uncertainty on the trajectories and the breakpoint positions could be easily obtained by computing the individual trajectory computed for each MCMC value of the latent variables. The results are shown in Figures 7.5 and 7.6 for a sample subject following the corresponding population distribution shown in Figures 7.1 and 7.2. The trajectories are well estimated for $K = 3$ breaks, even for large levels of noise. In particular, the positions of the breakpoints are well identified. In contrast, the breakpoints are much more difficult to estimate at $K = 5$ breaks, even for moderate levels of noise. This may be a combined consequence of the poor estimation of the population average breakpoints and the strong non-convexity of the posterior distribution landscape.

From a quantitative perspective, the estimation of individual trajectories can be evaluated by measuring the prediction error on missing data. Figure 7.7 shows the average errors obtained by estimating the observed and missing values with the posterior mean of the individual latent variables. Although missing data imputation is not the major focus of this chapter, note that the prediction could be easily extended to obtain confidence intervals, by predicting the missing data for each sampled value of the distribution $(z_i \mid y)$. We can see that, in every experiment, the estimation error mostly depends on the noise level, which is coherent. The error remains stable as the number of breaks K increases for experiments (A) and (A') where the average observation count per subject grows with K. In experiments (B) and (B') where this average count is the same for each value of K, the error increases with K; again, this is coherent as the estimation problem gets more difficult with larger values of K. It can be seen that the error on missing data is not larger than the error on the full data, which means that the missing data are estimated as well as observed data. Considering the figures 7.5 and 7.6 representing the estimated individual trajectories, this result is an understandable consequence of the missing data arrangement, which is chosen completely at random in these experiments. It is likely that more complex choices for missing data would lead to degraded performances; however no clear pattern in the missing data arrangement appeared in our application to the PPMI cohort apart from basic implications (e.g., patients having no treatment impact score because their treatment has not started).

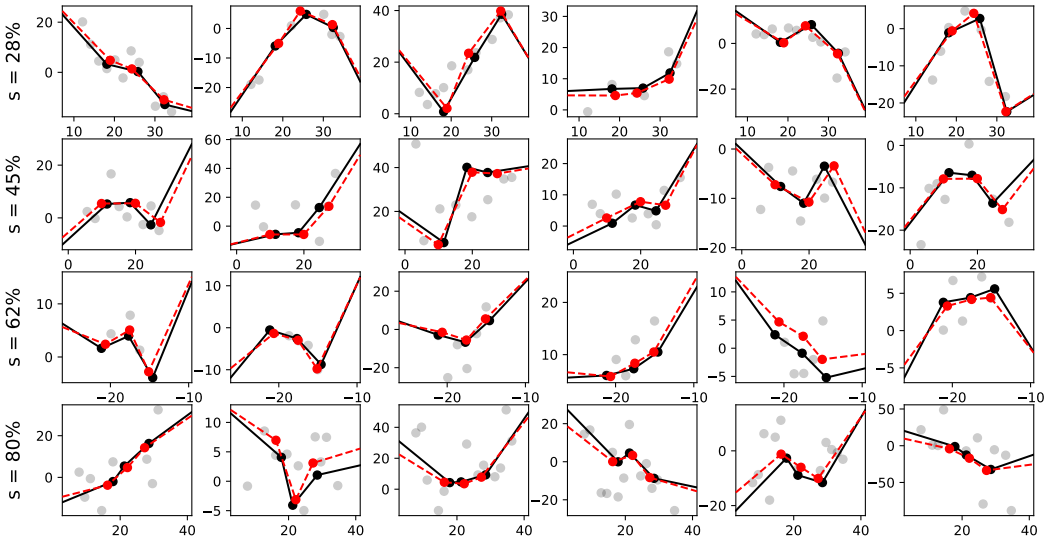

Figure 7.5: True individual trajectories (black) and their estimation (red) displayed with the observed data (gray) for data sets with $K = 3$ breaks in experiment (A). Each row shows the $d = 6$ components of a population trajectory, with observations sampled with noise level s.

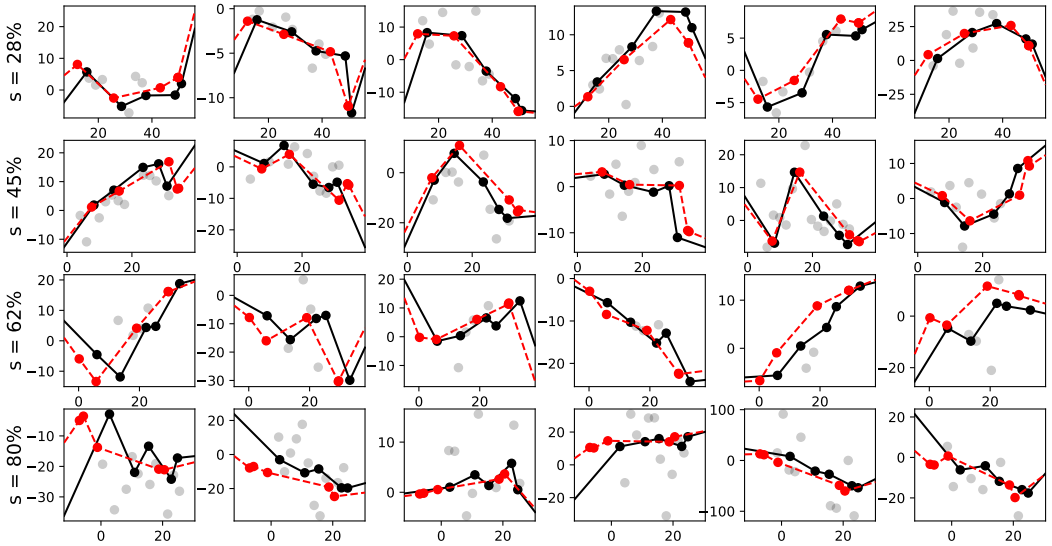

Figure 7.6: True individual trajectories (black) and their estimation (red) displayed with the observed data (gray) for data sets with $K = 5$ breaks in experiment (A).

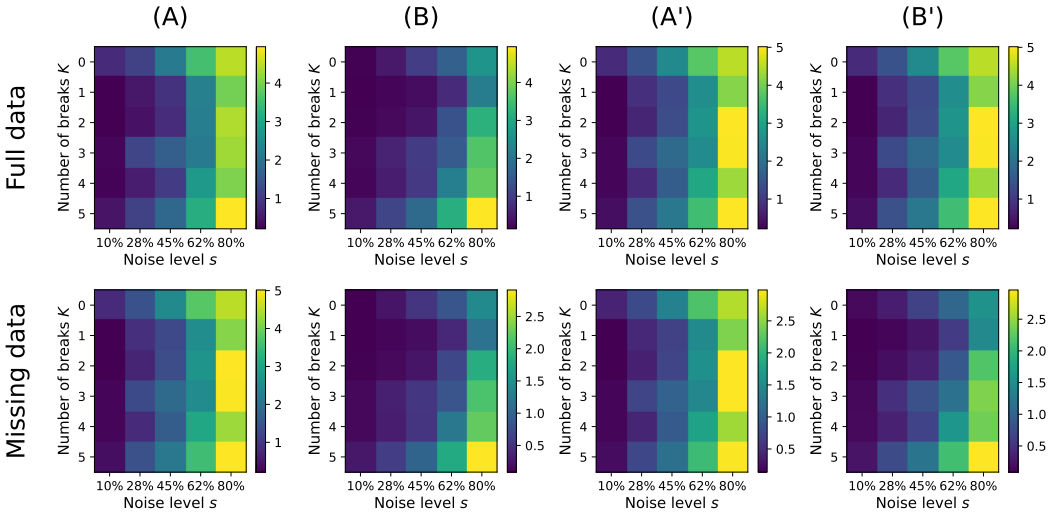

Figure 7.7: Average error for the individual trajectories. For each individual i and each time point t_{ij}, the values y_{ijk} are compared to the prediction $D_i(t_{ij})_k$ obtained using the posterior mean of z_i given y and the parameters estimated by the MCMC-SAEM algorithm. Each coordinate k is normalized by $1/\sigma_{w,k}$. The mean error of each individual is then averaged over the population. The top row shows the average reconstruction error on all values; the bottom row shows the error on missing values only.

Model selection. We compute the information criteria discussed in Section 7.3.2 from the marginal log-likelihoods $\log p(y \mid \theta)$ obtained by Bridge Sampling. We compare AIC and the hybrid BIC of Delattre et al. [2014] introduced in the previous section. In Figure 7.8, we show the proportion of correctly selected models for each number of breaks K and noise level s. As can be expected, the model selection performance degrades as the noise level and the number of breaks increases, and as the number of samples decreases. BIC_h performs on average better than AIC: in experiment (A), AIC selects the true model on 67% data sets, versus 73% for BIC_h, and similar results are observed on the other experiments. Furthermore, in the cases where the information criterion selects the correct model, the BIC_h does so with a higher margin than AIC: the average margin is 67% higher for BIC_h for experiment (A) (88% higher for (A'), 32% for (B) and 54% for (B')).

For $N = 400$ subjects (experiments (A) and (B)), it appears that the true model is selected relatively consistently as long as $K \leq 3$ and $s \leq 62\%$. On experiment (B) for $K \leq 3$, where data sets with smaller number of breaks have as many observations as five breaks models, the selection often finds the true model even for $s = 80\%$ noise in the data. The poor performances obtained for $K \in \{4, 5\}$ are a natural consequence of the bad estimation of the break times t_B: if the estimator at $K = 5$ has two breakpoints very close to each other, it does not bring any advantage over the same model with one of the neighbor breakpoints removed. Besides, the good performance obtained for $K \leq 3$ should not be understood as an absolute guarantee that, in practice, small selected models are necessarily the correct ones: it could be that the true model has a high number of breaks, which the selection procedure fail to detect. However, as we mentioned above, given the restricted number of observations available in practice, it seems unlikely that real world dynamics are best described by population trajectories with a high number of breaks.

As a last remark, it can be noted that, although the model selection procedure sometimes fails to detect the true number of breaks, it almost always selects an estimated value $\widehat{K}$ value of K such that $|\widehat{K} - K| \leq 1$. This result is shown on Figure 7.D.7.

Remark. The model selection with the standard BIC yields results comparable to BIC_h, in the sense that they often select the same model. However, when the BIC_h selects the correct model, it does so with a margin greater than the standard BIC's (from 10% greater for experiment (B) to 30% for experiment (A')). The good performance of both BICs (as opposed to AIC) can be partly attributed to their robustness to a possibly non-negligible optimization error.

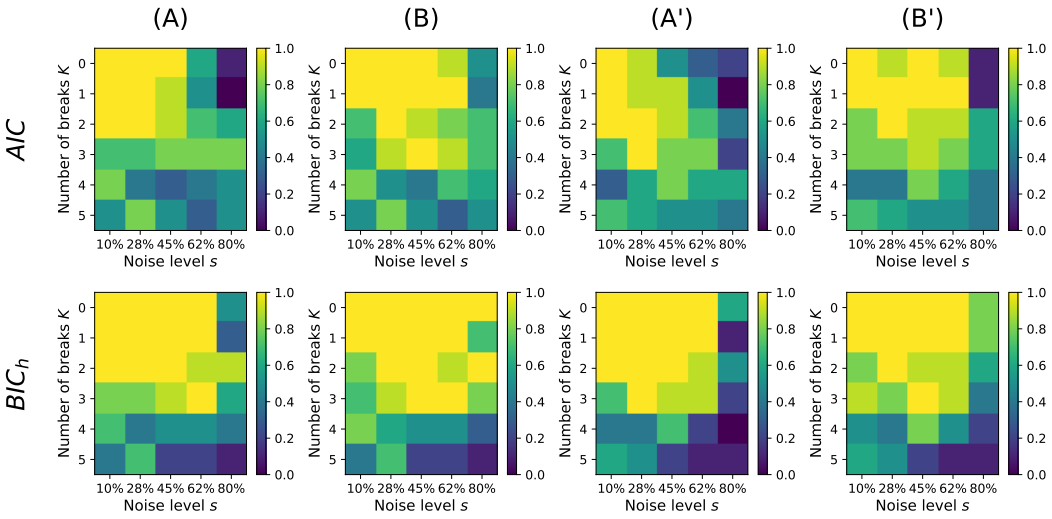

Figure 7.8: Proportion of correctly selected data sets for each experiment, respectively using AIC (top row) and BIC_h (bottom row).

7.4.2 Application to disease progression modeling

We now apply the proposed methods to the case of Parkinson's Disease (PD). We use a database from the Parkinson Progression Marker Initiative [Marek et al., 2011]. The PPMI consists in several cohorts of subjects followed over around a decade. Some subjects are affected by PD because of specific genetic factors, some for other unknown causes (in which case PD is called idiopathic). Some subjects, in the so-called prodromal phase, are in the course of developing PD and already exhibit pre-clinical symptoms. Finally, some subjects (with or without aggravating genetic factors) are unaffected by PD and form reference control groups. Overall all the 513 subjects studied here, the ages range from 31 to 92 years old, with 50% of subjects having between 55 and 69 years old at their first examination.

Most of the subjects affected by PD undergo a treatment aiming at compensating the loss of dopaminergic neurons that characterizes PD. Dopaminergic neurons are in charge of synthesizing of the dopamine used in the brain; their death causes most of the symptoms observed in PD. Dopaminergic treatments supply dopamine to the brain by injecting a dopamine precursor (e.g., L-dopa) in the blood, which is then able to cross the blood-brain barrier. Another category of treatments relies on dopamine agonists, chemical compounds that activate dopamine receptors.

The subjects affected by PD often start a treatment shortly after their situation starts deteriorating, but this delay may vary from one individual to another, with some subjects preventively starting a treatment while still having stable symptoms. In this chapter, the evolution state of PD is measured by two types of clinical scores. First, the Symbol Digit Modalities Test (SDMT) and the Montreal Cognitive Assessment (MoCA) measure the subject's cognitive abilities, which decline as PD progresses. Second, the Unified PD Rating Scale (UPDRS) measures a wide range of aspects and symptoms in the patient's daily life, some being during the examination. The PPMI relies on a version of the UPDRS produced by the Movement Disorder Society, called the MDS-UPDRS. It consists in three series of questions and tests on 1) non-motor aspects of experiences of daily living, 2) motor aspects of experiences of daily living, and 3) a motor examination. The last series of tests is performed both when the patient's treatment is active and when its effect has faded. The MDS-UPDRS results are aggregated into four clinical scores which measure the seriousness of the symptoms: one score for each of the two first series, and two scores for the third series, one "on treatment" and one "off treatment". These four scores increase with the symptoms' gravity, while the two first cognitive scores decrease with the cognitive decline.

The evolution of a patient's state can undergo several changes depending on the state of the disease and the treatment effect. It may first start deteriorating, then stagnate due to the effect of a treatment, and start a second decay phase after the treatment's impact weakens. In the light of the segmented longitudinal disease progression modeling framework introduced above, we wish to determine to what extent the population dynamics in the PPMI cohorts can be described by an average piecewise linear population trajectory, and how many breaks this trajectory would require accounting for the population variability.

We apply the estimation procedure to several cohorts. In each case, we run the MCMC-SAEM algorithm for 200,000 iterations, and perform Bridge Sampling with 500,000 MCMC samples (for each of the two chains) and 500,000 importance samples. Given the lack of clear tendency in subjects with few observations, we restrict each cohort to individuals having a minimal number of observations (at least 10 for large cohorts ($N \geq 200$), and at least 8 for smaller cohorts). We estimate models from $K = 0$ to $K = 5$ breaks, and select the model which achieves the highest BIC_h. We repeat this procedure for several groups of subjects:

1. subjects with PD with genetic factors (133 subjects after observation count filtering),
2. subjects with idiopathic PD (376 subjects after filtering)
3. subjects with PD (either with genetic factors or idiopathic) (458 subjects),
4. subjects in the above cohort that follow a treatment (dopamine agonist or dopaminergic treatment) for at least 90% of their observations (135 subjects),
5. subjects in the prodromal phase, which in a large majority have not started following a treatment (62 subjects).

For each experiment, the Bridge Sampling variances on the marginal log-likelihoods is much smaller than increments in the BIC_h, and much smaller than the margins between the selected model and the next ranked competitor. In each case, the proportion of unexplained variance lies between 20% and 30%, which in the equivalent setup for synthetic data allows for consistent results in terms of estimation and model selection.

Idiopathic and genetic factors cohorts. The selected models have respectively two and three breaks for the idiopathic and genetic factors cohorts, and two breaks for the combined cohort. The population trajectories are shown in Figure 7.9. It can first be noted that the three trajectories have similar overall shapes, with a degradation phase followed by an improvement phase, and a second degradation phase. The resemblance between the trajectories of the idiopathic cohort and the grouped cohort is coherent with the dominant proportion of the idiopathic subjects in the data set. The second piece in the genetic cohort contains only 4% of the observations, the remaining three pieces having the same profile as for the idiopathic cohort. However, in contrast to the idiopathic cohort, the breaks in the genetic cohorts are very distant from one another, with the two last breaks positioned at relatively advanced ages. This phenomenon relates to a very high concentration of individual trajectories: on average, in the genetic cohort, each individual has of its 95% observations in one trajectory piece (which varies from one subject to another, as the most loaded piece gathers only 56% of the observations). This might be a consequence of the restricted number of subject in the genetic cohort. Note that this phenomenon has a lower amplitude in the idiopathic cohort, with each individual having 81% of its observations in one piece on average. However, in both experiments, a large majority of subjects have almost all their observations in only two pieces.

It can be noted that the cognitive clinical scores (SDMT and MoCA) rarely exhibit strong ruptures in individual trajectories. This is coherent with the fact that the treatments used against PD symptoms do not affect the cognitive decline. Most individual trajectories are characterized by a more or less steady decline. This observation also holds for the other cohorts considered in this study.

The estimated average trajectories obtained here thus may not represent the evolution of every patient. Each patient is located in a portion of the population trajectory best corresponding to their evolution. This fact is best interpreted by looking directly at the estimated individual trajectories. In Figure 7.10, the estimation results for four representative subjects of the idiopathic cohort are shown. Individual (a) has points in each of the pieces, and the latent trajectory provides a good description of their evolution: at first, the disease symptoms are worsening; then, around 78 years old, the MDS scores improve for a short period; finally they worsen again in the last observations. Individual (b) only has points in the two last pieces, which again provides a good description of their observations, with a degradation phase on MDS2 and MDS3 (off treatment) followed by a progressive aggravation of the symptoms. Individuals (c) and (d) have all their observations either in the first or in the last piece of the trajectory, and are characterized by a continuous degradation of the clinical scores.

Two remarks must be made here. First, the individual trajectories displayed here (as with the other cohorts) are shown to illustrate our point, because their observations and latent trajectories are relatively simple to interpret. Some other individuals in the same data set have much more erratic evolutions, with no coherent tendencies from one feature to another. The time evolution of these individuals would however likely be difficult to account for, even if we used a more sophisticated model. Second, on some individuals the estimated trajectories describe the evolution of some scores better than others. This is a natural consequence of their respective weights in the likelihood: if a score has many missing data points and shows a trend opposite to the trend of other features, the estimated trajectory will poorly fit this specific score. This observation holds similarly for other disease progression models taking stock on the framework of Schiratti et al. [2015].

Overall, the selected models provide a coherent description of the population average trajectory and the individual level variability. However, its interpretability is not as straightforward as the initial model formulation might suggest. For a non-negligible proportion of subjects, the very strong fluctuations from one examination to another do not exhibit any coherent trend or structural break; the population average may be partially biased by these subjects. When clear breaks are present

in the observations, they are often retrieved relatively accurately - in particular when the subjects start a treatment and the symptoms stop degrading.

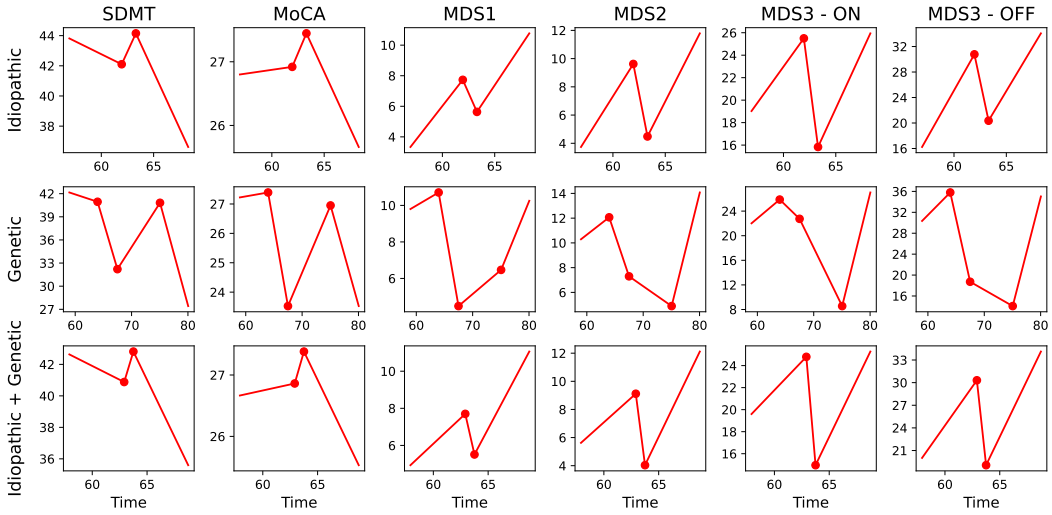

Figure 7.9: Population trajectories obtained on the idiopathic cohort (first row), the genetic factors cohort (second row) and the grouping of both (third row). Each column shows the time evolution of a clinical score. The SDMT and the MoCA degrade as PD progresses, and the four MDS scores increase as PD progresses.

Subjects with high proportion of treatment. The best model for the subset of individuals following a treatment for at least 90% of their observations has $K = 0$ breaks, with an affine trajectory decreasing for the cognitive scores (almost stagnating for the MoCA) and increasing for the MDS scores. This result is coherent with the fact that, for many individuals, the symptoms evolution after starting the treatment is relatively monotonous. For some individuals, the symptoms stagnate after starting the treatment and their trajectory can be well described by a flat line; for some individuals the treatment has little effect, and their trajectory approximately follows an increasing line. This result does however not fully account for the real disease progression of patients undergoing a treatment. The expected trajectory should *a priori* feature an initial stagnation or improvement phase, followed by a degradation phase. This evolution is clearly present in some subjects, but enough to select the one break model, which does precisely show such an evolution. For comparison purposes, we show in Figure 7.11 both the $K = 0$ and the $K = 1$ population trajectories. Figure 7.12 shows selected trajectories with three individuals selected to represent distinct behaviors. Individual (a) has an evolution under treatment in two phases, with first a slight improvement of the MDS scores, followed by a degradation. Individuals (b) and (c) have steadier evolutions, with a continuous aggravation of the symptoms for individual (b) and a relative stagnation for individual (c), with some improvements on the MDS3 scores. The model with $K = 0$ breaks models individuals (a) and (b) relatively well by adjusting the slope to fit the data. On individual (c), it produces an increasing trend, whereas the model with one break produces an overall decreasing trend. By definition, the model with $K = 0$ breaks can only change the magnitude of the slope, and not its sign; this constraint shows its limitations in this example.

Prodromal cohort. Finally, we study the results for the subjects in the prodromal phase of PD. The best selected model has $K = 2$ breaks. Its population trajectory is shown in Figure 7.11. It shows a progressive degradation of the clinical scores, which accelerates across time. On average, each individual has more than 98% of its observations in only two of the three pieces; the third piece only gathers 5% of the observations. It is thus relevant, as with the high treatment subgroup, to also show the estimated model at $K = 1$ for comparison purposes. Three selected individual trajectories are shown in Figure 7.13. The evolution of individual (a) undergoes a clear rupture at

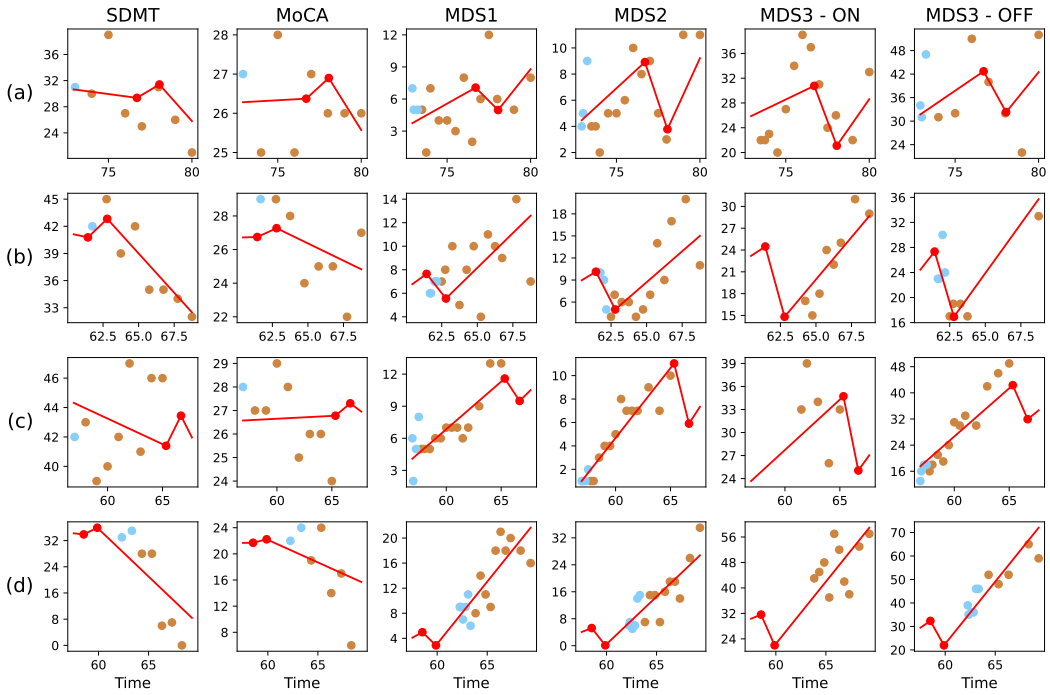

Figure 7.10: Individual trajectories of selected subjects of the idiopathic cohort. Each row represents the time evolution of a subject, and each column represents a clinical score. The measures clinical scores are displayed in blue and brown, with blue points denoting examinations where the subject is not receiving any treatment, in contrast to the brown points, which correspond to the subject receiving any treatment. The estimated latent trajectory is displayed in red. Note that no blue point is shown on the MDS3 - ON, which examines subjects while their treatment is active.

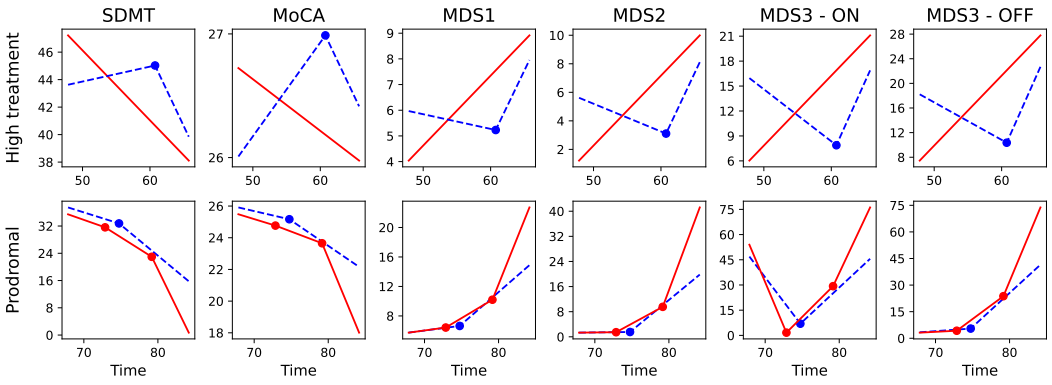

Figure 7.11: Population trajectories obtained on the combined idiopathic and genetic cohorts for subjects mainly undergoing a treatment for most of their observations points (first row), and the prodromal cohort (second row). Both rows show the best selected model (in red), and the model at $K = 1$ break (in dotted blue) for comparison purpose. The MDS3 (ON) column has a very large proportion of missing values on the prodromal cohort (most of the subjects are not undergoing a treatment); it thus does not have a population trajectory. As a consequence, the population trajectory in the fifth column of the prodromal cohort (in particular the first piece), is estimated from almost no data point.

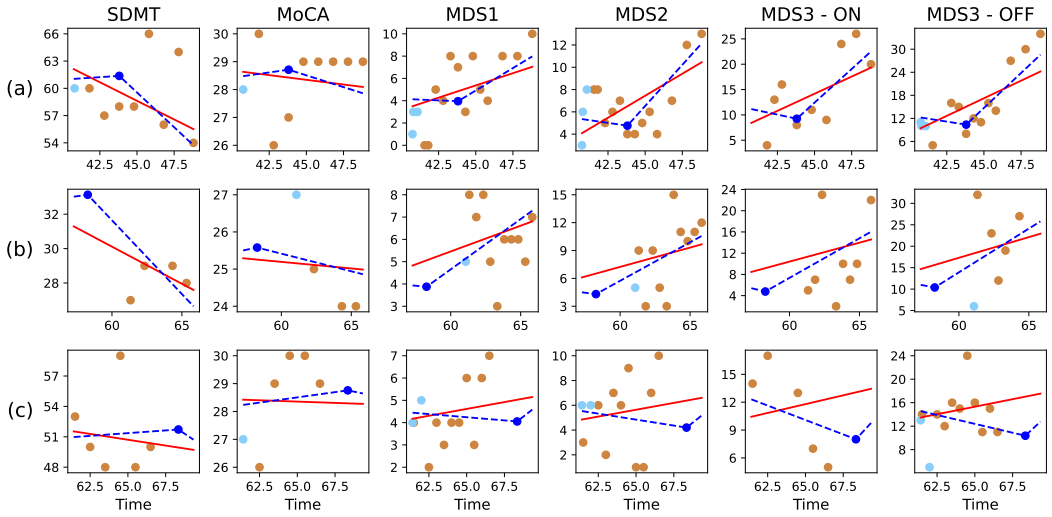

Figure 7.12: Individual trajectories of selected subjects following a treatment on most of their observations. Each row represents the data and trajectories of an individual. The estimated trajectories for $K = 0$ (in red) and $K = 1$ (in dotted blue) are both displayed for comparison purpose.

70 years old, where the MDS scores start degrading. Individual (b) shows a regular aggravation of the MDS scores, and its observations are placed in a single trajectory piece. Individual (c) does not show signs of degradation; all of its observation are coherently placed in the first piece of the population trajectory. For most of the individuals in the prodromal cohort, the time evolution has a profile which can easily be interpreted. The pieces of the population trajectory correspond to true distinct stages of the disease. In contrast, the interpretation of the population trajectories is more difficult for the genetic and idiopathic cohorts, as the time evolution of each subject results of the combined effects of two dynamics: the disease progression and the response to the treatment. Since each patient starts undergoing the treatment at a different stage of the disease, its overall time evolution has a stronger variability with respect to the population average.

Remark. In this study, we did not experiment on the cohorts of healthy subjects (with or without aggravating genetic factors). For the clinical scores considered in this chapter, most of the individuals in these two cohorts show little or no sign of degradation across time.

7.5 Discussion

7.5.1 Practical considerations on the selection of K

Aside from the models selected by information criteria, the overall difficulty of the inference problem for segmented longitudinal models invites considering alternate indicators to better understand the estimation results.

Occupation rates. As seen on synthetic data experiments, the algorithm may produce breaks very close from one another, or trajectories with certain pieces containing very few observations after time reparameterization. These possible issues can be easily detected in what could be called the occupation rates of the model, i.e., the proportion of observations falling in each trajectory piece. The experiments on both synthetic data and the PPMI cohort show that occupation rates are a relevant tool to understand the output of the estimation algorithm. In particular, if the BIC_h selects models with breaks that are almost empty, it may be coherent from a practical perspective to look at the model with one less break instead.

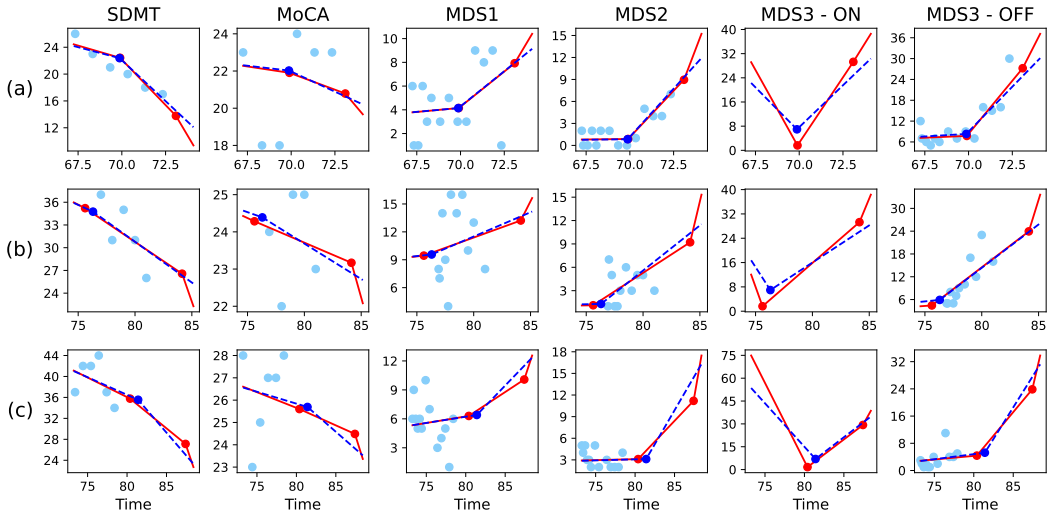

Figure 7.13: Individual trajectories of selected subjects from the prodromal cohort. Each row represents the data and trajectories of an individual. The estimated trajectories for $K = 2$ (in red) and $K = 1$ (in dotted blue) are both displayed for comparison purpose. The MDS3 (ON) column has no observations, as the three individuals do not follow a treatment.

Small sample sizes. In the settings considered in this chapter, the data is available in sufficient proficiency that the estimation can be performed on the model parameters with little uncertainty, apart from the optimization error when the number of breaks gets large. However, in cases where the number of observations is more restricted (typically 50 subjects or fewer), the uncertainty on the trajectory parameters becomes non-negligible, and the performance of information criteria degrades accordingly.

In this setting, considering the posterior distribution of the speed vectors v_k given the observed data y_{obs} allows quantifying this uncertainty. In particular, if the posterior distribution $(v \mid y, \theta)$ produces breaks forming a very low angle, removing the phantom break may lead to a better data description. The similarity between consecutive speed vectors v_k and v_{k+1} can be measured by computing the Mahalanobis norm $\mu_k^\top \Sigma_k^{-1} \mu_k$, with $\mu_k = \mathbb{E}[v_{k+1} - v_k \mid y, \theta]$ and $\Sigma_k = \text{Cov}(v_{k+1} - v_k \mid y, \theta)$. These means and covariances can be computed simply, by running a joint MCMC over the individual latent variables and the parameter v. In other words, the Mahalanobis norm computes the difference between the slopes of two consecutive pieces, rescaled by the uncertainty on the joint posterior distribution of the slope vectors. Two consecutive vectors with very small Mahalanobis distance will thus either be very close, or have such a great uncertainty that their difference is not significant - if not both. In particular, trajectory pieces with fewer observations will have a greater uncertainty, and this similarity measure naturally encourages removing "phantom" pieces.

Although this criterion produces coherent and interpretable results on the small experiments we performed, it should only be used in practice as a helpful indicator rather than a decision-making tool. Deciding to merge pieces based on the Mahalanobis similarity between their slopes would require choosing a threshold beyond which pieces are considered distinct. The choice of this threshold is an ill-posed problem, as it depends on the degree of similarity considered meaningful by the user. Unlike in the classical framework of statistical tests, we are not testing for the equality of two estimated parameters using a restricted number of model samples - rather, we are comparing two posterior distributions, which can be sampled from as many times as necessary.

A more Bayesian approach to the idea proposed here would consist in performing a Bayesian equality test between v_k and v_{k+1}, i.e., computing the probability $p(v_k = v_{k+1} \mid y)$, using a spike and slab prior on v with a non-zero probability that $v_k = v_{k+1}$. The probability $p(v_k = v_{k+1} \mid y)$ would however be much more difficult to compute than the marginal likelihood we used in this study: in our specific case, $p(y)$ writes as a product of independent integrals in dimension $K + 1$, and we are able to compute each integral separately with great precision. This decoupling does

not work anymore here, as v is considered a random variable to be sampled from, and which is not independent of the z_i's.

7.5.2 Conclusion and perspectives

The results presented in this study show that segmented longitudinal models can be estimated for more than one break, large noise and significant proportion of missing data. The case of affine population trajectories allows for tractable computation of the marginal likelihoods, which allow for robust model selection using a hybrid BIC criterion. The estimation and model selection capacities are understandably limited by the number of observations per individual, and in real life scenarios considering models with four breaks or more are unlikely to be of practical relevance.

We showed that piecewise affine longitudinal trajectories can be used to describe the variability in cohorts of subjects affected by PD, or in the process of developing the disease. They provide an interpretable representation of the disease progression. This representation can be used to identify the impact of a treatment, measure the average evolution pace in each stage of the disease, and quantify the variability of this pace from one individual to another. This description is particularly relevant in the prodromal phase of the disease, before the patients start undergoing a treatment which slows the disease progression.

The proposed approach currently suffers from two inherent limitations to describe the impact of treatments. First, the description of the individual disease progression as a time reparameterization of a template trajectory does not allow changing the sign of the slope vectors (which are only multiplied by $e^{\xi_{i,k}}$) from an individual to another. This constraint is not adapted to describe the impact of a treatment: for some individuals the situation improves, and for others it keeps degrading, although more slowly. In that regard, the model of Severson et al. [2021] allows for a more flexible representation in terms of slopes. However, their protocol selects a model with seven distinct stages of disease progression, which may seem excessive from the perspective adopted in the present chapter, where the pieces correspond to interpretable stages of the disease. Finding a way to add some flexibility in the population model would improve the model's representation capacity. A possible solution could be to add a second transformation after the time reparameterization, which could affect the slopes.

Second, our model does currently not rely on the treatment information provided on each subject (which treatment is used at which time points), and we focused on identifying its impact on the individual and population trajectories. However, as mentioned in the previous section, it is difficult to find a coherent population trajectory that simultaneously accounts for the evolution of patients that start a treatment at different disease stages. A possible direction would be to include the treatment information as a known variable and use it in the individual deviation from the population average. Learning the relation between this variable and the disease progression at each stage of the disease would provide an interesting perspective on the treatment impact.

7.A Prior distribution

In the Bayesian setting considered in this chapter, we need to specify a prior distribution for each of the model parameters $\theta = (\bar{p}_0, \bar{t}_B, \bar{v}, \sigma^2, \sigma_w^2, \sigma_\tau^2, \sigma_\xi^2)$. The prior distributions for the trajectory parameters $(\bar{p}_0, \bar{t}_B, \bar{v})$ are defined in section 7.3.1 by the Gaussian distributions with large variances $(s_{p_0} = s_{t_B} = s_v = 500)$ and mean zero. In contrast, the variance of the population latent variables is set to a much smaller value $(\sigma_{p_0} = \sigma_{t_B} = \sigma_v = 0.1)$, in order to ensure that the exponentialized model remains close to the initial model.

Inverse Gamma priors $\Gamma^{-1}(m, \beta)$ are used for the variance parameters, with m the shape parameter and β the scale parameter. We take uninformative priors for σ_τ, σ_w and σ (with the hyperparameters $\beta_\tau = \beta_w = \beta = 1$ and $m_\tau = m_w = m = 6$). As mentioned in section 7.3.1, experiments on real data showed that, without regularization, the variables $\xi_{i,k}$ may take large values on some trajectory pieces, and thus yield individual trajectories with a shape very different from the population trajectory's. This behavior is undesirable: it often causes some trajectory pieces to be much longer than in the population trajectory, and hence a large amount (if not all) the observations may fall into a single trajectory piece, although the trajectory may clearly

exhibit a slope change. This phenomenon might be a numerical issue related to the strong non convexity of the posterior landscape of the latent variables given the observed data. It can be controlled through the variance σ_ξ: by imposing a strong regularization on σ_ξ, the acceleration factors are constrained to take smaller values. The individual trajectories are thus more similar to the population trajectories, and they better describe individual breaks. In particular, the occupation rates are spread more equally between the pieces when applying a strong regularization on σ_ξ. In the PPMI experiment, we use $m_\xi = N/2$ and $\beta_\xi = 1$. This prior depends on the number of samples, which does not fully fit into a purely Bayesian perspective, but allows obtaining coherent results for different numbers of samples.

When investigating this issue, we experimented with different values of m_ξ, and we performed the same experiment with $m_\xi = N/4$ and $m_\xi = N$. The first one produces results very similar to $m_\xi = N/2$. The main difference is that the model at $K = 3$ breaks is selected over $K = 2$, by a small BIC_h margin. However, the $K = 3$ model has a very short piece containing less than 4% of all observations, which makes it very close to a model with $K = 2$ breaks. The models obtained with $m_\xi = N$ had a very small value for σ_ξ, so that the individual trajectories would not vary from the population trajectory. The BIC_h selected $K = 7$ breaks, which is not very surprising: when the individual trajectories have no flexibility (apart from space and time shifts), trajectories with the highest number of pieces are more likely to produce time and space shifts that fit the individual observations.

7.B Sufficient statistics and MAP formulas

The model variables are divided between the parameters $\theta = (\bar{p}_0, \bar{t}_B, \bar{v}, \sigma^2, \sigma_w^2, \sigma_\tau^2, \sigma_\xi^2)$, the individual latent variables $z_i = (\xi_i, \tau_i, w_i)$, the population latent variables $z_{pop} = (p_0, t_B, v)$ and the observations y_{ij}. For each observation y_{ij}, we denote as $\mathcal{D}_{ij}$ the set of non-missing coordinates. The model's complete log-likelihood of the observed data writes as follows, up to a constant c.

$$
\begin{aligned}
\log p(y_{\text{obs}}, z, \theta) &= \sum_{i=1}^{N} \sum_{j=1}^{n_i} \sum_{\ell \in \mathcal{D}_{ij}} \log p(y_{ij\ell} \mid z_i, \theta) + \sum_{i=1}^{N} \log p(z_i \mid \theta) + \log p(\theta) \qquad (\star) \\
&= -\sum_{i=1}^{N} \sum_{j=1}^{n_i} \sum_{\ell \in \mathcal{D}_{ij}} \frac{1}{2\sigma_\ell^2} (y_{ij\ell} - D_i(t_{ij})_\ell)^2 - \frac{1}{2} \log(\sigma_\ell^2) \\
&\quad - \frac{1}{2\sigma_\tau^2} \sum_{i=1}^{N} \tau_i^2 - \sum_{i=1}^{N} \sum_{k=1}^{K+1} \frac{1}{2\sigma_{\xi,k}^2} \xi_{ik}^2 - \sum_{i=1}^{N} \sum_{\ell=1}^{d} \frac{1}{2\sigma_{w,\ell}^2} w_{i\ell}^2 - \frac{N}{2} (\log(\sigma_\tau^2) + \log(\sigma_\xi^2) + \log(\sigma_w^2)) \\
&\quad - \frac{1}{s_{p_0}^2} \left\| p_0 - \bar{p}_0 \right\|^2 - \frac{1}{s_{t_B}^2} \left\| t_B - \bar{t}_B \right\|^2 - \frac{1}{s_v^2} \left\| v - \bar{v} \right\|^2 \\
&\quad - \frac{\beta_\tau^2}{2\sigma_\tau^2} - \frac{m_\tau + 2}{2} \log(\sigma_\tau^2) - \sum_{k=1}^{K} \left[\frac{\beta_\xi^2}{2\sigma_{\xi,k}^2} + \frac{m_\xi + 2}{2} \log(\sigma_{\xi,k}^2) \right] \\
&\quad - \sum_{\ell=1}^{d} \left[\frac{\beta_w^2}{2\sigma_{w,\ell}^2} + \frac{m_w + 2}{2} \log(\sigma_{w,\ell}^2) + \frac{\beta^2}{2\sigma_\ell^2} + \frac{m + 2}{2} \log(\sigma_\ell^2) \right]
\end{aligned}
$$

Note that t_B has $\max(K, 1)$ components: if $K = 0$ the variable t_0 is still needed in the expression of $D(t)$. Estimating the model parameters with missing data amounts to implementing the MCMC-SAEM procedure with the likelihood described above, deriving sufficient statistics from that only rely on the observed components. Denoting N_ℓ the total number of observations on feature ℓ, the sufficient statistics are given by:

$$S(y,z) = \begin{cases} S_1 = p_0 \\ S_2 = t_B \\ S_3 = v \\ S_4 = \frac{1}{N}\sum_{i=1}^N \tau_i^2 \\ S_{5,k} = \frac{1}{N}\sum_{i=1}^N \xi_{ik}^2 & (1 \le k \le K+1) \\ S_{6,\ell} = \frac{1}{N}\sum_{i=1}^N w_{i\ell}^2 & (1 \le \ell \le d) \\ S_{7,\ell} = \frac{1}{N_\ell}\sum_{i=1}^N \sum_{j|\ell \in \mathcal{D}_{ik}} (y_{ij\ell} - D_i(t_{ij})_\ell)^2 & (1 \le \ell \le d)\,. \end{cases}$$

And, given a value S of the sufficient statistics, the optimal value for θ write as:

$$\widehat{\theta}(S) = \begin{cases} \bar{p}_0 = S_1 \\ \bar{t}_B = S_2 \\ \bar{v} = S_3 \\ \sigma_\tau^2 = \frac{NS_4 + \beta_\tau^2}{N + m_\tau + 2} \\ \sigma_{\xi,k}^2 = \frac{NS_{5,k} + \beta_\xi^2}{N + m_\xi + 2} \\ \sigma_{w,\ell}^2 = \frac{NS_{6,\ell} + \beta_w^2}{N + m_w + 2} \\ \sigma_\ell^2 = \frac{N_\ell S_{7,\ell} + \beta^2}{N\ell + m + 2}\,. \end{cases}$$

These formulas, coupled with a MCMC procedure can be used directly in Algorithm 7.3.1 to estimate the model parameters.

7.C Conjugate posterior factorization for the space shifts

Simple computations from the expression $(\star)$ allow deriving the factorization:

$$p(w_i, \xi_i, \tau_i, y_i \mid \theta) = \prod_{\ell=1}^d p(w_{i\ell} \mid \xi_i, \tau_i, y_i, \theta) \times p(\xi_i, \tau_i, y_i \mid \theta)\,.$$

Let us denote $n_{i\ell}$ the number of time points for which the component ℓ is observed for individual i; let $D_i(t)$ be the latent trajectory for individual i without its space shift ($D_i(t)$ is defined using τ_i and ξ_i only), and let $\sigma_{i\ell}^2 = (n_{i\ell}/\sigma_\ell^2 + 1/\sigma_{w,\ell}^2)^{-1}$. We have

$$p(w_{i\ell} \mid \xi_i, \tau_i, y_i, \theta) = \mathcal{N}\left(\frac{\sigma_{i\ell}^2}{\sigma_\ell^2} \sum_{j|\ell \in \mathcal{D}_{ij}} (y_{ij\ell} - D_i^0(t_{ij})_\ell),\ \sigma_{i\ell}^2\right)\,,$$

and, with $\tilde{n}_i = n_{i1} + \cdots + n_{id}$:

$$p(\xi_i, \tau_i, y_i \mid \theta) = \frac{\sigma_{i1}\ldots\sigma_{id}}{\sigma_{w,\ell}^d} \frac{1}{(\sqrt{2\pi}\sigma)^{\tilde{n}_i}} \exp\left(\sum_\ell \frac{\sigma_{i\ell}^2}{2\sigma_\ell^4} A_\ell^2 - \frac{1}{2\sigma_\ell^2} B_\ell\right) p(\tau_i \mid \theta) p(\xi_i \mid \theta)\,,$$

with the additional notations

$$\begin{cases} A_\ell = \sum_{j|\ell \in \mathcal{D}_{ij}} (y_{ij\ell} - D_i^0(t_{ij})_\ell) \\ B_\ell = \sum_{j|\ell \in \mathcal{D}_{ij}} (y_{ij\ell} - D_i^0(t_{ij})_\ell)^2\,. \end{cases}$$

7.D Additional figures on synthetic data experiments

Figures 7.D.1 to 7.D.7 provide additional information on the synthetic data experiments.

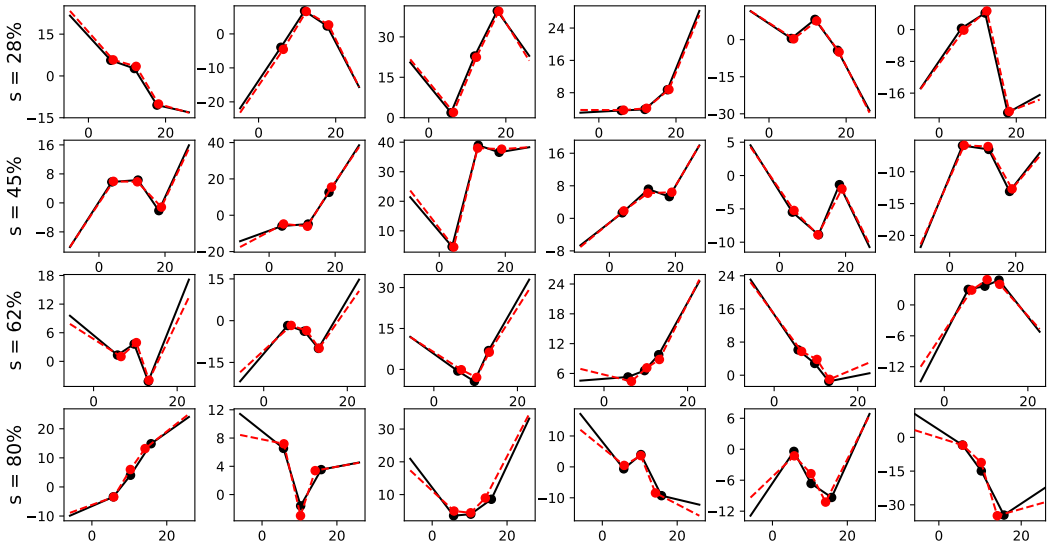

Figure 7.D.1: True population trajectories (black) and their estimation (red) for data sets with $K = 3$ breaks in experiment (A'), with 200 subjects and 12 observations per subject on average.

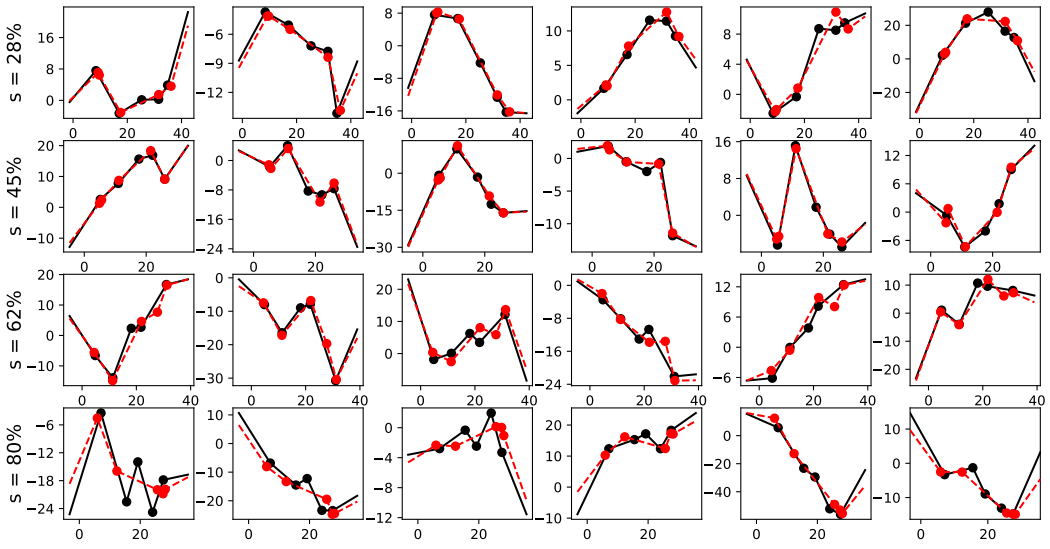

Figure 7.D.2: True population trajectories (black) and their estimation (red) for data sets with $K = 5$ breaks in experiment (A'), with 200 subjects and 16 observations per subject on average.

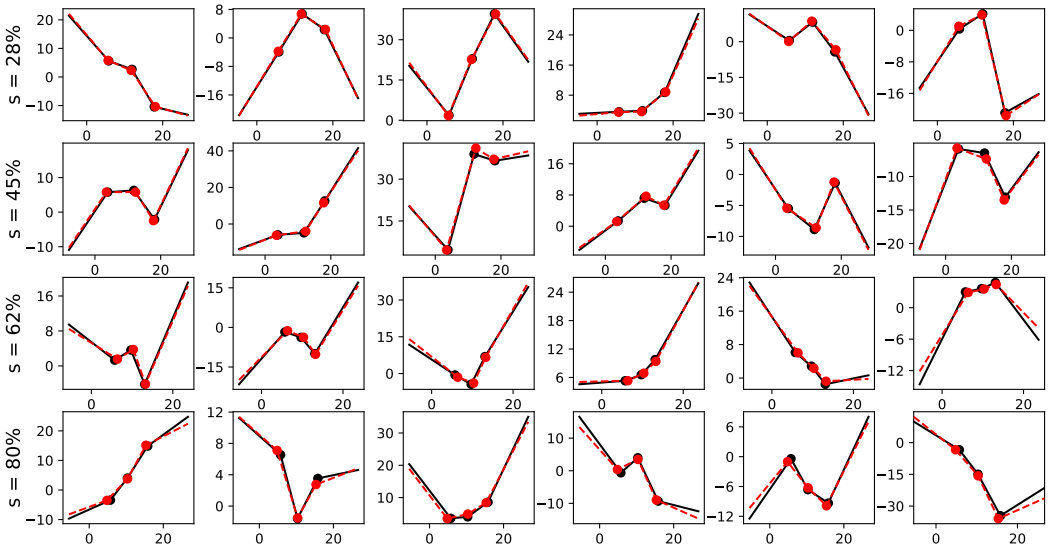

Figure 7.D.3: True population trajectories (black) and their estimation (red) for data sets with $K = 3$ breaks in experiment (B), with 400 subjects and 16 observations per subject on average.

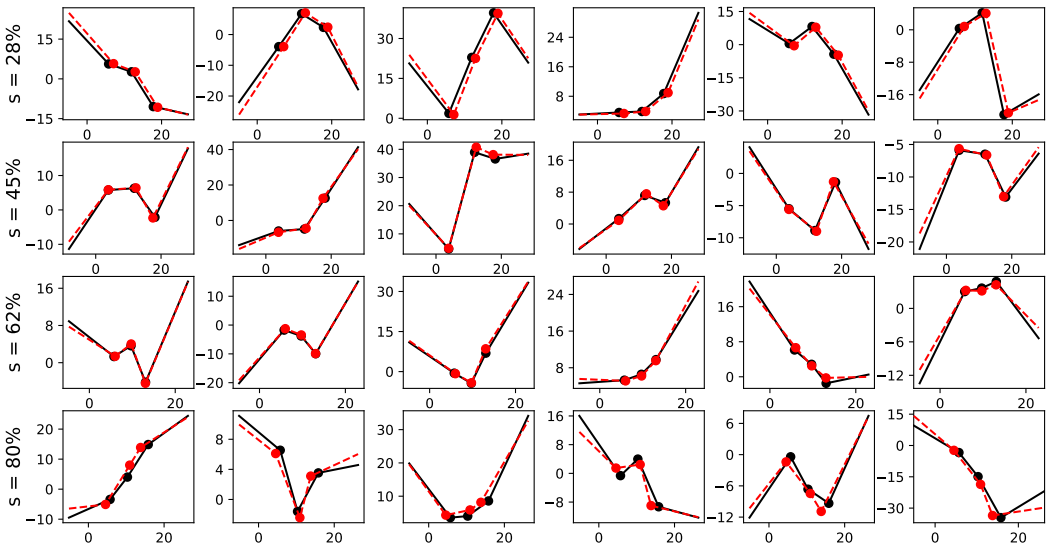

Figure 7.D.4: True population trajectories (black) and their estimation (red) for data sets with $K = 3$ breaks in experiment (B'), with 200 subjects and 16 observations per subject on average.

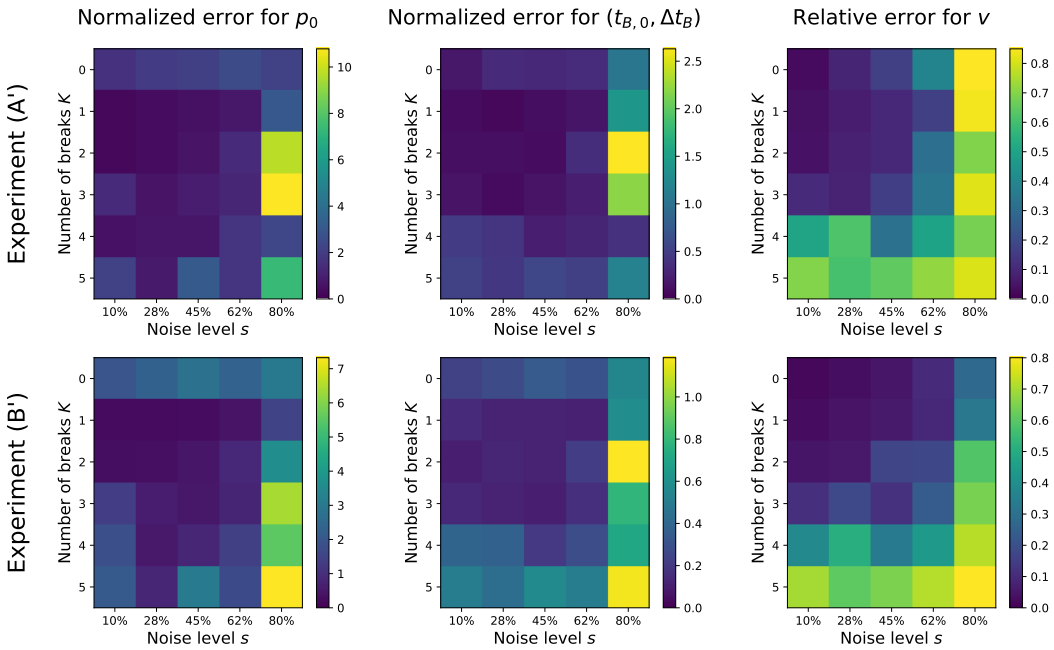

Figure 7.D.5: Estimation error for the model parameters, averaged over the ten data sets of each noise level s and number of breaks K, on experiments (A') and (B').

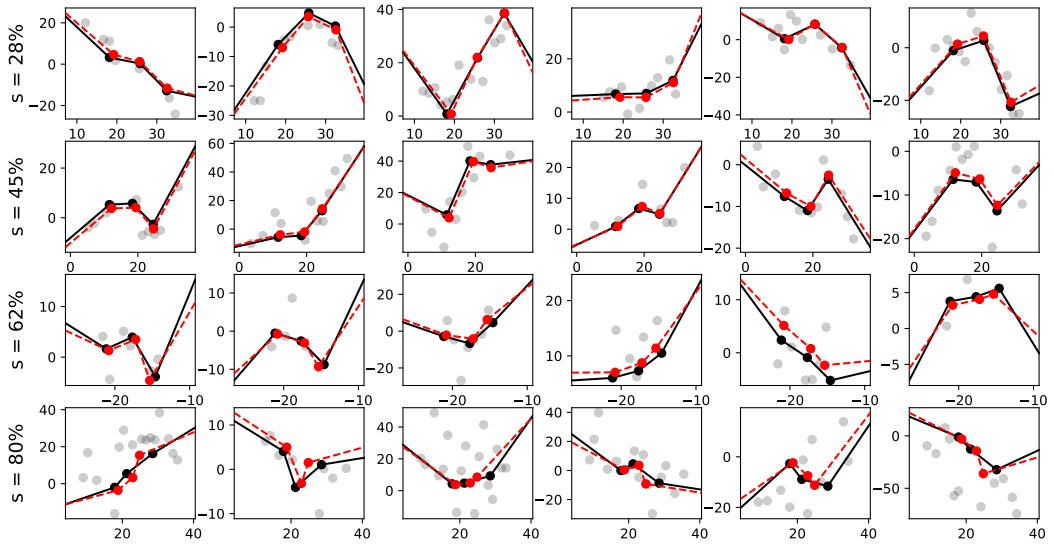

Figure 7.D.6: True individual trajectories (black) and their estimation (red) displayed with the observed data (gray) for data sets with $K = 3$ breaks in experiment (B).

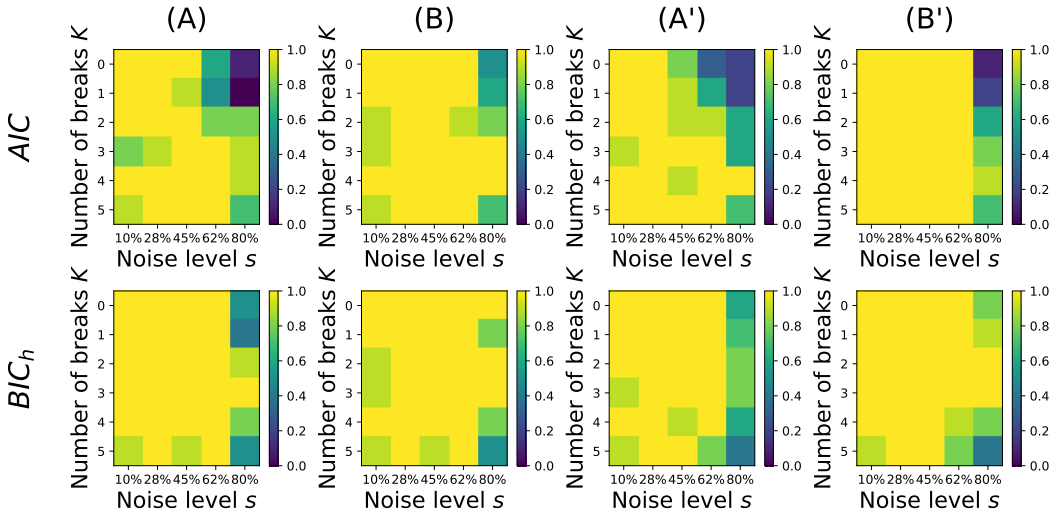

Figure 7.D.7: Proportion of data sets such that $\widehat{K}$, the selected value of K, is such that $|\widehat{K}-K| \leq 1$.

7.E Additional results on the PPMI experiment

Table 7.E.1 shows additional information on the estimation results. We show the standard deviation σ_ξ for each experiment, which quantifies to what extent the shapes of the individual trajectories may differ from the population average. We also give the unexplained variance ratio, which ensures that the model captures a significant proportion of the observed variability.

Figure 7.E.1 shows the BIC_h obtained for several values of K. Note that the curves are not always concave; this is partly due to the fact that the log-likelihoods from one value of K to another are not always increasing at $K \in \{4, 5\}$. This may be due to the optimization error, with the MCMC-SAEM not finding the optimal values of the break times. Another contributing factor may be that, in the hierarchical model considered in this chapter, increasing the number of parameters also increases the dimension of the latent space, which does not necessarily lead to a better marginal likelihood for the observed data.

	Standard deviation σ_ξ	Unexplained variance ratio
Idiopathic	0.20	25%
Genetic	0.15	26%
Idiopathic + Genetic	0.18	24%
High treatment	0.71	28%
Prodromal	0.38	23%

Table 7.E.1: For each group of subjects, we show a) the variability σ_ξ of the log-acceleration factors $\xi_{i,k}$, averaged over the pieces; b) the proportion of unexplained variance.

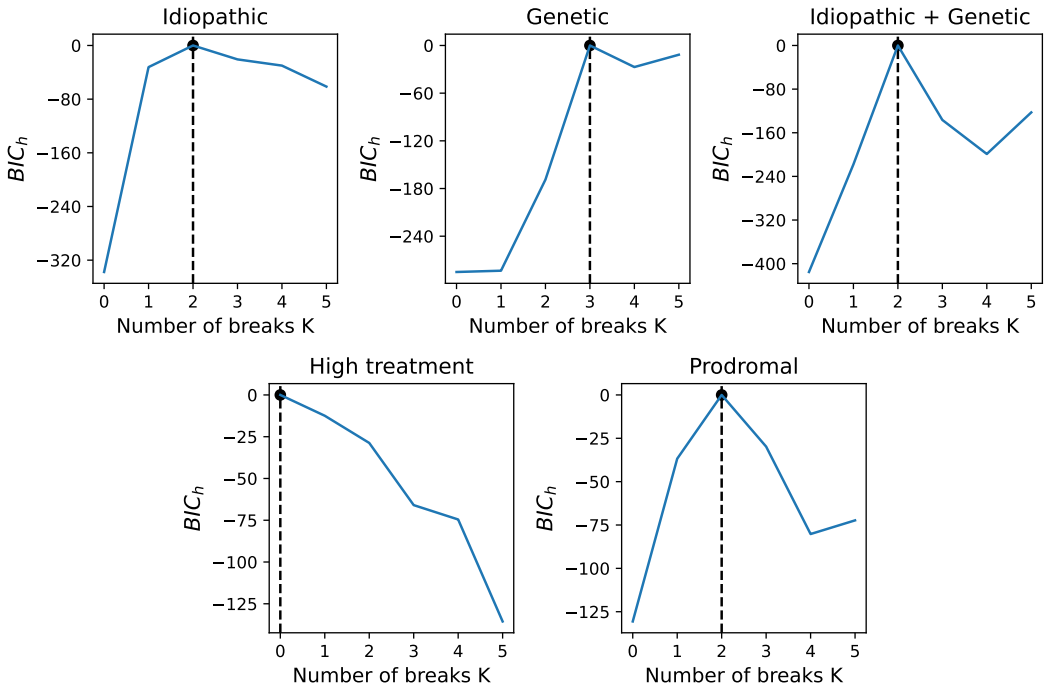

Figure 7.E.1: BIC_h values for the five experiments considered on every feature, depending on the number of breaks K of the considered models.

Chapter 8

Conclusion

We have proposed several contributions in population modeling for network analysis and longitudinal disease progression. This chapter concludes the thesis with general remarks, open questions and possible research perspectives.

Sparse low-rank modeling for network matrices. The parametric modeling approach proposed in Chapter 5 seems overall better suited than the variational approach of Chapter 4 to provide a meaningful description of the variability in populations of networks. While it does not specifically focus on matrix sparsity, sparsity is easily obtained by using Bernoulli sampling, and considering the low rank matrices as logits (i.e., numbers h such that the Bernoulli probability is given by $1/(1+e^{-h})$). This type of approaches can be compared to the highly interpretable models mentioned in the introduction, like the Exponential Random Graph Model. On the one hand, our approach is not well suited to analyze network parameters like the average degree or the clustering coefficient from a theoretical perspective: it seems difficult in general to reconcile the non-exchangeability of the networks nodes (the brain regions) with simple characterizations of network parameters. On the other hand, the low-rank approach provides a good compromise between modeling expressiveness and accounting for real-world network properties. The low rank representation reflects the modularity of the network and naturally lends itself to clustering.

Extending the sparse low-rank framework. In that regard, the sparse low-rank decomposition framework of Chapter 4 mainly acts as a denoising method rather than a predictive statistical model. Nonetheless, very similar computations can be carried out to shift our perspective and instead perform regression. As an example, if we want to model the brain structural connectivity matrix A_i of an individual i as a linear function of its age t_i, we could estimate the model $A_i = T + t_i V + \varepsilon_i$, imposing that T and V are both sparse with low rank. This problem can be solved using the Douglas-Rachford algorithm in a very similar way as in Chapter 4. Other cofactors like sex or brain damage could be included to compare between different types of populations.

From a different perspective, the main improvement of Chapter 5 with respect to Chapter 4 is to provide a data generation mechanism to explain the variation between the template and the observed data. An alternative direction for such a mechanism would have been a "sparse low-rank ICA" model, i.e., a model assuming that the observations are random linear combinations of low-rank templates. Formally, this would write as

$$A_i = a_i^1 T_1 + \cdots + a_i^K T_K \,,$$

with the (T_k)'s fixed sparse low-rank matrices and the (a_i)'s independent random coefficients. Such a model could be estimated using classical EM algorithms combined with sparse low-rank penalties, replacing the maximization step of the EM by the Douglas-Rachford algorithm (or only a few steps of the algorithm). This formulation is similar to the dictionary model proposed by D'Souza et al. [2018], except that the rank of the dictionary elements can be higher than one.

Exact estimation of the spectral model. An important open question left in Chapter 5 relates to the estimation of the model parameters. More specifically, we showed that our implementation of the MCMC-SAEM algorithm systematically over-estimates the variance of the eigenvectors, despite specific adaptations to prevent the variables from spreading across the Stiefel manifold. This estimation error may come from either the estimation of the normalizing constant of von Mises-Fisher distribution provided by Kume et al. [2013], or from the MCMC-SAEM algorithm itself. Regarding the first point, little improvement is in sight, as no tractable exact inference algorithm has yet been found for von Mises-Fisher distributions on Stiefel manifolds. The more recent work of Kume and Sei [2018] proposes an exact method for the special case of distributions on the unit sphere, but we could not find a simple way to adapt it to higher-dimensional Stiefel manifolds.

Regarding the second point, it is possible that the model structure is such that the MCMC-SAEM algorithm tends to land on poor local maxima of the likelihood. This hypothesis is hard to test in practice, as the likelihood itself cannot be computed explicitly due to the high dimension of the latent variables. In order to investigate this issue, alternative estimation procedures could be considered. As an example, a variational Bayesian approach could be proposed, approximating the posterior distribution of the latent variables with a simple expression, e.g., von Mises-Fisher distributions. Another option would be to use the recent developments on Bayesian inference on Stiefel manifolds to work on a fully Bayesian approach, and sample from the posterior distribution of the model parameters [Pal et al., 2020, Meng, 2021].

Dependency structure in the spectral model. An important assumption in the spectral model of Chapter 5 is the independence between the eigenvectors and the eigenvalues of network matrices. This hypothesis allows having a small number of interpretable parameters, but it is restrictive: in practice, it is likely that eigenvectors are strongly correlated to their eigenvalues. Two directions could be explored to account for this correlation in the model structure, with only slight changes in the MCMC-SAEM algorithm structure, exponentializing some parameters when necessary as in Chapter 7.

A first option would be to use the Cayley transform (or any other retraction) to define the distribution of the eigenvectors. Given a central point $M \in \mathcal{V}_{np}$, an eigenvector matrix could be drawn by sampling a random Gaussian vector $D \in T_M\mathcal{V}_{np}$ and computing $C_M(D)$. The tangent vector D and the eigenvalues vector $\lambda \in \mathbb{R}^p$ could be drawn simultaneously from a joint Gaussian distribution, which would induce a correlation between eigenvalues and eigenvectors. Another similar option would be to work in the tangent space at identity, and drawing $D \in T_{I_{np}}\mathcal{V}_{np}$ from a non-centered Gaussian distribution. The first approach is more interpretable, but makes the model non-exponential, whereas the second approach is somewhat less robust (as it could hardly model points far from I_{np}) but keeps an exponential model structure.

A second option would be to formalize the fact that knowing an eigenvector x_k imposes a strong constraint on the related eigenvalue λ_k. In other words, it would consist in considering λ_k as a random function of x_k, for instance $\lambda_k \sim \mathcal{N}(\mu(x_k), \sigma_\lambda^2)$, with $\mu : \mathbb{S}^{n-1} \to \mathbb{R}$. This relation gives a generative model that defines a correlation between each eigenvector and their respective eigenvalue, leaving the eigenvectors independent – apart from the orthogonality constraint. This second formulation may be more interesting from a statistical modeling point of view, as it opens a large choice in the parameterization of the function μ.

Extensions of the spectral model. The model introduced in Chapter 5 is very flexible, in the sense that it can be easily modified to account for many types of network matrices. Its hierarchical structure also opens the possibility to model the relationship between the network and cofactors at the population level. More specifically, the latent variables X and λ could be used as regression variables to predict factors like age, sex, or more interestingly cognitive scores. Node embedding and dictionary methods have proven efficient to perform such tasks, so that the application of our model to this problem would be an interesting question. From a statistical modeling perspective, the prediction of a cofactor c_k could be achieved by specifying its relation to X and λ, for example as

$$c_k \sim \mathcal{N}(\bar{c}_k(X, \lambda), \sigma_c^2)\,,$$

with $\bar{c}_k$, e.g., a linear function. Learning the $\bar{c}_k$'s jointly with the other model parameters could also change the learned representation of X and λ by accounting for more observations.

Alternatively, cofactors could be considered as the primary source of variability, by defining the distribution of X and λ conditionally to them. This approach remains relatively unexplored in the literature on brain network analysis, which rather focuses on a regression-oriented perspective as in the previous paragraph. However, it has been widely studied in other fields of statistics, in particular mixed-effect models. Extending the hierarchical model structure of Chapter 5, the cofactors $C_i = (c_{i1}, \ldots, c_{iK})$ of individual i could be included with a formulation of the type

$$\begin{cases} C_i \sim \mathcal{N}(\bar{C}, \Sigma_c) \\ (X_i \mid C_i) \sim \mathrm{vMF}(F(C_i)) \\ (\lambda_i \mid C_i) \sim \mathcal{N}(\mu(C_i), \sigma_\lambda^2 I_p)\,. \end{cases}$$

In particular, such models could be used on data sets of brain networks to model aging. In this context, the main cofactor of interest is the age t, and the distribution of the eigenvectors and eigenvalues could thus be defined as a function of t. As an example, $\mu(t)$ could be chosen as a linear function, and $F(t)$ could be expressed as $F(t) = M(t)\mathrm{Diag}(s)$, with

$$M(t) = C_{M(t_0)}((t - t_0)V) \in \mathcal{V}_{np}$$

defined with the Cayley transform and $s \in \mathbb{R}_+^p$. In other words, the parameter V would parameterize the average evolution direction of X, starting from $M(t_0)$ at time t_0. The work of Chikuse [2006] on state-space models for Stiefel and Grassmann manifolds could be relied on to devise tailored models.

Finally, the hierarchical model of Chapter 5 could be used in the longitudinal framework of Schiratti et al. [2015] to analyze populations of brain networks measured at different time points. The framework could be adapted to network matrices by describing the time evolution of the latent variables $X_i(t)$ and $\lambda_i(t)$ of each individual. The curve $\lambda_i(t)$ could be modeled as in the classical case of Euclidean data; the evolution of $X_i(t)$ on the Stiefel manifold could be modeled as a deviation from an average trajectory as in the previous paragraph, by using the Cayley transform and the vector transport it induces between the tangent spaces [Li et al., 2020b, Zhu and Sato, 2021]. Such longitudinal modeling for brain connectivity networks could provide interpretable descriptions of aging and the progression of neurodegenerative diseases.

Handling multimodality. This thesis leaves open extensions to multi-view networks, i.e., networks where each edge has several weights measuring different types of interactions. The most important example related to our topic is the multimodal brain connectivity: the joint modeling of structural and functional connectivity is an interesting problem, which low-rank methods have already helped understanding [D'Souza et al., 2021a]. Reframing the multi-view approach in the framework of this thesis could provide an interpretable description of the relations between both aspects of brain connectivity. In terms of modeling, this goal could be achieved by considering a coupling between the latent variables of both types of connectivities. However, studies on multi-view networks require specific data sets where the same brain regions are used for functional and structural connectivity, which is not often the case and thus requires careful project planning.

Treatment impact for longitudinal models. As noted in the discussion of Chapter 7, the impact of the treatment is a crucial element of disease progression modeling. This impact has a great variability, depending on the progression stage of the disease and individual patient characteristics. Rather than considering the treatment impact – which is expected to induce a break in the patient trajectory – as a new stage of the disease progression, our results suggest that it would be more interesting to rethink the longitudinal modeling approach to account for the treatment as an exogenous cofactor rather than an evolution intrinsic to the disease progression. The framework of Schiratti et al. [2015] and Chevallier et al. [2021] could be adapted, in the case of Euclidean valued data, by introducing additional non-linear fixed and random effects related to the treatment impact. The shape of these effects, determined at both the population and individual levels, could provide a better understanding of the treatment impact, and its relation to other cofactors.

Bibliography

S. Adhikary, S. Srinivasan, and B. Boots. Learning Quantum Graphical Models using Constrained Gradient Descent on the Stiefel Manifold. *arXiv:1903.03730 [quant-ph, stat]*, Mar. 2019. URL http://arxiv.org/abs/1903.03730. arXiv: 1903.03730.

C. Aicher, A. Z. Jacobs, and A. Clauset. Learning Latent Block Structure in Weighted Networks. *Journal of Complex Networks*, 3(2):221–248, June 2015. ISSN 2051-1310, 2051-1329. doi: 10.1093/comnet/cnu026. URL http://arxiv.org/abs/1404.0431. arXiv: 1404.0431.

H. Akaike. Information Theory and an Extension of the Maximum Likelihood Principle. In E. Parzen, K. Tanabe, and G. Kitagawa, editors, *Selected Papers of Hirotugu Akaike*, Springer Series in Statistics, pages 199–213. Springer, New York, NY, 1998. ISBN 978-1-4612-1694-0. doi: 10.1007/978-1-4612-1694-0_15. URL https://doi.org/10.1007/978-1-4612-1694-0_15.

R. Albert and A.-L. Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47–97, Jan. 2002. doi: 10.1103/RevModPhys.74.47. URL https://link.aps.org/doi/10.1103/RevModPhys.74.47. Publisher: American Physical Society.

M. Ali and J. Gao. Classification of matrix-variate Fisher–Bingham distribution via Maximum Likelihood Estimation using manifold valued data. *Neurocomputing*, 295:72–85, June 2018. ISSN 0925-2312. doi: 10.1016/j.neucom.2018.01.048. URL http://www.sciencedirect.com/science/article/pii/S0925231218300730.

E. Aliverti and D. Durante. Spatial modeling of brain connectivity data via latent distance models with nodes clustering. *Stat. Anal. Data Min.*, 2019. doi: 10.1002/SAM.11412.

S. Allassonnière and L. Younes. A stochastic algorithm for probabilistic independent component analysis. *Annals of Applied Statistics*, 6(1):125–160, Mar. 2012. ISSN 1932-6157, 1941-7330. doi: 10.1214/11-AOAS499. URL https://projecteuclid.org/euclid.aoas/1331043391. Publisher: Institute of Mathematical Statistics.

S. Allassonnière, Y. Amit, and A. Trouvé. Toward a Coherent Statistical Framework for Dense Deformable Template Estimation. *Journal of the Royal Statistical Society: Series B (Methodological)*, 69(1):3–29, Jan. 2007. doi: https://doi.org/10.1111/j.1467-9868.2007.00574.x. tex.ids= allassonniereCoherentStatisticalFramework2007.

S. Allassonnière, E. Kuhn, and A. Trouvé. Construction of Bayesian deformable models via a stochastic approximation algorithm: A convergence study. *Bernoulli*, 16(3):641–678, Aug. 2010. ISSN 1350-7265. doi: 10.3150/09-BEJ229. URL https://projecteuclid.org/journals/bernoulli/volume-16/issue-3/Construction-of-Bayesian-deformable-models-via-a-stochastic-approximation-algorithm/10.3150/09-BEJ229.full. Publisher: Bernoulli Society for Mathematical Statistics and Probability.

E. S. Allman, C. Matias, and J. A. Rhodes. Identifiability of parameters in latent structure models with many observed variables. *The Annals of Statistics*, 37(6A):3099–3132, Dec. 2009. ISSN 0090-5364. doi: 10.1214/09-AOS689. URL http://arxiv.org/abs/0809.5032. arXiv: 0809.5032.

N. Alon and A. Naor. Approximating the cut-norm via Grothendieck's inequality. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, STOC '04, pages 72–80, Chicago, IL, USA, June 2004. Association for Computing Machinery. ISBN 978-1-58113-852-8. doi: 10.1145/1007352.1007371. URL https://doi.org/10.1145/1007352.1007371.

T. W. Anderson and Y. Amemiya. The Asymptotic Normal Distribution of Estimators in Factor Analysis under General Conditions. *The Annals of Statistics*, 16(2):759–771, June 1988. ISSN 0090-5364, 2168-8966. doi: 10.1214/aos/1176350834. URL https://projecteuclid.org/journals/annals-of-statistics/volume-16/issue-2/The-Asymptotic-Normal-Distribution-of-Estimators-in-Factor-Analysis-under/10.1214/aos/1176350834.full. Publisher: Institute of Mathematical Statistics.

O. Ariyo, E. Lesaffre, G. Verbeke, and A. Quintero. Model selection for Bayesian linear mixed models with longitudinal data: Sensitivity to the choice of priors. *Communications in Statistics - Simulation and Computation*, 51(4):1591–1615, Apr. 2022. ISSN 0361-0918. doi: 10.1080/03610918.2019.1676439. URL https://doi.org/10.1080/03610918.2019.1676439. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/03610918.2019.1676439.

S. Atasoy, I. Donnelly, and J. Pearson. Human brain networks function in connectome-specific harmonic waves. *Nature Communications*, 7, Jan. 2016. ISSN 2041-1723. doi: 10.1038/ncomms10340. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4735826/.

R. Azari, L. Li, and C.-L. Tsai. Longitudinal data model selection. *Computational Statistics & Data Analysis*, 50(11):3053–3066, July 2006. ISSN 0167-9473. doi: 10.1016/j.csda.2005.05.009. URL https://www.sciencedirect.com/science/article/pii/S0167947305001283.

J. Bai and P. Perron. Computation and analysis of multiple structural change models. *Journal of Applied Econometrics*, 18(1):1–22, 2003. ISSN 1099-1255. doi: 10.1002/jae.659. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/jae.659. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jae.659.

A. Banka and I. Rekik. Adversarial Connectome Embedding for Mild Cognitive Impairment Identification Using Cortical Morphological Networks. In M. D. Schirmer, A. Venkataraman, I. Rekik, M. Kim, and A. W. Chung, editors, *Connectomics in NeuroImaging*, Lecture Notes in Computer Science, pages 74–82, Cham, 2019. Springer International Publishing. ISBN 978-3-030-32391-2. doi: 10.1007/978-3-030-32391-2_8.

D. Banks and K. Carley. Metric inference for social networks. *Journal of Classification*, 11(1):121–149, Mar. 1994. ISSN 1432-1343. doi: 10.1007/BF01201026. URL https://doi.org/10.1007/BF01201026.

O. E. Barndorff-Nielsen. Identifiability of Mixtures of Exponential Families. *Journal of Mathematical Analysis and Applications*, 12:115–121, 1965.

O. E. Barndorff-Nielsen. *Information and exponential families: in statistical theory*. Wiley series in probability and mathematical statistics. Wiley, Chichester ; New York, 1978. ISBN 978-0-471-99545-6.

P. J. Bickel, Y. Ritov, and T. Rydén. Asymptotic normality of the maximum-likelihood estimator for general hidden Markov models. *The Annals of Statistics*, 26(4):1614–1635, Aug. 1998. ISSN 0090-5364. doi: 10.1214/aos/1024691255. URL https://projecteuclid.org/journals/annals-of-statistics/volume-26/issue-4/Asymptotic-normality-of-the-maximum-likelihood-estimator-for-general-hidden/10.1214/aos/1024691255.full.

E. D. Bigler. *Neuroimaging I: Basic Science*. Springer Science & Business Media, June 2013. ISBN 978-1-4899-1701-0. Google-Books-ID: OBL3BwAAQBAJ.

S. Bonhomme and J.-M. Robin. Consistent noisy independent component analysis. *Journal of Econometrics*, 149(1):12–25, Apr. 2009. ISSN 0304-4076. doi: 10.1016/j.jeconom.2008.12.019. URL https://www.sciencedirect.com/science/article/pii/S030440760900030X.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2009.

S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, July 2011. ISSN 1935-8237, 1935-8245. doi: 10.1561/2200000016. URL https://www.nowpublishers.com/article/Details/MAL-016. Publisher: Now Publishers, Inc.

E. Bullmore and O. Sporns. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience*, 10(3):186–198, Mar. 2009. ISSN 1471-0048. doi: 10.1038/nrn2575. URL https://www.nature.com/articles/nrn2575.

F. Bunea, A. Tsybakov, and M. Wegkamp. Sparsity oracle inequalities for the Lasso. *Electronic Journal of Statistics*, 1(none):169–194, Jan. 2007. ISSN 1935-7524, 1935-7524. doi: 10.1214/07-EJS008. URL https://projecteuclid.org/journals/electronic-journal-of-statistics/volume-1/issue-none/Sparsity-oracle-inequalities-for-the-Lasso/10.1214/07-EJS008.full. Publisher: Institute of Mathematical Statistics and Bernoulli Society.

K. P. Burnham and D. R. Anderson. *Model Selection and Inference*. Springer New York, New York, NY, 1998. ISBN 978-1-4757-2919-1 978-1-4757-2917-7. doi: 10.1007/978-1-4757-2917-7. URL http://link.springer.com/10.1007/978-1-4757-2917-7.

R. W. Butler. *Saddlepoint Approximations With Applications*. Cambridge University Press, 2007.

A. Cacciola, A. Muscoloni, V. Narula, A. Calamuneri, S. Nigro, E. A. Mayer, J. S. Labus, G. Anastasi, A. Quattrone, A. Quartarone, D. Milardi, and C. V. Cannistraci. Coalescent embedding in the hyperbolic space unsupervisedly discloses the hidden geometry of the brain. *arXiv:1705.04192 [cond-mat, q-bio]*, May 2017. URL http://arxiv.org/abs/1705.04192. arXiv: 1705.04192.

D. Cai, N. Ackerman, and C. Freer. An iterative step-function estimator for graphons. *arXiv:1412.2129 [math, stat]*, May 2015. URL http://arxiv.org/abs/1412.2129. arXiv: 1412.2129.

A. Calissano, A. Feragen, and S. Vantini. Populations of Unlabeled Networks: Graph Space Geometry and Geodesic Principal Components. Technical Report 14/2020, MOX, Dipartimento di Matematica, Politecnico di Milano, 2020.

A. Calissano, A. Feragen, and S. Vantini. Graph-valued regression: Prediction of unlabelled networks in a non-Euclidean graph space. *Journal of Multivariate Analysis*, 190:104950, July 2022. ISSN 0047259X. doi: 10.1016/j.jmva.2022.104950. URL https://linkinghub.elsevier.com/retrieve/pii/S0047259X22000021.

E. J. Candes and Y. Plan. Matrix Completion With Noise. *Proceedings of the IEEE*, 98(6):925–936, June 2010. ISSN 1558-2256. doi: 10.1109/JPROC.2009.2035722.

E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM*, 58(3):11:1–11:37, June 2011. ISSN 0004-5411. doi: 10.1145/1970392.1970395. URL https://doi.org/10.1145/1970392.1970395.

G. Celeux, S. Frühwirth-Schnatter, and C. P. Robert. Model Selection for Mixture Models – Perspectives and Strategies. In S. Frühwirth-Schnatter, G. Celeux, and C. P. Robert, editors, *Handbook of Mixture Analysis*, pages 117–154. Chapman and Hall/CRC, Boca Raton, Florida : CRC Press, [2019], 1 edition, Jan. 2019. ISBN 978-0-429-05591-1. doi: 10.1201/9780429055911-7. URL https://www.taylorfrancis.com/books/9780429508240/chapters/10.1201/9780429055911-7.

I. Chami, Z. Ying, C. Ré, and J. Leskovec. Hyperbolic Graph Convolutional Neural Networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d. Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*,

pages 4868–4879. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/8733-hyperbolic-graph-convolutional-neural-networks.pdf.

S. Chandna and P.-A. Maugis. Nonparametric regression for multiple heterogeneous networks. *arXiv:2001.04938 [stat]*, Jan. 2020. URL http://arxiv.org/abs/2001.04938. arXiv: 2001.04938.

J. Chen and Z. Chen. Extended Bayesian Information Criteria for Model Selection with Large Model Spaces. *Biometrika*, 95(3):759–771, 2008. ISSN 0006-3444. URL https://www.jstor.org/stable/20441500. Publisher: [Oxford University Press, Biometrika Trust].

J. Chen, G. Han, H. Cai, J. Ma, M. Kim, P. Laurienti, and G. Wu. Estimating Common Harmonic Waves of Brain Networks on Stiefel Manifold. In A. L. Martel, P. Abolmaesumi, D. Stoyanov, D. Mateus, M. A. Zuluaga, S. K. Zhou, D. Racoceanu, and L. Joskowicz, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, Lecture Notes in Computer Science, pages 367–376, Cham, Oct. 2020. Springer International Publishing. ISBN 978-3-030-59728-3. doi: 10.1007/978-3-030-59728-3_36.

J. Chevallier. *Modèles statistiques et algorithmes stochastiques pour l'analyse de données longitudinales à dynamiques multiples et à valeurs sur des variétés riemanniennes*. PhD thesis, École polytechnique, 2019.

J. Chevallier, V. Debavelaere, and S. Allassonnière. A coherent framework for learning spatiotemporal piecewise-geodesic trajectories from longitudinal manifold-valued data. *SIAM Journal on Imaging Sciences*, 14(1):349–388, 2021. doi: https://doi.org/10.1137/20M1328026. URL https://hal.archives-ouvertes.fr/hal-01646298. Publisher: Society for Industrial and Applied Mathematics.

Y. Chikuse. Distributions of orientations on Stiefel manifolds. *Journal of Multivariate Analysis*, 33(2):247–264, May 1990. ISSN 0047-259X. doi: 10.1016/0047-259X(90)90049-N. URL http://www.sciencedirect.com/science/article/pii/0047259X9090049N.

Y. Chikuse. Concentrated matrix Langevin distributions. *Journal of Multivariate Analysis*, 85 (2):375–394, May 2003a. ISSN 0047-259X. doi: 10.1016/S0047-259X(02)00065-9. URL http://www.sciencedirect.com/science/article/pii/S0047259X02000659.

Y. Chikuse. The Inference on the Parameters of the Matrix Langevin Distributions. In Y. Chikuse, editor, *Statistics on Special Manifolds*, Lecture Notes in Statistics, pages 109–132. Springer, New York, NY, 2003b. ISBN 978-0-387-21540-2. doi: 10.1007/978-0-387-21540-2_5. URL https://doi.org/10.1007/978-0-387-21540-2_5.

Y. Chikuse. *Statistics on Special Manifolds*. Lecture Notes in Statistics. Springer-Verlag, New York, 2003c. ISBN 978-0-387-00160-9. doi: 10.1007/978-0-387-21540-2. URL https://www.springer.com/gp/book/9780387001609.

Y. Chikuse. State space models on special manifolds. *Journal of Multivariate Analysis*, 97(6): 1284–1294, July 2006. ISSN 0047-259X. doi: 10.1016/j.jmva.2006.03.002. URL https://www.sciencedirect.com/science/article/pii/S0047259X06000364.

M. K. Chung, J. L. Hanson, J. Ye, R. J. Davidson, and S. D. Pollak. Persistent Homology in Sparse Regression and Its Application to Brain Morphometry. *IEEE transactions on medical imaging*, 34(9):1928–1939, Sept. 2015. ISSN 0278-0062. doi: 10.1109/TMI.2015.2416271. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4629505/.

M. K. Chung, H. Lee, V. Solo, R. J. Davidson, and S. D. Pollak. Topological Distances Between Brain Networks. In G. Wu, P. Laurienti, L. Bonilha, and B. C. Munsell, editors, *Connectomics in NeuroImaging*, volume 10511, pages 161–170. Springer International Publishing, Cham, 2017a. ISBN 978-3-319-67158-1 978-3-319-67159-8. doi: 10.1007/978-3-319-67159-8_19. URL http://link.springer.com/10.1007/978-3-319-67159-8_19. Series Title: Lecture Notes in Computer Science.

M. K. Chung, V. Villalta-Gil, H. Lee, P. J. Rathouz, B. B. Lahey, and D. H. Zald. Exact Topological Inference for Paired Brain Networks via Persistent Homology. In M. Niethammer, M. Styner, S. Aylward, H. Zhu, I. Oguz, P.-T. Yap, and D. Shen, editors, *Information Processing in Medical Imaging: 25th International Conference, Proceedings*, volume 10265 of *Lecture Notes in Computer Science*. Springer International Publishing, June 2017b. ISBN 978-3-319-59049-3 978-3-319-59050-9. doi: 10.1007/978-3-319-59050-9. URL http://link.springer.com/10.1007/978-3-319-59050-9.

M. K. Chung, S.-G. Huang, A. Gritsenko, L. Shen, and H. Lee. Statistical Inference on the Number of Cycles in Brain Networks. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pages 113–116, Venice, Italy, Apr. 2019. IEEE. ISBN 978-1-5386-3641-1. doi: 10.1109/ISBI.2019.8759222. URL https://ieeexplore.ieee.org/document/8759222/.

G. Claeskens and N. L. Hjort. The Focused Information Criterion. *Journal of the American Statistical Association*, 98(464):900–916, Dec. 2003. ISSN 0162-1459. doi: 10.1198/016214503000000819. URL https://doi.org/10.1198/016214503000000819. Publisher: Taylor & Francis _eprint: https://doi.org/10.1198/016214503000000819.

R. Couronné. *Progression models for Parkinson's Disease*. PhD thesis, Sorbonne Université, 2021. URL https://tel.archives-ouvertes.fr/tel-03491211/file/These_RaphaelCouronne.pdf.

E. Crespo Marques, N. Maciel, L. Naviner, H. Cai, and J. Yang. A Review of Sparse Recovery Algorithms. *IEEE Access*, 7:1300–1322, 2019. ISSN 2169-3536. doi: 10.1109/ACCESS.2018.2886471. Conference Name: IEEE Access.

J. S. Damoiseaux. Effects of aging on functional and structural brain connectivity. *NeuroImage*, 160:32–40, Oct. 2017. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2017.01.077. URL http://www.sciencedirect.com/science/article/pii/S1053811917301015.

J. S. Damoiseaux, S. A. R. B. Rombouts, F. Barkhof, P. Scheltens, C. J. Stam, S. M. Smith, and C. F. Beckmann. Consistent resting-state networks across healthy subjects. *Proceedings of the National Academy of Sciences of the United States of America*, 103(37):13848–13853, Sept. 2006. ISSN 0027-8424. doi: 10.1073/pnas.0601417103. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1564249/.

V. Debavelaere and S. Allassonnière. On the curved exponential family in the Stochastic Approximation Expectation Maximization Algorithm. *ESAIM: Probability and Statistics*, 25:408–432, 2021. ISSN 1262-3318. doi: 10.1051/ps/2021015. URL https://www.esaim-ps.org/articles/ps/abs/2021/01/ps210011/ps210011.html. Publisher: EDP Sciences.

V. Debavelaere, S. Durrleman, S. Allassonnière, and for the Alzheimer's Disease Neuroimaging Initiative. Learning the Clustering of Longitudinal Shape Data Sets into a Mixture of Independent or Branching Trajectories. *International Journal of Computer Vision*, 128(12):2794–2809, Dec. 2020. ISSN 0920-5691, 1573-1405. doi: 10.1007/s11263-020-01337-8. URL http://link.springer.com/10.1007/s11263-020-01337-8.

M. Delattre, M. Lavielle, and M.-A. Poursat. A note on BIC in mixed-effects models. *Electronic Journal of Statistics*, 8(1):456–475, Jan. 2014. ISSN 1935-7524, 1935-7524. doi: 10.1214/14-EJS890. URL https://projecteuclid.org/journals/electronic-journal-of-statistics/volume-8/issue-1/A-note-on-BIC-in-mixed-effects-models/10.1214/14-EJS890.full. Publisher: Institute of Mathematical Statistics and Bernoulli Society.

B. Delyon, M. Lavielle, and E. Moulines. Convergence of a stochastic approximation version of the EM algorithm. *The Annals of Statistics*, 27(1):94–128, Mar. 1999. ISSN 0090-5364, 2168-8966. doi: 10.1214/aos/1018031103. URL https://projecteuclid.org/euclid.aos/1018031103.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data Via the EM Algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22,

1977. ISSN 2517-6161. doi: https://doi.org/10.1111/j.2517-6161.1977.tb01600.x. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.2517-6161.1977.tb01600.x. _eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.2517-6161.1977.tb01600.x.

R. Douc. Non singularity of the asymptotic Fisher information matrix in hidden Markov models. *arXiv:math/0511631*, Nov. 2005. URL http://arxiv.org/abs/math/0511631. arXiv: math/0511631.

R. Douc, E. Moulines, J. Olsson, and R. van Handel. Consistency of the Maximum Likelihood Estimator for General Hidden Markov Models. *The Annals of Statistics*, 39(1):474–513, 2011. ISSN 0090-5364. URL https://www.jstor.org/stable/29783645. Publisher: Institute of Mathematical Statistics.

R. Douc, F. Roueff, and T. Sim. Necessary and sufficient conditions for the identifiability of observation-driven models. *Journal of Time Series Analysis*, 42(2):140–160, 2021. ISSN 1467-9892. doi: 10.1111/jtsa.12559. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/jtsa.12559. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/jtsa.12559.

N. S. D'Souza, M. B. Nebel, N. Wymbs, S. Mostofsky, and A. Venkataraman. A Generative-Discriminative Basis Learning Framework to Predict Clinical Severity from Resting State Functional MRI Data. In A. F. Frangi, J. A. Schnabel, C. Davatzikos, C. Alberola-López, and G. Fichtinger, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, volume 11072, pages 163–171. Springer International Publishing, Cham, 2018. ISBN 978-3-030-00930-4 978-3-030-00931-1. doi: 10.1007/978-3-030-00931-1_19. URL http://link.springer.com/10.1007/978-3-030-00931-1_19. Series Title: Lecture Notes in Computer Science.

N. S. D'Souza, M. B. Nebel, N. Wymbs, S. Mostofsky, and A. Venkataraman. A Coupled Manifold Optimization Framework to Jointly Model the Functional Connectomics and Behavioral Data Spaces. In A. C. S. Chung, J. C. Gee, P. A. Yushkevich, and S. Bao, editors, *Information Processing in Medical Imaging*, volume 11492, pages 605–616. Springer International Publishing, Cham, 2019a. ISBN 978-3-030-20350-4 978-3-030-20351-1. doi: 10.1007/978-3-030-20351-1_47. URL http://link.springer.com/10.1007/978-3-030-20351-1_47. Series Title: Lecture Notes in Computer Science.

N. S. D'Souza, M. B. Nebel, N. Wymbs, S. Mostofsky, and A. Venkataraman. Integrating Neural Networks and Dictionary Learning for Multidimensional Clinical Characterizations from Functional Connectomics Data. In D. Shen, T. Liu, T. M. Peters, L. H. Staib, C. Essert, S. Zhou, P.-T. Yap, and A. Khan, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, volume 11766, pages 709–717. Springer International Publishing, Cham, 2019b. ISBN 978-3-030-32247-2 978-3-030-32248-9. doi: 10.1007/978-3-030-32248-9_79. URL http://link.springer.com/10.1007/978-3-030-32248-9_79. Series Title: Lecture Notes in Computer Science.

L. L. Duan, G. Michailidis, and M. Ding. Spiked Laplacian Graphs: Bayesian Community Detection in Heterogeneous Networks. *arXiv:1910.02471 [stat]*, Mar. 2020. URL http://arxiv.org/abs/1910.02471. arXiv: 1910.02471.

D. Durante, D. B. Dunson, and J. T. Vogelstein. Nonparametric Bayes Modeling of Populations of Networks. *Journal of the American Statistical Association*, 112(520):1516–1530, Oct. 2017. ISSN 0162-1459. doi: 10.1080/01621459.2016.1219260. URL https://amstat.tandfonline.com/doi/abs/10.1080/01621459.2016.1219260. Publisher: Taylor & Francis.

D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams. Convolutional Networks on Graphs for Learning Molecular Fingerprints. *Advances in Neural Information Processing Systems*, 28:2224–2232, 2015. URL https://papers.nips.cc/paper/2015/hash/f9be311e65d81a9ad8150a60844bb94c-Abstract.html.

N. S. D'Souza, M. B. Nebel, D. Crocetti, J. Robinson, S. Mostofsky, and A. Venkataraman. A Matrix Autoencoder Framework to Align the Functional and Structural Connectivity Manifolds as Guided by Behavioral Phenotypes. In M. de Bruijne, P. C. Cattin, S. Cotin, N. Padoy, S. Speidel, Y. Zheng, and C. Essert, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021*, Lecture Notes in Computer Science, pages 625–636, Cham, 2021a. Springer International Publishing. ISBN 978-3-030-87234-2. doi: 10.1007/978-3-030-87234-2_59.

N. S. D'Souza, M. B. Nebel, D. Crocetti, J. Robinson, N. Wymbs, S. H. Mostofsky, and A. Venkataraman. Deep sr-DDL: Deep structurally regularized dynamic dictionary learning to integrate multimodal and dynamic functional connectomics data for multidimensional clinical characterizations. *NeuroImage*, 241:118388, Nov. 2021b. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2021.118388. URL https://www.sciencedirect.com/science/article/pii/S1053811921006649.

A. Edelman, T. A. Arias, and S. T. Smith. The Geometry of Algorithms with Orthogonality Constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2):303–353, Jan. 1998. ISSN 0895-4798, 1095-7162. doi: 10.1137/S0895479895290954. URL http://epubs.siam.org/doi/10.1137/S0895479895290954.

P. Erdős and A. Rényi. On random graphs. *Publicationes Mathematicae*, 6:290–297, 1959.

M. W. Eysenck. *Cognitive psychology : a student's handbook*. New York, NY : Psychology Press, 2010. ISBN 978-1-84169-540-2 978-1-84169-539-6. URL http://archive.org/details/cognitivepsychol0000eyse_t0m9.

K. Fan. On a Theorem of Weyl Concerning Eigenvalues of Linear Transformations I. *Proceedings of the National Academy of Sciences*, 35(11):652–655, Nov. 1949. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.35.11.652. URL https://www.pnas.org/content/35/11/652. Publisher: National Academy of Sciences Section: Mathematics.

G. Fang and P. Li. On Estimation in Latent Variable Models. In *Proceedings of the 38th International Conference on Machine Learning*, pages 3100–3110. PMLR, July 2021. URL https://proceedings.mlr.press/v139/fang21a.html. ISSN: 2640-3498.

S. Fiori, T. Kaneko, and T. Tanaka. Learning on the compact Stiefel manifold by a cayley-transform-based pseudo-retraction map. In *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, June 2012. doi: 10.1109/IJCNN.2012.6252841. ISSN: 2161-4407.

A. Fornito, A. Zalesky, and E. Bullmore. *Fundamentals of Brain Network Analysis*. Academic press, 2016. ISBN 978-0-12-407908-3.

P. J. Forrester. *Log-Gases and Random Matrices (LMS-34)*, volume 34 of *London Mathematical Society Monographs*. Princeton University Press, July 2010. ISBN 978-1-4008-3541-6. URL https://www.degruyter.com/document/doi/10.1515/9781400835416/html. Publication Title: Log-Gases and Random Matrices (LMS-34).

C. Fraikin, K. Hüper, and P. V. Dooren. Optimization over the Stiefel manifold. In *PAMM: Proceedings in Applied Mathematics and Mechanics*, volume 7, pages 1062205–1062206. Wiley Online Library, 2007. Issue: 1.

C. Gao, Y. Lu, and H. H. Zhou. Rate-optimal graphon estimation. *The Annals of Statistics*, 43(6):2624–2652, Dec. 2015. ISSN 0090-5364. doi: 10.1214/15-AOS1354. URL http://arxiv.org/abs/1410.5837. arXiv: 1410.5837.

S. Gao, G. Mishne, and D. Scheinost. Poincaré Embedding Reveals Edge-Based Functional Networks of the Brain. In A. L. Martel, P. Abolmaesumi, D. Stoyanov, D. Mateus, M. A. Zuluaga, S. K. Zhou, D. Racoceanu, and L. Joskowicz, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, volume 12267, pages 448–457, Cham, 2020. Springer International Publishing. ISBN 978-3-030-59727-6 978-3-030-59728-3. doi: 10.1007/

978-3-030-59728-3_44. URL http://link.springer.com/10.1007/978-3-030-59728-3_44. Series Title: Lecture Notes in Computer Science.

S. Ghosh, N. Das, T. Gonçalves, P. Quaresma, and M. Kundu. The journey of graph kernels through two decades. *Computer Science Review*, 27:88–111, Feb. 2018. ISSN 1574-0137. doi: 10.1016/j.cosrev.2017.11.002. URL http://www.sciencedirect.com/science/article/pii/S1574013717301429.

Q. F. Gronau, A. Sarafoglou, D. Matzke, A. Ly, U. Boehm, M. Marsman, D. S. Leslie, J. J. Forster, E.-J. Wagenmakers, and H. Steingroever. A tutorial on bridge sampling. *Journal of Mathematical Psychology*, 81:80–97, Dec. 2017. ISSN 0022-2496. doi: 10.1016/j.jmp.2017.09.005. URL https://www.sciencedirect.com/science/article/pii/S0022249617300640.

Y. Gu and G. Xu. Identifiability of Hierarchical Latent Attribute Models. *arXiv:1906.07869 [cs, stat]*, Jan. 2021. URL http://arxiv.org/abs/1906.07869. arXiv: 1906.07869.

A. A. Hagberg, P. J. Swart, and D. A. Schult. Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conference (SciPy2008), Gäel Varoquaux, Travis Vaught, and Jarrod Millman (Eds)*, pages 11–15, Pasadena, CA USA, Aug. 2008.

D. K. Hammond, P. Vandergheynst, and R. Gribonval. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30(2):129–150, Mar. 2011. ISSN 1063-5203. doi: 10.1016/j.acha.2010.04.005. URL http://www.sciencedirect.com/science/article/pii/S1063520310000552.

S. Hanneke, W. Fu, and E. P. Xing. Discrete temporal models of social networks. *Electronic Journal of Statistics*, 4:585–605, 2010. ISSN 1935-7524. doi: 10.1214/09-EJS548. URL https://projecteuclid.org/euclid.ejs/1276694116. Publisher: The Institute of Mathematical Statistics and the Bernoulli Society.

J. K. Harris. *An introduction to exponential random graph modeling*. Number 173 in Quantitative applications in the social sciences. SAGE, Thousand Oaks, California, 2014. ISBN 978-1-4522-2080-2.

W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, Apr. 1970. ISSN 0006-3444. doi: 10.1093/biomet/57.1.97. URL https://doi.org/10.1093/biomet/57.1.97.

Y. He and A. Evans. Graph theoretical modeling of brain connectivity. *Current Opinion in Neurology*, 23(4):341–350, Aug. 2010. ISSN 1473-6551. doi: 10.1097/WCO.0b013e32833aa567.

T. Head, MechCoder, G. Louppe, I. Shcherbatyi, fcharras, Z. Vinícius, cmmalone, C. Schröder, nel215, N. Campos, T. Young, S. Cereda, T. Fan, rene rex, K. K. Shi, J. Schwabedal, carlosdanielcsantos, Hvass-Labs, M. Pak, SoManyUsernamesTaken, F. Callaway, L. Estève, L. Besson, M. Cherti, K. Pfannschmidt, F. Linzberger, C. Cauet, A. Gut, A. Mueller, and A. Fabisch. scikit-optimize/scikit-optimize: v0.5.2, Mar. 2018. URL https://zenodo.org/record/1207017.

M. Henaff, J. Bruna, and Y. LeCun. Deep Convolutional Networks on Graph-Structured Data. *arXiv:1506.05163 [cs]*, June 2015. URL http://arxiv.org/abs/1506.05163. arXiv: 1506.05163.

A. M. Hermundstad, D. S. Bassett, K. S. Brown, E. M. Aminoff, D. Clewett, S. Freeman, A. Frithsen, A. Johnson, C. M. Tipper, M. B. Miller, S. T. Grafton, and J. M. Carlson. Structural foundations of resting-state and task-based functional connectivity in the human brain. *Proceedings of the National Academy of Sciences of the United States of America*, 110(15):6169–6174, Apr. 2013. ISSN 1091-6490. doi: 10.1073/pnas.1219562110.

Q. Ho, A. P. Parikh, and E. P. Xing. Multiscale Community Blockmodel for Network Exploration. *Journal of the American Statistical Association*, 107(499), 2012. ISSN 0162-1459. doi: 10.1080/01621459.2012.682530. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3840468/.

P. D. Hoff. Model Averaging and Dimension Selection for the Singular Value Decomposition. *Journal of the American Statistical Association*, 102(478):674–685, 2007a. ISSN 0162-1459. URL https://www.jstor.org/stable/27639896. Publisher: [American Statistical Association, Taylor & Francis, Ltd.].

P. D. Hoff. Modeling homophily and stochastic equivalence in symmetric relational data. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS'07, pages 657–664, Red Hook, NY, USA, Dec. 2007b. Curran Associates Inc. ISBN 978-1-60560-352-0.

P. D. Hoff. A hierarchical eigenmodel for pooled covariance estimation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(5):971–992, Nov. 2009a. ISSN 13697412, 14679868. doi: 10.1111/j.1467-9868.2009.00716.x. URL http://doi.wiley.com/10.1111/j.1467-9868.2009.00716.x.

P. D. Hoff. Simulation of the Matrix Bingham—von Mises—Fisher Distribution, With Applications to Multivariate and Relational Data. *Journal of Computational and Graphical Statistics*, 18(2):438–456, 2009b. ISSN 1061-8600. doi: 10.1198/jcgs.2009.07177. URL https://www.jstor.org/stable/25651254. Publisher: [American Statistical Association, Taylor & Francis, Ltd., Institute of Mathematical Statistics, Interface Foundation of America].

P. D. Hoff and M. D. Ward. Modeling Dependencies in International Relations Networks. *Political Analysis*, 12(2):160–175, 2004. ISSN 1047-1987, 1476-4989. doi: 10.1093/pan/mph012. URL https://www.cambridge.org/core/product/identifier/S1047198700009773/type/journal_article.

M. D. Hoffman. Learning Deep Latent Gaussian Models with Markov Chain Monte Carlo. In *International Conference on Machine Learning*, pages 1510–1519, July 2017. URL http://proceedings.mlr.press/v70/hoffman17a.html.

H. Holzmann, A. Munk, and B. Stratmann. Identifiability of Finite Mixtures - with Applications to Circular Distributions. *Sankhyā: The Indian Journal of Statistics (2003-2007)*, 66(3):440–449, 2004. ISSN 0972-7671. URL https://www.jstor.org/stable/25053372. Publisher: Springer.

A. Horn, D. Ostwald, M. Reisert, and F. Blankenburg. The structural–functional connectome and the default mode network of the human brain. *NeuroImage*, 102:142–151, Nov. 2014. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2013.09.069. URL https://www.sciencedirect.com/science/article/pii/S1053811913010057.

Z. Hu, F. Nie, L. Tian, R. Wang, and X. Li. A Comprehensive Survey for Low Rank Regularization. *arXiv:1808.04521 [cs]*, Sept. 2018. URL http://arxiv.org/abs/1808.04521. arXiv: 1808.04521.

G.-X. Huang, F. Yin, and K. Guo. An iterative method for the skew-symmetric solution and the optimal approximate solution of the matrix equation AXB=C. *Journal of Computational and Applied Mathematics*, 212(2):231–244, Mar. 2008. ISSN 0377-0427. doi: 10.1016/j.cam.2006.12.005. URL https://www.sciencedirect.com/science/article/pii/S0377042706007266.

J. G. Ibrahim and G. Molenberghs. Missing data methods in longitudinal studies: a review. *TEST*, 18(1):1–43, May 2009. ISSN 1863-8260. doi: 10.1007/s11749-009-0138-x. URL https://doi.org/10.1007/s11749-009-0138-x.

S. Janson. Graphons, cut norm and distance, couplings and rearrangements. *NYJM Monographs*, 4:76, 2013a.

S. Janson. Graphons, cut norm and distance, couplings and rearrangements, vol. 4 of New York Journal of Mathematics. *NYJM Monographs, State University of New York, University at Albany, Albany, NY*, 4:76, 2013b. ISSN 1076-9803. URL http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-284062.

M. Jauch, P. D. Hoff, and D. B. Dunson. Monte Carlo Simulation on the Stiefel Manifold via Polar Expansion. *Journal of Computational and Graphical Statistics*, 0 (0):1–10, Dec. 2020a. ISSN 1061-8600. doi: 10.1080/10618600.2020.1859382. URL https://doi.org/10.1080/10618600.2020.1859382. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/10618600.2020.1859382.

M. Jauch, P. D. Hoff, and D. B. Dunson. Random orthogonal matrices and the Cayley transform. *Bernoulli*, 26(2):1560–1586, May 2020b. ISSN 1350-7265. doi: 10.3150/19-BEJ1176. URL https://projecteuclid.org/journals/bernoulli/volume-26/issue-2/Random-orthogonal-matrices-and-the-Cayley-transform/10.3150/19-BEJ1176.full. Publisher: Bernoulli Society for Mathematical Statistics and Probability.

B. Jie, X. Jiang, C. Zu, and D. Zhang. The New Graph Kernels on Connectivity Networks for Identification of MCI. In I. Rish, G. Langs, L. Wehbe, G. Cecchi, K.-m. K. Chang, and B. Murphy, editors, *Machine Learning and Interpretation in Neuroimaging*, Lecture Notes in Computer Science, pages 12–20, Cham, 2016. Springer International Publishing. ISBN 978-3-319-45174-9. doi: 10.1007/978-3-319-45174-9_2.

R. H. Jones. Bayesian information criterion for longitudinal and clustered data. *Statistics in Medicine*, 30(25):3050–3056, Nov. 2011. ISSN 1097-0258. doi: 10.1002/sim.4323.

P. E. Jupp and K. V. Mardia. Maximum Likelihood Estimators for the Matrix Von Mises-Fisher and Bingham Distributions. *Annals of Statistics*, 7(3):599–606, May 1979. ISSN 0090-5364, 2168-8966. doi: 10.1214/aos/1176344681. URL https://projecteuclid.org/euclid.aos/1176344681. Publisher: Institute of Mathematical Statistics.

T. Kanada, M. Onuki, and Y. Tanaka. Low-rank Sparse Decomposition of Graph Adjacency Matrices for Extracting Clean Clusters. In *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1153–1159, Nov. 2018. doi: 10.23919/APSIPA.2018.8659769. ISSN: 2640-0103.

T. Kaneko, S. Fiori, and T. Tanaka. Empirical Arithmetic Averaging Over the Compact Stiefel Manifold. *IEEE Transactions on Signal Processing*, 61(4):883–894, Feb. 2013. ISSN 1941-0476. doi: 10.1109/TSP.2012.2226167. Conference Name: IEEE Transactions on Signal Processing.

W. Karush. Minima of functions of several variables with inequalities as side constraints. *M. Sc. Dissertation. Dept. of Mathematics, Univ. of Chicago*, 1939. URL https://ci.nii.ac.jp/naid/10027639655/.

J. T. Kent. Identifiability of Finite Mixtures for Directional Data. *The Annals of Statistics*, 11(3), Sept. 1983. ISSN 0090-5364. doi: 10.1214/aos/1176346264. URL https://projecteuclid.org/journals/annals-of-statistics/volume-11/issue-3/Identifiability-of-Finite-Mixtures-for-Directional-Data/10.1214/aos/1176346264.full.

B. S. Khan and M. A. Niazi. Network Community Detection: A Review and Visual Survey, Aug. 2017. URL http://arxiv.org/abs/1708.00977. arXiv:1708.00977 [cs] type: article.

C. G. Khatri and K. V. Mardia. The von Mises–Fisher Matrix Distribution in Orientation Statistics. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):95–106, 1977. ISSN 2517-6161. doi: https://doi.org/10.1111/j.2517-6161.1977.tb01610.x. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.2517-6161.1977.tb01610.x. _eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.2517-6161.1977.tb01610.x.

A. Khetan and M. Mj. Cheeger inequalities for graph limits. *arXiv:1807.02225 [math]*, Nov. 2018. URL http://arxiv.org/abs/1807.02225. arXiv: 1807.02225.

D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes, May 2014. URL http://arxiv.org/abs/1312.6114. Number: arXiv:1312.6114 arXiv:1312.6114 [cs, stat].

T. N. Kipf and M. Welling. Variational Graph Auto-Encoders. Barcelona, Spain, Nov. 2016. URL http://arxiv.org/abs/1611.07308. arXiv: 1611.07308.

T. N. Kipf and M. Welling. Semi-Supervised Classification with Graph Convolutional Networks. In *ICLR 2017*, Feb. 2017. URL http://arxiv.org/abs/1609.02907. arXiv: 1609.02907.

V. Kiviniemi, J.-H. Kantola, J. Jauhiainen, A. Hyvärinen, and O. Tervonen. Independent component analysis of nondeterministic fMRI signal sources. *NeuroImage*, 19(2):253–260, June 2003. ISSN 1053-8119. doi: 10.1016/S1053-8119(03)00097-1. URL http://www.sciencedirect.com/science/article/pii/S1053811903000971.

S. Kojaku and N. Masuda. Core-periphery structure requires something else in the network. *New Journal of Physics*, 20(4):043012, Apr. 2018. ISSN 1367-2630. doi: 10.1088/1367-2630/aab547. URL https://doi.org/10.1088%2F1367-2630%2Faab547. Publisher: IOP Publishing.

V. Koltchinskii, K. Lounici, and A. B. Tsybakov. Estimation of low-rank covariance function. *Stochastic Processes and their Applications*, 126(12):3952–3967, Dec. 2016. ISSN 0304-4149. doi: 10.1016/j.spa.2016.04.006. URL http://www.sciencedirect.com/science/article/pii/S030441491630028X.

Z. Kong, A. Kendre, J. Yu, H. Peng, C. Yang, L. Sun, A. Leow, and L. He. Structure-Preserving Graph Kernel for Brain Network Classification. *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*, 2022. doi: 10.1109/ISBI52829.2022.9761456.

S. Konishi and G. Kitagawa. *Information Criteria and Statistical Modeling*. Springer Series in Statistics. Springer New York, New York, NY, 2008. ISBN 978-0-387-71886-6 978-0-387-71887-3. doi: 10.1007/978-0-387-71887-3. URL http://link.springer.com/10.1007/978-0-387-71887-3.

I. Koval. *Learning Multimodal Digital Models of Disease Progression from Longitudinal Data: Methods & Algorithms for the Description, Prediction and Simulation of Alzheimer's Disease Progression*. PhD thesis, École polytechnique, 2020.

D. Krioukov, F. Papadopoulos, M. Kitsak, A. Vahdat, and M. Boguna. Hyperbolic Geometry of Complex Networks. *Physical Review E*, 82(3):036106, Sept. 2010. ISSN 1539-3755, 1550-2376. doi: 10.1103/PhysRevE.82.036106. URL http://arxiv.org/abs/1006.5169. arXiv: 1006.5169.

E. Kuhn and M. Lavielle. Coupling a stochastic approximation version of EM with an MCMC procedure. *ESAIM: Probability and Statistics*, 8:115–131, Aug. 2004. ISSN 1292-8100, 1262-3318. doi: 10.1051/ps:2004007. URL http://www.esaim-ps.org/10.1051/ps:2004007. tex.ids= kuhnCouplingStochasticApproximation2004 publisher: EDP Sciences.

E. Kuhn and M. Lavielle. Maximum likelihood estimation in nonlinear mixed effects models. *Computational Statistics & Data Analysis*, 49(4):1020–1038, June 2005. ISSN 0167-9473. doi: 10.1016/j.csda.2004.07.002. URL https://www.sciencedirect.com/science/article/pii/S0167947304002221.

A. Kume and T. Sei. On the exact maximum likelihood inference of Fisher–Bingham distributions using an adjusted holonomic gradient method. *Statistics and Computing*, 28(4):835–847, July 2018. ISSN 1573-1375. doi: 10.1007/s11222-017-9765-3. URL https://doi.org/10.1007/s11222-017-9765-3.

A. Kume, S. P. Preston, and A. T. A. Wood. Saddlepoint approximations for the normalizing constant of Fisher–Bingham distributions on products of spheres and Stiefel manifolds. *Biometrika*, 100(4):971–984, Dec. 2013. ISSN 0006-3444. doi: 10.1093/biomet/ast021. URL https://doi.org/10.1093/biomet/ast021.

J. Kunegis, M. Blattner, and C. Moser. Preferential attachment in online networks: measurement and explanations. In *Proceedings of the 5th Annual ACM Web Science Conference*, WebSci '13, pages 205–214, New York, NY, USA, May 2013. Association for Computing Machinery. ISBN

978-1-4503-1889-1. doi: 10.1145/2464464.2464514. URL https://doi.org/10.1145/2464464.2464514.

N. M. Laird and J. H. Ware. Random-Effects Models for Longitudinal Data. *Biometrics*, 38(4): 963–974, 1982. ISSN 0006-341X. doi: 10.2307/2529876. URL https://www.jstor.org/stable/2529876. Publisher: [Wiley, International Biometric Society].

S. K. Lam, A. Pitrou, and S. Seibert. Numba: a LLVM-based Python JIT compiler. In *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*, LLVM '15, pages 1–6, New York, NY, USA, Nov. 2015. Association for Computing Machinery. ISBN 978-1-4503-4005-2. doi: 10.1145/2833157.2833162. URL https://doi.org/10.1145/2833157.2833162.

P. Latouche and S. Robin. Variational Bayes model averaging for graphon functions and motif frequencies inference in W-graph models. *Statistics and Computing*, 26(6):1173–1185, Nov. 2016. ISSN 0960-3174, 1573-1375. doi: 10.1007/s11222-015-9607-0. URL http://link.springer.com/10.1007/s11222-015-9607-0.

M. Lavielle. *Mixed Effects Models for the Population Approach: Models, Tasks, Methods and Tools*. Chapman and Hall/CRC, 2014. URL https://hal.archives-ouvertes.fr/hal-01122873.

M. Lavielle and L. Aarons. What do we mean by identifiability in mixed effects models? *Journal of Pharmacokinetics and Pharmacodynamics*, 43(1):111–122, Feb. 2016. ISSN 1567-567X, 1573-8744. doi: 10.1007/s10928-015-9459-4. URL http://link.springer.com/10.1007/s10928-015-9459-4.

P. D. Lax. *Functional analysis*. Wiley, New York, 2002. ISBN 978-0-471-55604-6.

J. M. Lee. *Introduction to Smooth Manifolds*. Graduate Texts in Mathematics. Springer-Verlag, New York, 2003. ISBN 978-0-387-21752-9. doi: 10.1007/978-0-387-21752-9. URL https://www.springer.com/gp/book/9780387217529.

J. M. Lee. *Introduction to Riemannian Manifolds*. Graduate Texts in Mathematics. Springer International Publishing, 2 edition, 2018. ISBN 978-3-319-91754-2. doi: 10.1007/978-3-319-91755-9. URL https://www.springer.com/gp/book/9783319917542.

E. L. Lehmann and G. Casella. *Theory of point estimation*. Springer texts in statistics. Springer, New York, 2. ed., corrected 4. printing edition, 2003. ISBN 978-0-387-98502-2. OCLC: 250004226.

J. Li, C. Bian, D. Chen, X. Meng, H. Luo, H. Liang, and L. Shen. Persistent Feature Analysis of Multimodal Brain Networks Using Generalized Fused Lasso for EMCI Identification. In A. L. Martel, P. Abolmaesumi, D. Stoyanov, D. Mateus, M. A. Zuluaga, S. K. Zhou, D. Racoceanu, and L. Joskowicz, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, volume 12267, pages 44–52, Cham, 2020a. Springer International Publishing. ISBN 978-3-030-59727-6 978-3-030-59728-3. doi: 10.1007/978-3-030-59728-3_5. URL http://link.springer.com/10.1007/978-3-030-59728-3_5. Series Title: Lecture Notes in Computer Science.

J. Li, F. Li, and S. Todorovic. Efficient Riemannian Optimization on the Stiefel Manifold via the Cayley Transform. Apr. 2020b. URL https://iclr.cc/virtual_2020/poster_HJxV-ANKDH.html.

X. Li, N. C. Dvornek, Y. Zhou, J. Zhuang, P. Ventola, and J. S. Duncan. Graph Neural Network for Interpreting Task-fMRI Biomarkers. In D. Shen, T. Liu, T. M. Peters, L. H. Staib, C. Essert, S. Zhou, P.-T. Yap, and A. Khan, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, Lecture Notes in Computer Science, pages 485–493, Cham, 2019. Springer International Publishing. ISBN 978-3-030-32254-0. doi: 10.1007/978-3-030-32254-0_54.

X. Liang, L. Wang, L.-H. Zhang, and R.-C. Li. On Generalizing Trace Minimization. *arXiv:2104.00257 [cs, math]*, Apr. 2021. URL http://arxiv.org/abs/2104.00257. arXiv: 2104.00257.

L. Lin, V. Rao, and D. Dunson. Bayesian nonparametric inference on the Stiefel manifold. *Statistica Sinica*, 27(2):535–553, 2017. ISSN 1017-0405. URL https://www.jstor.org/stable/26383289. Publisher: Institute of Statistical Science, Academia Sinica.

T. J. Littlejohns, J. Holliday, L. M. Gibson, S. Garratt, N. Oesingmann, F. Alfaro-Almagro, J. D. Bell, C. Boultwood, R. Collins, M. C. Conroy, N. Crabtree, N. Doherty, A. F. Frangi, N. C. Harvey, P. Leeson, K. L. Miller, S. Neubauer, S. E. Petersen, J. Sellors, S. Sheard, S. M. Smith, C. L. M. Sudlow, P. M. Matthews, and N. E. Allen. The UK Biobank imaging enhancement of 100,000 participants: rationale, data collection, management and future directions. *Nature Communications*, 11(1):2624, May 2020. ISSN 2041-1723. doi: 10.1038/s41467-020-15948-9. URL https://www.nature.com/articles/s41467-020-15948-9. Number: 1 Publisher: Nature Publishing Group.

M. Liu, Z. Zhang, and D. B. Dunson. Auto-encoding graph-valued data with applications to brain connectomes. *arXiv:1911.02728 [cs, q-bio, stat]*, Nov. 2019. URL http://arxiv.org/abs/1911.02728. arXiv: 1911.02728.

F. Llorente, L. Martino, D. Delgado, and J. Lopez-Santiago. Marginal likelihood computation for model selection and hypothesis testing: an extensive review. *arXiv:2005.08334 [cs, stat]*, Jan. 2022. URL http://arxiv.org/abs/2005.08334. arXiv: 2005.08334.

L. Lovász. *Large Networks and Graph Limits*, volume 60 of *Colloquium Publications*. American Mathematical Society, Providence, Rhode Island, Dec. 2012. ISBN 978-0-8218-9085-1 978-1-4704-1583-9. doi: 10.1090/coll/060. URL http://www.ams.org/coll/060.

L. Lu and T. Zhou. Link Prediction in Complex Networks: A Survey. *Physica A: Statistical Mechanics and its Applications*, 390(6):1150–1170, Mar. 2011. ISSN 03784371. doi: 10.1016/j.physa.2010.11.027. URL http://arxiv.org/abs/1010.0725. arXiv: 1010.0725.

S. Lunagómez, S. C. Olhede, and P. J. Wolfe. Modeling Network Populations via Graph Distances. *Journal of the American Statistical Association*, 116(536):2023–2040, Oct. 2021. ISSN 0162-1459, 1537-274X. doi: 10.1080/01621459.2020.1763803. URL https://www.tandfonline.com/doi/full/10.1080/01621459.2020.1763803.

G. Ma, C.-T. Lu, L. He, P. S. Yu, and A. B. Ragin. Multi-view Graph Embedding with Hub Detection for Brain Network Analysis. In *2017 IEEE International Conference on Data Mining (ICDM)*, pages 967–972, Nov. 2017. doi: 10.1109/ICDM.2017.123. ISSN: 2374-8486.

J. Ma, L. Xu, and M. I. Jordan. Asymptotic Convergence Rate of the EM Algorithm for Gaussian Mixtures. *Neural Computation*, 12(12):2881–2907, Dec. 2000. ISSN 0899-7667. doi: 10.1162/089976600300014764. URL https://doi.org/10.1162/089976600300014764.

J. Ma, X. Zhu, D. Yang, J. Chen, and G. Wu. Attention-Guided Deep Graph Neural Network for Longitudinal Alzheimer's Disease Analysis. In A. L. Martel, P. Abolmaesumi, D. Stoyanov, D. Mateus, M. A. Zuluaga, S. K. Zhou, D. Racoceanu, and L. Joskowicz, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, volume 12267, pages 387–396, Cham, 2020. Springer International Publishing. ISBN 978-3-030-59727-6 978-3-030-59728-3. doi: 10.1007/978-3-030-59728-3_38. URL http://link.springer.com/10.1007/978-3-030-59728-3_38. Series Title: Lecture Notes in Computer Science.

E. Macías-Virgós, M. Pereira-Sáez, and D. Tanré. Cayley transform on Stiefel manifolds. *Journal of Geometry and Physics*, 123:53–60, Jan. 2018. ISSN 0393-0440. doi: 10.1016/j.geomphys.2017.08.011. URL https://www.sciencedirect.com/science/article/pii/S039304401730205X.

C. Mantoux, B. Couvy-Duchesne, F. Cacciamani, S. Epelbaum, S. Durrleman, and S. Allassonnière. Understanding the Variability in Graph Data Sets through Statistical Modeling on the Stiefel Manifold. *Entropy*, 23(4):490, Apr. 2021. doi: 10.3390/e23040490. URL https://www.mdpi.com/1099-4300/23/4/490. Number: 4 Publisher: Multidisciplinary Digital Publishing Institute.

C. Mantoux, S. Durrleman, and S. Allassonnière. Asymptotic Analysis of a Matrix Latent Decomposition Model. *ESAIM: Probability and Statistics*, 26:208–242, 2022. ISSN 1262-3318. doi: 10.1051/ps/2022004. URL https://www.esaim-ps.org/articles/ps/abs/2022/01/ps220004/ps220004.html. Publisher: EDP Sciences.

K. Marek, D. Jennings, S. Lasch, A. Siderowf, C. Tanner, T. Simuni, C. Coffey, K. Kieburtz, E. Flagg, S. Chowdhury, W. Poewe, B. Mollenhauer, P.-E. Klinik, T. Sherer, M. Frasier, C. Meunier, A. Rudolph, C. Casaceli, J. Seibyl, S. Mendick, N. Schuff, Y. Zhang, A. Toga, K. Crawford, A. Ansbach, P. De Blasio, M. Piovella, J. Trojanowski, L. Shaw, A. Singleton, K. Hawkins, J. Eberling, D. Brooks, D. Russell, L. Leary, S. Factor, B. Sommerfeld, P. Hogarth, E. Pighetti, K. Williams, D. Standaert, S. Guthrie, R. Hauser, H. Delgado, J. Jankovic, C. Hunter, M. Stern, B. Tran, J. Leverenz, M. Baca, S. Frank, C.-A. Thomas, I. Richard, C. Deeley, L. Rees, F. Sprenger, E. Lang, H. Shill, S. Obradov, H. Fernandez, A. Winters, D. Berg, K. Gauss, D. Galasko, D. Fontaine, Z. Mari, M. Gerstenhaber, D. Brooks, S. Malloy, P. Barone, K. Longo, T. Comery, B. Ravina, I. Grachev, K. Gallagher, M. Collins, K. L. Widnell, S. Ostrowizki, P. Fontoura, T. Ho, J. Luthman, M. v. d. Brug, A. D. Reith, and P. Taylor. The Parkinson Progression Marker Initiative (PPMI). *Progress in Neurobiology*, 95(4):629–635, Dec. 2011. ISSN 0301-0082. doi: 10.1016/j.pneurobio.2011.09.005. URL https://www.sciencedirect.com/science/article/pii/S0301008211001651.

V. Martínez, F. Berzal, and J.-C. Cubero. A Survey of Link Prediction in Complex Networks. *ACM Computing Surveys*, 49(4):69:1–69:33, Dec. 2016. ISSN 0360-0300. doi: 10.1145/3012704. URL https://doi.org/10.1145/3012704.

E. Mathieu, C. Le Lan, C. J. Maddison, R. Tomioka, and Y. W. Teh. Continuous Hierarchical Representations with Poincaré Variational Auto-Encoders. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d. Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 12565–12576. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/9420-continuous-hierarchical-representations-with-poincare-variational-auto-encoders.pdf.

T. Matsuda, M. Uehara, and A. Hyvarinen. Information criteria for non-normalized models. *Journal of Machine Learning Research*, 22(158):1–33, 2021. ISSN 1533-7928. URL http://jmlr.org/papers/v22/20-1366.html.

G. J. McLachlan and T. Krishnan. *The EM algorithm and extensions*. Wiley series in probability and statistics. Wiley-Interscience, Hoboken, N.J, 2nd ed edition, 2008. ISBN 978-0-471-20170-0. OCLC: ocn137325058.

F. Meng. *Bayesian Inference on the Stiefel Manifold: Models, Applications and Algorithms*. PhD thesis, UC Santa Barbara, 2021. URL https://escholarship.org/uc/item/5wh0r5vq.

X.-L. Meng and W. H. Wong. Simulating Ratios of Normalizing Constants via a Simple Identity: A Theoretical Exploration. *Statistica Sinica*, 6(4):831–860, 1996. ISSN 1017-0405. URL https://www.jstor.org/stable/24306045. Publisher: Institute of Statistical Science, Academia Sinica.

E. C. Merkle, D. Furr, and S. Rabe-Hesketh. Bayesian Comparison of Latent Variable Models: Conditional Versus Marginal Likelihoods. *Psychometrika*, 84(3):802–829, Sept. 2019. ISSN 1860-0980. doi: 10.1007/s11336-019-09679-0. URL https://doi.org/10.1007/s11336-019-09679-0.

A. Mira and G. Nicholls. Bridge Estimation of the Probability Density at a Point. *Statistica Sinica*, 14(2):603–612, 2004. ISSN 1017-0405. URL https://www.jstor.org/stable/24307211. Publisher: Institute of Statistical Science, Academia Sinica.

F. Monti, M. Bronstein, and X. Bresson. Geometric Matrix Completion with Recurrent Multi-Graph Neural Networks. *Advances in Neural Information Processing Systems*, 30:3697–3707, 2017. URL https://papers.nips.cc/paper/2017/hash/2eace51d8f796d04991c831a07059758-Abstract.html.

S. S. Mukherjee and S. Chakrabarti. Graphon Estimation from Partially Observed Network Data. *arXiv:1906.00494 [cs, stat]*, June 2019. URL http://arxiv.org/abs/1906.00494. arXiv: 1906.00494.

K. P. Murphy. *Machine learning: a probabilistic perspective*. Adaptive computation and machine learning series. MIT Press, Cambridge, MA, 2012. ISBN 978-0-262-01802-9.

T. Narayanan and S. Subramaniam. Community Structure Analysis of Gene Interaction Networks in Duchenne Muscular Dystrophy. *PLoS ONE*, 8(6), June 2013. ISSN 1932-6203. doi: 10.1371/journal.pone.0067237. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3686745/.

Y. Nesterov. Nonsmooth Convex Optimization. In Y. Nesterov, editor, *Lectures on Convex Optimization*, Springer Optimization and Its Applications, pages 139–240. Springer International Publishing, Cham, 2018. ISBN 978-3-319-91578-4. doi: 10.1007/978-3-319-91578-4_3. URL https://doi.org/10.1007/978-3-319-91578-4_3.

M. E. J. Newman. *Networks - An Introduction*. Oxford University Press, Feb. 2012. URL https://doi.org/10.1162/artl_r_00062.

L. T. Nguyen, J. Kim, and B. Shim. Low-Rank Matrix Completion: A Contemporary Survey. *IEEE Access*, 7:94215–94237, 2019. ISSN 2169-3536. doi: 10.1109/ACCESS.2019.2928130. Conference Name: IEEE Access.

C.-C. Ni, Y.-Y. Lin, F. Luo, and J. Gao. Community Detection on Networks with Ricci Flow. *Scientific Reports*, 9(1):9984, Dec. 2019. ISSN 2045-2322. doi: 10.1038/s41598-019-46380-9. URL http://www.nature.com/articles/s41598-019-46380-9.

C. Obando and F. D. V. Fallani. A statistical model for brain networks inferred from large-scale electrophysiological signals. *Journal of The Royal Society Interface*, 14(128):20160940, Mar. 2017. ISSN 1742-5689, 1742-5662. doi: 10.1098/rsif.2016.0940. URL http://arxiv.org/abs/1611.06893. arXiv: 1611.06893.

C. Obando, C. Rosso, J. Siegel, M. Corbetta, and F. D. V. Fallani. Temporal connection signatures of human brain networks after stroke. *arXiv:1907.10009 [q-bio, stat]*, July 2019. URL http://arxiv.org/abs/1907.10009. arXiv: 1907.10009.

S. C. Olhede and P. J. Wolfe. Network histograms and universality of blockmodel approximation. *Proceedings of the National Academy of Sciences*, 111(41):14722–14727, Oct. 2014. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1400374111. URL http://arxiv.org/abs/1312.5306. arXiv: 1312.5306.

T. Opsahl and P. Panzarasa. Clustering in weighted networks. *Social Networks*, 31(2):155–163, May 2009. ISSN 0378-8733. doi: 10.1016/j.socnet.2009.02.002. URL http://www.sciencedirect.com/science/article/pii/S0378873309000070.

K. Oualkacha and L.-P. Rivest. On the estimation of an average rigid body motion. *Biometrika*, 99(3):585–598, 2012. ISSN 0006-3444. URL https://www.jstor.org/stable/41720716. Publisher: Biometrika Trust.

Z. Ouyang. *Bayesian Additive Regression Kernels*. PhD thesis, Duke University, 2008.

A. Owrang and M. Jansson. Model selection for high-dimensional data. In *2016 50th Asilomar Conference on Signals, Systems and Computers*, pages 606–609, Nov. 2016. doi: 10.1109/ACSSC.2016.7869114.

S. Oymak and B. Hassibi. Finding Dense Clusters via "Low Rank + Sparse" Decomposition. *arXiv:1104.5186 [cs, math, stat]*, Apr. 2011. URL http://arxiv.org/abs/1104.5186. arXiv: 1104.5186 version: 1.

S. Pal, S. Sengupta, R. Mitra, and A. Banerjee. Conjugate Priors and Posterior Inference for the Matrix Langevin Distribution on the Stiefel Manifold. *Bayesian Analysis*, 15(3):871–908, Sept. 2020. ISSN 1936-0975. doi: 10.1214/19-BA1176. URL https://projecteuclid.org/euclid.ba/1570586976.

N. Parikh and S. Boyd. *Proximal Algorithms*. Now Publishers Inc., Jan. 2014. URL https://doi.org/10.1561/2400000003.

T. P. Peixoto. Bayesian stochastic blockmodeling. In P. Doreian, V. Batagelj, and A. Ferligoj, editors, *Advances in Network Clustering and Blockmodeling*, Wiley Series in Computational and Quantitative Social Science, pages 289–332. Wiley, Feb. 2020. ISBN 978-1-119-22470-9. URL http://arxiv.org/abs/1705.10225. arXiv: 1705.10225.

X. Pennec. Intrinsic Statistics on Riemannian Manifolds: Basic Tools for Geometric Measurements. *Journal of Mathematical Imaging and Vision*, 25(1):127–154, July 2006. ISSN 0924-9907, 1573-7683. doi: 10.1007/s10851-006-6228-4. URL http://link.springer.com/10.1007/s10851-006-6228-4.

G. Peyré. Chapter 13 Convex Optimization. In *Mathematical foundations of data sciences*. 2018.

F. A. Pozzi, E. Fersini, E. Messina, and B. Liu. *Sentiment analysis in social networks*. Morgan Kaufmann, 2016.

D. Purves, G. J. Augustine, D. Fitzpatrick, W. C. Hall, A.-S. LaMantia, R. D. Mooney, M. L. Platt, and L. E. White, editors. *Neuroscience*. Sinauer Associates is an imprint of Oxford University Press, New York, 6th edition edition, Oct. 2017. ISBN 978-1-60535-380-7.

H. Raguet, J. Fadili, and G. Peyré. Generalized Forward-Backward Splitting. *SIAM Journal on Imaging Sciences*, 6(3):1199–1226, Jan. 2013. ISSN 1936-4954. doi: 10.1137/120872802. URL http://arxiv.org/abs/1108.4404. arXiv: 1108.4404.

L. L. Raket. Statistical Disease Progression Modeling in Alzheimer Disease. *Frontiers in Big Data*, 3, 2020. ISSN 2624-909X. doi: 10.3389/fdata.2020.00024. URL https://www.frontiersin.org/articles/10.3389/fdata.2020.00024/full. Publisher: Frontiers.

X. Ren, J. Lin, G. T. Stebbins, C. G. Goetz, and S. Luo. Prognostic Modeling of Parkinson's Disease Progression Using Early Longitudinal Patterns of Change. *Movement Disorders*, 36(12):2853–2861, 2021. ISSN 1531-8257. doi: 10.1002/mds.28730. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/mds.28730. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/mds.28730.

Z. Ren, T. Sun, C.-H. Zhang, and H. H. Zhou. Asymptotic normality and optimalities in estimation of large Gaussian graphical models. *The Annals of Statistics*, 43(3), June 2015. ISSN 0090-5364. doi: 10.1214/14-AOS1286. URL http://arxiv.org/abs/1309.6024. arXiv: 1309.6024.

Q. Rentmeesters. *Algorithms for data fitting on some common homogeneous spaces*. PhD thesis, UCL - Université Catholique de Louvain, 2013. URL https://dial.uclouvain.be/pr/boreal/object/boreal:132587.

E. Richard, P.-A. Savalle, and N. Vayatis. Estimation of Simultaneously Sparse and Low Rank Matrices. In *ICML 2012*, June 2012. URL http://arxiv.org/abs/1206.6474. arXiv: 1206.6474.

E. Richard, F. Bach, and J.-P. Vert. Intersecting singularities for multi-structured estimation. In *ICML 2013 - 30th International Conference on Machine Learning*, pages –, Atlanta, United States, June 2013. URL https://hal.archives-ouvertes.fr/hal-00918253.

J. Richiardi, S. Achard, H. Bunke, and D. Van De Ville. Machine Learning with Brain Graphs: Predictive Modeling Approaches for Functional Imaging in Systems Neuroscience. *IEEE Signal Processing Magazine*, 30(3):58–70, May 2013. ISSN 1558-0792. doi: 10.1109/MSP.2012.2233865. Conference Name: IEEE Signal Processing Magazine.

C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer, New York, NY, Nov. 2010. ISBN 978-1-4419-1939-7.

G. Rosenthal, F. Váša, A. Griffa, P. Hagmann, E. Amico, J. Goñi, G. Avidan, and O. Sporns. Mapping higher-order relations between brain structure and function with embedded vector representations of connectomes. *Nature Communications*, 9, June 2018. ISSN 2041-1723. doi: 10.1038/s41467-018-04614-w. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5988787/.

M. Rubinov and O. Sporns. Complex network measures of brain connectivity: Uses and interpretations. *NeuroImage*, 52(3):1059–1069, Sept. 2010. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2009.10.003. URL http://www.sciencedirect.com/science/article/pii/S105381190901074X.

J.-B. Schiratti. *Methods and algorithms to learn spatio-temporal changes from longitudinal manifold-valued observations*. These de doctorat, Université Paris-Saclay (ComUE), Jan. 2017. URL https://theses.fr/2017SACLX009.

J.-B. Schiratti, S. Allassonnière, O. Colliot, and S. Durrleman. Learning spatiotemporal trajectories from manifold-valued longitudinal data. *Advances in Neural Information Processing Systems*, 28:2404–2412, 2015. URL https://papers.nips.cc/paper/2015/hash/186a157b2992e7daed3677ce8e9fe40f-Abstract.html.

G. Schwarz. Estimating the Dimension of a Model. *The Annals of Statistics*, 6 (2):461–464, Mar. 1978. ISSN 0090-5364, 2168-8966. doi: 10.1214/aos/1176344136. URL https://projecteuclid.org/journals/annals-of-statistics/volume-6/issue-2/Estimating-the-Dimension-of-a-Model/10.1214/aos/1176344136.full. Publisher: Institute of Mathematical Statistics.

K. A. Severson, L. M. Chahine, L. A. Smolensky, M. Dhuliawala, M. Frasier, K. Ng, S. Ghosh, and J. Hu. Discovery of Parkinson's disease states and disease progression modelling: a longitudinal data study using machine learning. *The Lancet Digital Health*, 3(9):e555–e564, Sept. 2021. ISSN 2589-7500. doi: 10.1016/S2589-7500(21)00101-1. URL https://www.sciencedirect.com/science/article/pii/S2589750021001011.

A. A. Shabalin and A. B. Nobel. Reconstruction of a low-rank matrix in the presence of Gaussian noise. *Journal of Multivariate Analysis*, 118:67–76, July 2013. ISSN 0047-259X. doi: 10.1016/j.jmva.2013.03.005. URL http://www.sciencedirect.com/science/article/pii/S0047259X13000328.

X. Shen, E. S. Finn, D. Scheinost, M. D. Rosenberg, M. M. Chun, X. Papademetris, and R. T. Constable. Using connectome-based predictive modeling to predict individual behavior from brain connectivity. *Nature Protocols*, 12(3):506–518, 2017. ISSN 1750-2799. doi: 10.1038/nprot.2016.178.

M. Signorelli and E. C. Wit. Model-based clustering for populations of networks. *Statistical Modelling*, 20(1):9–29, Feb. 2020. ISSN 1471-082X. doi: 10.1177/1471082X19871128. URL https://doi.org/10.1177/1471082X19871128. Publisher: SAGE Publications India.

M. Simonovsky and N. Komodakis. GraphVAE: Towards Generation of Small Graphs Using Variational Autoencoders. In V. Kůrková, Y. Manolopoulos, B. Hammer, L. Iliadis, and I. Maglogiannis, editors, *Artificial Neural Networks and Machine Learning – ICANN 2018*, Lecture Notes in Computer Science, pages 412–422, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01418-6. doi: 10.1007/978-3-030-01418-6_41.

B. Sischka and G. Kauermann. EM-based smooth graphon estimation using MCMC and spline-based approaches. *Social Networks*, 68:279–295, Jan. 2022. ISSN 0378-8733. doi: 10.1016/j.socnet.2021.08.007. URL https://www.sciencedirect.com/science/article/pii/S0378873321000691.

D. J. Spiegelhalter, N. G. Best, B. P. Carlin, and A. van der Linde. The deviance information criterion: 12 years on. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(3):485–493, 2014. ISSN 1467-9868. doi: 10.1111/rssb.12062. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/rssb.12062. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/rssb.12062.

B. Strikholm. Determining the number of breaks in a piecewise linear regression model. Technical Report 648, Stockholm School of Economics, Dec. 2006. URL https://ideas.repec.org/p/hhs/hastef/0648.html. Publication Title: SSE/EFI Working Paper Series in Economics and Finance.

C. Sudlow, J. Gallacher, N. Allen, V. Beral, P. Burton, J. Danesh, P. Downey, P. Elliott, J. Green, M. Landray, B. Liu, P. Matthews, G. Ong, J. Pell, A. Silman, A. Young, T. Sprosen, T. Peakman, and R. Collins. UK Biobank: An Open Access Resource for Identifying the Causes of a Wide Range of Complex Diseases of Middle and Old Age. *PLOS Medicine*, 12(3):e1001779, Mar. 2015. ISSN 1549-1676. doi: 10.1371/journal.pmed.1001779. URL https://journals.plos.org/plosmedicine/article?id=10.1371/journal.pmed.1001779. Publisher: Public Library of Science.

E. Tabrizi, E. B. Samani, and M. Ganjali. A note on the identifiability of latent variable models for mixed longitudinal data. *Statistics & Probability Letters*, 167:108882, Dec. 2020. ISSN 0167-7152. doi: 10.1016/j.spl.2020.108882. URL https://www.sciencedirect.com/science/article/pii/S0167715220301851.

B. Tadić, M. Andjelković, and R. Melnik. Functional Geometry of Human Connectomes. *Scientific Reports*, 9(1):12060, Dec. 2019. ISSN 2045-2322. doi: 10.1038/s41598-019-48568-5. URL http://www.nature.com/articles/s41598-019-48568-5.

S. Takerkart, G. Auzias, B. Thirion, and L. Ralaivola. Graph-Based Inter-Subject Pattern Analysis of fMRI Data. *PLoS ONE*, 9(8), Aug. 2014. ISSN 1932-6203. doi: 10.1371/journal.pone.0104586. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4134217/.

K. Takeuchi. Distribution of information statistics and criteria for adequacy of models. *Mathematical Science*, 153:12–18, 1976.

H. Teicher. Identifiability of Finite Mixtures. *The Annals of Mathematical Statistics*, 34(4):1265–1269, Dec. 1963. ISSN 0003-4851, 2168-8990. doi: 10.1214/aoms/1177703862. URL https://projecteuclid.org/journals/annals-of-mathematical-statistics/volume-34/issue-4/Identifiability-of-Finite-Mixtures/10.1214/aoms/1177703862.full. Publisher: Institute of Mathematical Statistics.

P. Tewarie, B. Prasse, J. M. Meier, F. A. N. Santos, L. Douw, M. M. Schoonheim, C. J. Stam, P. Van Mieghem, and A. Hillebrand. Mapping functional brain networks from the structural connectome: Relating the series expansion and eigenmode approaches. *NeuroImage*, 216:116805, Aug. 2020. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2020.116805. URL http://www.sciencedirect.com/science/article/pii/S1053811920302925.

T. Traynor. Change of Variables for Hausdorff measure (from the beginning). *Università degli Studi di Trieste. Dipartimento di Scienze Matematiche*, 26 suppl.:327–347, 1994. ISSN 0049-4704. URL https://www.openstarts.units.it/handle/10077/4629. Accepted: 2011-05-24T11:06:04Z Publisher:.

A. W. van der Vaart. *Asymptotic statistics*. Cambridge series in statistical and probabilistic mathematics. Cambridge Univ. Press, Cambridge, 1. paperback ed., 8. printing edition, 1998. ISBN 978-0-521-49603-2 978-0-521-78450-4. OCLC: 838749444.

K. van Montfort, J. H. Oud, and A. Satorra, editors. *Longitudinal Research with Latent Variables*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. ISBN 978-3-642-11759-6 978-3-642-11760-2. doi: 10.1007/978-3-642-11760-2. URL http://link.springer.com/10.1007/978-3-642-11760-2.

C. S. Venuto, N. B. Potter, E. Ray Dorsey, and K. Kieburtz. A review of disease progression models of Parkinson's disease and applications in clinical trials. *Movement Disorders*, 31(7):947–956, 2016. ISSN 1531-8257. doi: 10.1002/mds.26644. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/mds.26644. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/mds.26644.

L. Wang, Z. Zhang, and D. Dunson. Common and individual structure of brain networks. *The Annals of Applied Statistics*, 13(1):85–112, Mar. 2019. ISSN 1932-6157, 1941-7330. doi: 10.1214/18-AOAS1193. URL https://projecteuclid.org/journals/annals-of-applied-statistics/volume-13/issue-1/Common-and-individual-structure-of-brain-networks/10.1214/18-AOAS1193.full. Publisher: Institute of Mathematical Statistics.

L. Waschke, M. Alavash, J. Obleser, and J. Erb. AUDADAPT. Nov. 2018. doi: 10.17605/OSF.IO/28R57. URL https://osf.io/28r57/. Publisher: OSF.

S. Watanabe. A Widely Applicable Bayesian Information Criterion. *Journal of Machine Learning Research*, 14(Mar):867–897, 2013. ISSN ISSN 1533-7928. URL https://jmlr.csail.mit.edu/papers/v14/watanabe13a.html.

Z. Wen and W. Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142(1):397–434, Dec. 2013. ISSN 1436-4646. doi: 10.1007/s10107-012-0584-1. URL https://doi.org/10.1007/s10107-012-0584-1.

A. H. Westveld and P. D. Hoff. A mixed effects model for longitudinal relational and network data, with applications to international trade and conflict. *The Annals of Applied Statistics*, 5 (2A):843–872, June 2011. ISSN 1932-6157. doi: 10.1214/10-AOAS403. URL http://arxiv.org/abs/1009.1436. arXiv: 1009.1436.

M. J. Williams and M. Musolesi. Spatio-temporal networks: reachability, centrality and robustness. *Royal Society Open Science*, 3(6):160196, June 2016. ISSN 2054-5703, 2054-5703. doi: 10.1098/rsos.160196. URL https://royalsocietypublishing.org/doi/10.1098/rsos.160196.

S. Wold, M. Sjöström, and L. Eriksson. PLS-regression: a basic tool of chemometrics. *Chemometrics and Intelligent Laboratory Systems*, 58(2):109–130, Oct. 2001. ISSN 0169-7439. doi: 10.1016/S0169-7439(01)00155-1. URL https://www.sciencedirect.com/science/article/pii/S0169743901001551.

P. J. Wolfe and S. C. Olhede. Nonparametric graphon estimation. *arXiv:1309.5936 [math, stat]*, Sept. 2013. URL http://arxiv.org/abs/1309.5936. arXiv: 1309.5936.

Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu. A Comprehensive Survey on Graph Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–21, 2020. ISSN 2162-237X, 2162-2388. doi: 10.1109/TNNLS.2020.2978386. URL http://arxiv.org/abs/1901.00596. arXiv: 1901.00596.

J. Xu. Rates of Convergence of Spectral Methods for Graphon Estimation. In *International Conference on Machine Learning*, pages 5433–5442, July 2018. URL http://proceedings.mlr.press/v80/xu18a.html.

M. Xu, Z. Wang, H. Zhang, D. Pantazis, H. Wang, and Q. Li. Gaussian embedding-based functional brain connectomic analysis for amnestic mild cognitive impairment patients with cognitive training. preprint, Neuroscience, Sept. 2019. URL http://biorxiv.org/lookup/doi/10.1101/779744.

S. J. Yakowitz and J. D. Spragins. On the Identifiability of Finite Mixtures. *The Annals of Mathematical Statistics*, 39(1):209–214, 1968. ISSN 0003-4851. URL https://www.jstor.org/stable/2238925. Publisher: Institute of Mathematical Statistics.

J. Zhang, W. W. Sun, and L. Li. Mixed-effect time-varying network model and application in brain connectivity analysis. *Journal of the American Statistical Association*, 115(532):2022–2036, 2020a. URL http://arxiv.org/abs/1806.03829. arXiv: 1806.03829 Publisher: Taylor & Francis.

M. Zhang and Y. Chen. Link Prediction Based on Graph Neural Networks. *Advances in Neural Information Processing Systems*, 31:5171–5181, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/53f0d7c537d99b3824f0f99d62ea2428-Abstract.html.

Y. Zhang. Consistent polynomial-time unseeded graph matching for Lipschitz graphons. *arXiv:1807.11027 [cs, math, stat]*, July 2018. URL http://arxiv.org/abs/1807.11027. arXiv: 1807.11027 version: 1.

Z. Zhang, P. Cui, and W. Zhu. Deep Learning on Graphs: A Survey. *arXiv:1812.04202 [cs, stat]*, Mar. 2020b. URL http://arxiv.org/abs/1812.04202. arXiv: 1812.04202.

M. Zheng, A. Allard, P. Hagmann, Y. Alemán-Gómez, and M. Ángeles Serrano. Geometric renormalization unravels self-similarity of the multiscale human connectome. *arXiv e-prints*, 1904: arXiv:1904.11793, Apr. 2019a. URL http://adsabs.harvard.edu/abs/2019arXiv190411793Z.

W. Zheng, Z. Yao, Y. Li, Y. Zhang, B. Hu, D. Wu, and f. t. A. D. N. Initiative. Brain Connectivity Based Prediction of Alzheimer's Disease in Patients With Mild Cognitive Impairment Based on Multi-Modal Images. *Frontiers in Human Neuroscience*, 13, 2019b. ISSN 1662-5161. doi: 10.3389/fnhum.2019.00399. URL https://www.frontiersin.org/articles/10.3389/fnhum.2019.00399/full. Publisher: Frontiers.

S.-L. Zhou, N.-H. Xiu, Z.-Y. Luo, and L.-C. Kong. Sparse and Low-Rank Covariance Matrix Estimation. *Journal of the Operations Research Society of China*, 3(2):231–250, June 2015. ISSN 2194-6698. doi: 10.1007/s40305-014-0058-7. URL https://doi.org/10.1007/s40305-014-0058-7.

Y. Zhou and H.-G. Müller. Dynamic Network Regression. *arXiv:2109.02981 [stat]*, Sept. 2021. URL http://arxiv.org/abs/2109.02981. arXiv: 2109.02981.

X. Zhu and H. Sato. Cayley-transform-based gradient and conjugate gradient algorithms on Grassmann manifolds. *Advances in Computational Mathematics*, 47(4):56, Aug. 2021. ISSN 1019-7168, 1572-9044. doi: 10.1007/s10444-021-09880-9. URL https://link.springer.com/10.1007/s10444-021-09880-9.

R. Zimmermann. A Matrix-Algebraic Algorithm for the Riemannian Logarithm on the Stiefel Manifold under the Canonical Metric. *SIAM Journal on Matrix Analysis and Applications*, 38 (2):322–342, Jan. 2017. ISSN 0895-4798. doi: 10.1137/16M1074485. URL https://epubs.siam.org/doi/abs/10.1137/16M1074485. Publisher: Society for Industrial and Applied Mathematics.

H. I. Zonneveld, R. H. Pruim, D. Bos, H. A. Vrooman, R. L. Muetzel, A. Hofman, S. A. Rombouts, A. van der Lugt, W. J. Niessen, M. A. Ikram, and M. W. Vernooij. Patterns of functional connectivity in an aging population: The Rotterdam Study. *NeuroImage*, 189:432–444, Apr. 2019. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2019.01.041. URL http://www.sciencedirect.com/science/article/pii/S1053811919300412.

M. Zorzi and A. Chiuso. Sparse plus low rank network identification: A nonparametric approach. *Automatica*, 76:355–366, Feb. 2017. ISSN 0005-1098. doi: 10.1016/j.automatica.2016.08.014. URL http://www.sciencedirect.com/science/article/pii/S0005109816303946.

ECOLE
DOCTORALE
DE MATHEMATIQUES
HADAMARD

Titre : Modélisation statistique et inférence pour les populations de réseaux de connectivité cérébrale et les données longitudinales

Mots clés : Modélisation statistique, Populations de réseaux, Connectivité cérébrale, Données longitudinales

Résumé : Le développement et la massification des bases de données d'imagerie médicale et de suivi clinique ouvrent de nouvelles perspectives pour la compréhension de phénomènes complexes comme le vieillissement ou les maladies neurodégénératives. En particulier, la connectivité cérébrale, c'est-à-dire l'étude des connexions et des interactions entre les régions du cerveau, peut maintenant être étudiée à l'échelle d'une population et non plus d'un individu isolé. Ce cadre offre la possibilité d'une meilleure prise en compte des spécificités individuelles dans le développement d'outils de suivi.
Dans cette thèse, nous proposons dans un premier temps de nouvelles approches pour modéliser et comprendre la variabilité de la connectivité cérébrale au sein d'un groupe de sujets. Plus généralement, nous nous intéressons aux collections de réseaux où chaque réseau décrit des interactions entre les mêmes entités. Nous nous appuyons sur la propriété empirique de rang faible des matrices d'adjacence de ces réseaux pour rendre compte de leur distribution. Nous proposons deux approches, l'une variationnelle et l'autre statistique, pour rendre compte de l'hétérogénéité de ces matrices. En particulier, dans le second cas, nous montrons qu'un nombre restreint de paramètres suffit à donner une description fidèle et interprétable de la variabilité de la connectivité cérébrale. Nous montrons également la consistence et l'identifiabilité de notre approche sur le plan théorique.
Dans un second temps, nous étudions un modèle longitudinal pour le suivi de la progression de la maladie de Parkinson. Dans ce modèle, la trajectoire de chaque patient est divisée en plusieurs morceaux pouvant correspondre aux différentes phases de la maladie ou d'un traitement. Nous nous montrons qu'il est possible d'estimer les trajectoires constituées de plusieurs morceaux, et de sélectionner le nombre de ruptures le mieux adapté pour décrire l'évolution moyenne de la population.

Title : Statistical Modeling and Inference for Populations of Networks and Longitudinal Data

Keywords : Statistical modeling, Populations of networks, Brain connectivity, Longitudinal data

Abstract : The development and massification of medical imaging and clinical followup databases open up new perspectives for understanding complex phenomena such as ageing or neurodegenerative diseases. In particular, brain connectivity, i.e., the study of connections and interactions between brain regions, can now be studied on the scale of a population scale rather than on an individual basis. This framework offers the possibility of better taking into account individual specificities in the development of monitoring tools.
In this thesis, we first propose new approaches to model and understand the variability of brain connectivity within a group of subjects. More generally, we are interested in collections of networks where each network describes interactions between the same entities. We rely on the empirical low rank property of the adjacency matrices of these networks to account for their distribution. We propose two approaches, one variational and the other statistical, to account for the heterogeneity of these matrices. In particular, in the second case, we show that a limited number of parameters is sufficient to give a faithful and interpretable description of the variability of brain connectivity. We also show the theoretical consistency and identifiability of our approach.
In a second part, we study a longitudinal model for the progression monitoring of Parkinson's disease. In this model, the trajectory of each patient is divided into several pieces that may correspond to the different phases of the disease or of a treatment. We show that it is possible to estimate trajectories consisting of several pieces, and to select the number of breaks best suited to describe the average evolution of the population.

Institut Polytechnique de Paris
91120 Palaiseau, France