A Riemannian-Geometry approach to the online estimation of elliptical distribution

Jialun Zhou

THÈSE PRÉSENTÉE
POUR OBTENIR LE GRADE DE

DOCTEUR DE L'UNIVERSITÉ DE BORDEAUX

ÉCOLE DOCTORALE SCIENCES PHYSIQUES ET DE L'INGÉNIEUR

Spécialité Automatique, Productique, Signal, Image et Ingénierie Cognitique

Par **ZHOU Jialun**

Une approche basée sur la géométrie riemannienne pour l'estimation en ligne des distributions elliptiques

---

Sous la direction de : **SAID Salem**

Date de soutenance: 27 Octobre 2021

Membres du jury :

M.LE BIHAN Nicolas	Directeur de Recherche	CNRS, Gipsa-lab	Président du jury
M.JUTTEN Christian	Professeur	Université Grenoble Alpes	Rapporteur
M.PASCAL Frédéric	Professeur	CentraleSupélec	Rapporteur
M.LE BIHAN Nicolas	Directeur de Recherche	CNRS, Gipsa-lab	Examinateur
M.BOUCHARD Florent	Chargé de Recherche	CNRS, L2S	Examinateur
M.SAID Salem	Chargé de Recherche	CNRS, IMS	Directeur de thèse

Préparée à l'université de Bordeaux
Laboratoire d'accueil : Laboratoire IMS
351, avenue de la libération - 33405 Talence cedex

Titre : Une approche basée sur la géométrie Riemannienne pour l'estimation en ligne des distributions elliptiques

Résumé : Durant les dix dernières années, les méthodes basées sur la géométrie Riemannienne et la géométrie de l'information ont eu un impact important sur le traitement des signaux et des images, la science des données, et l'intelligence artificielle. L'objectif de cette thèse est de proposer de nouveaux algorithmes, basés sur la géométrie Riemannienne et la géométrie de l'information, pour l'estimation en ligne des lois dites à contours elliptiques, et de leurs mélanges. En général, l'estimation en ligne est réalisée à travers la minimisation d'une divergence statistique, la divergence de Kullback-Leibler, grâce à l'application d'une méthode de gradient Riemannien stochastique. Afin d'implémenter cette méthode, l'espace des paramètres de la famille de lois elliptiques (ou de mélanges de lois elliptiques) doit être équipé d'une métrique Riemannienne, de préférence la métrique d'information de Fisher. Malheureusement, pour les lois elliptiques, cette métrique est souvent inconnue, ou n'a pas d'expression analytique exploitable. Pour répondre à cette difficulté, nous avons introduit une alternative à la métrique d'information de Fisher, que nous avons appelée métrique d'information par composantes. En utilisant cette métrique, nous avons développé la méthode du gradient d'information par composantes. La méthode du gradient d'information par composantes est une méthode en ligne, avec un faible coût calculatoire, qui lui permet de prendre en compte les jeux de donnés massifs ou de grandes dimensions. De plus, cette méthode a deux variantes, une à pas d'optimisation décroissants, et l'autre à pas d'optimisation adaptatifs. Cette seconde variante permet d'éviter le choix manuel (habituellement très long et pénible) des pas d'optimisation, et d'atteindre une vitesse de convergence qui s'approche d'une vitesse exponentielle. Nous avons appliqué la méthode du gradient d'information par composantes à l'estimation de deux familles de lois elliptiques, les lois Gaussiennes généralisée multivariées, et les lois de Student multivariées, ce qui nous a permis de mettre en évidence à la fois son faible coût calculatoire et sa vitesse de convergence optimale. Finalement, nous avons réalisé des applications concrètes, en traitement des images et vision par ordinateur, à la conversion de couleurs et à la classification de textures. Pour les images de haute résolution (avec plus de 2 millions de pixels), notre méthode du gradient d'information par composantes n'a besoin que d'une centaine de secondes pour effectuer le travail, avec des résultats nettement meilleurs qu'avec les autres méthodes.

Mots clés : Estimation en ligne, Géométrie Riemannienne, Gradient d'information, traitement des signaux et images

---

Title : A Riemannian-Geometry approach to the online estimation of elliptical distribution

Abstract : Over the past ten years, methods based on Riemannian geometry and on information geometry have had a notable impact on signal and image processing, as well as on data science and artificial intelligence. The aim of this thesis is to

propose new algorithms, based on Riemannian geometry and information geometry, for the online estimation of so-called elliptically-contoured distributions and of their mixture distributions. In general, online estimation is achieved by minimising a statistical divergence function, the Kullback-Leibler divergence, using a Riemannian stochastic gradient method. In order to implement this method, the parameter space of the family of elliptically-contoured distributions has to be equipped with a Riemannian metric, preferably the Fisher information metric. Unfortunately, this metric is often unknown, or does not have a useful closed-form expression. In order to overcome this difficulty, we have introduced an alternative metric, which we have called the component-wise information metric. Using this metric, we have developed the component-wise information gradient method. The component-wise information gradient method is an online method, with a low computational cost, which allows it to process large or high-dimensional datasets. Moreover, this method has two versions, a decreasing step-size version and an adaptive step-size version. This second version avoids the often very laborious manual choice of step-sizes, and also achieves a nearly-exponential rate of convergence. We have applied the component-wise information gradient method to two families of elliptically-contoured distributions, multivariate generalised Gaussian distributions, and multivariate Student t-distributions, and experimentally verified the low computational cost and optimal rate of convergence of this method. Finally, we have carried out two concrete applications, in image processing and computer vision, to color conversion and to texture classification. For high-resolution images (with more than 2 million pixels), our component-wise information gradient method only needs about a hundred seconds in order to do the job, with much better results, in comparison to other methods.

Keywords : Online estimation, Riemannian geometry, Information gradient, Signal and image processing

---

Unité de recherche

Université de Bordeaux

Laboratoire IMS

351, avenue de la libération - 33405 Talence cedex

Remerciements

Tout d'abord, je remercie sincèrement mon directeur de thèse, Salem Said. Pendant mes trois années de thèse, il m'a encouragé, soutenu, et aidé à découvrir le monde de la recherche. Il m'a aussi aidé à construire mon sujet de thèse, en m'initiant à plusieurs domaines passionnants, comme la géométrie riemannienne, la géométrie de l'information, ou les algorithmes stochastiques pour l'estimation en ligne.

Je tiens aussi à remercier Yannick Berthoumieu, le directeur du groupe Signal et Image, dans lequel j'ai effectué mes trois années de thèse. Yannick et Salem m'ont toujours écouté avec patience, et m'ont accordé beaucoup de temps, pour m'aider à mener à bien mon parcours de doctorant. Pour leur engagement, et pour les nombreux conseils très précieux qu'ils m'ont donnés, je leur exprime ma grande reconnaissance.

J'aimerais également dire merci aux membres de mon jury de thèse, Christian Jutten, Frédéric Pascal, Nicolas Le Bihan, et Florent Bouchard, pour le temps qu'ils ont accepté de consacrer à lire mon manuscrit de thèse, et leurs remarques qui vont sans doute enrichir mon manuscrit et me permettre de progresser, en corrigeant les erreurs qu'il contient et en envisageant de nouveaux problèmes, pour la suite de mon travail.

Finalement, toute ma gratitude à mes parents et ma copine, pour leur amour et leur soutien, au fil des ans. C'est cet amour qui m'a accompagné, durant les dix années de mon parcours d'études scientifiques en France.

List of publications

Journal paper

Jialun Zhou, Salem Said.
Fast, asymptotically efficient, recursive estimation in a Riemannian manifold.
Entropy, 2019.
Jialun Zhou, Salem Said, Yannick Berthoumieu.
Riemannian information gradient methods for the parameter estimation of ECD
Accepted by Elsevier Signal Processing.

Conference papers

Jialun Zhou, Salem Said, Yannick Berthoumieu.
Online estimation of MGGD: the Riemannian averaged natural gradient method.
2019, IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP 2019).

Jialun Zhou, Salem Said, Yannick Berthoumieu.
Recursive estimation of the scatter matrix of ECD: the Riemannian Information Gradient method.
IFAC-PapersOnLine, 2021.

Journal papers under review

Jialun Zhou, Salem Said, Yannick Berthoumieu.
Online estimation of Mixtures of ECD: Component-wise Information Gradient method
IEEE Transactions on Signal Processing, under review.

List of Figures

List of Tables

List of notations

Symbol	Meaning
n	index of iteration
N	number of iterations
k	index of mixture component
K	number of components
t	index of datapoint
T	number of datapoints
x	datapoint
$\mathcal{X}$	dataset
$\mathcal{X}_{n+1}$ or $\mathcal{X}_{mb}^{new}$	mini-batch
$p(x\|\theta)$ or p_θ	density of an ECD with parameter θ
ℓ_p	log-likelihood function of an ECD
θ	parameter of an ECD
Θ	parameter space of an ECD
μ	location parameter
Σ	scatter matrix
β	shape parameter
δ	$(x-\mu)^\dagger\Sigma^{-1}(x-\mu)$
$f(x\|\boldsymbol{\theta})$ or f_θ	density of a MECD with parameter $\boldsymbol{\theta}$
ℓ_f	log-likelihood function of an MECD
$\boldsymbol{\theta}$	parameter of a MECD
$\boldsymbol{\Theta}$	parameter space of a MECD
w	the mixture weights
θ_k	the parameter of the k-th component
v, U, or $U(\theta, x)$	tangent vector, or descent direction
$P = (P, \Theta, X)$	statistical (parametric) model with parameter θ
$D(\theta)$	KL divergence $D(\theta\|\|\theta^*)$
θ^*	the true parameter
$\langle\cdot,\cdot\rangle$ or $\langle\cdot,\cdot\rangle_\theta$	Riemannian metric in general
$\langle\cdot,\cdot\rangle^\uparrow$ or $\langle\cdot,\cdot\rangle_\theta^\uparrow$	affine-invariant metric (at point θ)
$\langle\cdot,\cdot\rangle^*$ or $\langle\cdot,\cdot\rangle_\theta^*$	Fisher information metric (at point θ)
$\langle\cdot,\cdot\rangle^\odot$ or $\langle\cdot,\cdot\rangle_\theta^\odot$	component-wise information metric (at point θ)
$d(\cdot,\cdot)$	Riemannian distance in general
$d_\uparrow(\cdot,\cdot)$	affine-invariant distance
$d_*(\cdot,\cdot)$	information distance
$d_\odot(\cdot,\cdot)$	component-wise Information distance
$\mathrm{grad}_\theta\ell$	classic Euclidean gradient of function ℓ with respect to θ
$\nabla_\theta\ell$	Riemannian gradient of ℓ with respect to θ in general
$\nabla_\theta^\uparrow\ell$	affine-invariant gradient of function ℓ with respect to θ

$\nabla^{*}_{\theta}\ell$	information gradient of function ℓ with respect to θ
$\nabla^{\odot}_{\theta}\ell$	component-wise information gradient of function ℓ with respect to θ
$\eta^{(n)}$	step-size

Table of acronyms

Acronym	**Meaning**
ECD	Elliptically Contoured Distribution
MGGD	Multivariate Generalized Gaussian Distribution
MECD	Mixture of Elliptically Contoured Distributions
FP	Fixed Point method
EMFP	Expectation-Maximisation with Fixed Point method
SG	Stochastic Gradient methode
FIM	Fisher Information Metric
CIM	Component-wise Information Metric
CIG	Component-wise Information Gradient
CIG-DS	Component-wise Information Gradient with Decreasing Step-size
CIG-OB	Component-wise Information Gradient with Online Backtracking
CIG-AS	Component-wise Information Gradient with Adaptive Step-size

Contents

Résumé en Français

Cette thèse s'appuie sur trois articles de revue [88, 90, 89], dont le premier est déjà paru dans le journal *Entropy*, et les deux autres sont en cours de révision, dans les journaux *Signal Processing* et *IEEE Transactions on Signal Processing*, respectivement. Sa contribution principale est de proposer une nouvelle famille de méthodes pour l'estimation des lois à contours elliptiques et de leurs mélanges, que nous avons appelées méthodes CIG (en anglais, component-wise information gradient).

Les lois à contours elliptiques

Rappelons qu'une loi à contours elliptiques est une loi de probabilité sur $\mathbb{R}^m$, qui a une densité de probabilité de la forme

$$p(x|\theta) = \frac{c(\beta)}{\sqrt{\det(\Sigma)}} g_\beta(\delta) \tag{1}$$

où $\theta = (\mu, \Sigma, \beta)$, avec $\mu \in \mathbb{R}^m$ le paramètre de position, Σ une matrice définie positive, dite matrice de dispersion, et $\beta > 0$ le paramètre de forme. Dans (1), $c(\beta)$ est une constante de normalisation, g_β une certaine fonction, dite fonction génératrice de la loi, et $\delta = (x-\mu)^\dagger \Sigma^{-1}(x-\mu)$.

Par ailleurs, un mélange de lois à contours elliptiques est une loi de probabilité sur $\mathbb{R}^m$, avec une densité de probabilité de la forme

$$f(x|\boldsymbol{\theta}) = \sum_{k=1}^{K} w_k \, p(x|\theta_k) \tag{2}$$

où K est le nombre de composantes de mélange, et chaque composante de mélange $p(x|\theta_k)$ est de la forme (1). Les paramètres w_k sont les poids du mélange, et vérifient $w_k \in (0,1)$ avec $\sum_{k=1}^K w_k = 1$. Le paramètre "général" $\boldsymbol{\theta}$ est donné par $\boldsymbol{\theta} = (w_k, \theta_k \, ; k = 1, \ldots, K)$, où $\theta_k = (\mu_k, \Sigma_k, \beta_k)$.

Parmi les lois à contours elliptiques, nous avons souvent considéré les cas particuliers des lois gaussiennes généralisées multivariées, et des lois t de Student multivariées. Ces deux sous-familles de lois ont prouvé leur utilité en traitement des images, vision par ordinateur, traitement radar, et imagerie biomédicale [35, 81, 26, 50].

Également, les mélanges de lois gaussiennes généralisées multivariées, ou de lois T de Student multivariées, sont reconnues en tant que généralisations des mélanges de lois gaussiennes multivariées, aux données à queues lourdes ou corrompues par des valeurs aberrantes [30, 41]. Ces mélanges ont été appliqués aux traitements des images et des vidéos, et plus généralement en analyse des données [57, 76, 38].

Contexte de la thèse

Le problème de l'estimation des lois à contours elliptiques est un problème classique, pour lequel il existe déjà plusieurs méthodes de référence. En revanche, l'estimation des mélanges de lois à contours elliptiques est un problème plus récent dans la littérature, et bien plus difficile.

En règle générale, les méthodes existantes sont basées sur l'estimation par maximum de vraisemblance (la méthode des moments, applicable à l'estimation des lois gaussiennes généralisées multivariées, est une exception à cette règle [82]). Bien sûr, l'estimation par maximum de vraisemblance garantit la consistance (convergence des estimateurs vers les vrais paramètres) et l'efficacité asymptotique (la performance asymptotique optimale). Cependant, les estimateurs du maximum de vraisemblance, pour les lois à contours elliptiques et leurs mélanges, ne peuvent pas être calculés sous forme explicite. Plutôt, des méthodes itératives spéciales doivent être développées pour les obtenir.

Pour les lois à contours elliptiques, la méthode itérative la plus connue, pour calculer l'estimateur du maximum de vraisemblance, est la méthode du point fixe [59, 50, 84]. Dans [75], cette méthode est comparée à des méthodes d'optimisation riemannienne, comme le gradient conjugué ou BFGS riemannien.

Pour les mélanges de lois à contours elliptiques, l'approche standard pour calculer l'estimateur du maximum de vraisemblance reste la méthode espérance-maximisation (e.m.) [23, 37, 59]. Des méthodes plus sophistiquées n'ont été proposées que très récemment, comme la méthode d'optimisation proximale alternée de [38].

Dans le cadre de la thèse, nous avons commencé par recenser les méthodes existantes, pour réaliser l'estimation par maximum de vraisemblance des lois à contours elliptiques et de leurs mélanges. Nous avons classé ces méthodes en méthodes hors ligne/en ligne, et aussi en méthodes euclidiennes/riemanniennes.

Les méthodes hors ligne utilisent toutes les données disponibles, à chacune de leurs itérations. En revanche, les méthodes en ligne n'utilisent qu'une seule donnée, ou mini-batch, à chaque itération. Les méthodes hors ligne sont mieux adaptées pour les jeux de données de taille modérée, alors que les méthodes en ligne sont mieux adaptées aux jeux de données de haute dimension ou de grande taille.

Par ailleurs, la différence entre les méthodes euclidiennes et riemanniennes est la suivante. Les paramètres des lois à contours elliptiques et de leurs mélanges obéissent à certaines contraintes non linéaires (par exemple, les matrices de dispersion doivent être définies positives). Les méthodes euclidiennes imposent ces contraintes à travers des techniques *ad hoc* (par exemple, rajouter une constante positive à la matrice de dispersion, afin qu'elle reste définie positive [38]), alors que les méthodes riemanniennes font appel à la géométrie intrinsèque des paramètres.

En réalité, les lois à contours elliptiques sont mieux adaptées à la modélisation des jeux de données de taille modérée, alors que les mélanges de lois à contours elliptiques répondent mieux aux jeux de données de grande taille (puisqu'on peut augmenter le nombre de composantes de mélange). Il n'est donc pas surprenant que les méthodes existantes pour l'estimation des lois à contours elliptiques soient toutes des méthodes hors ligne. Pour l'estimation des mélanges de lois à contours elliptiques, il est beaucoup plus intéressant de développer des méthodes en ligne. Or, on remarque que de telles méthodes n'ont été proposées que très récemment [57, 38]. De plus, il n'existe aucune méthode en ligne riemannienne, dans la littérature à présent.

C'est dans ce contexte que nous avons introduit, à travers cette thèse, des méthodes d'estimation en ligne riemanniennes, pour les lois à contours elliptiques et leurs mélanges.

Contribution de la thèse

La contribution principale de cette thèse est d'avoir introduit une nouvelle famille de méthodes, appelées méthodes CIG (pour component-wise information gradient), pour l'estimation des lois à contours elliptiques et de leurs mélanges. Le cas des lois à contours elliptiques a été traité en premier, dans [90], et celui des mélanges un peu plus tard, dans [89].

Les méthodes CIG sont des méthodes d'estimation riemanniennes, basées sur une nouvelle géométrie pour les lois à contours elliptiques, également développée dans le cadre de la thèse. Cette géométrie est fondée sur une métrique riemannienne, que nous avons proposée, sous le nom de CIM (pour component-wise information metric).

La motivation pour l'introduction de la CIM, et ensuite des méthodes CIG, se trouve dans notre premier article de revue [88]. Cet article a étudié une méthode d'estimation en ligne riemannienne, appelée méthode du gradient d'information riemannien.

La méthode du gradient d'information riemannien est une extension au cadre riemannien d'une méthode classique assez connue, qui est la méthode du gradient naturel [4]. C'est une méthode en ligne riemannienne, facile à utiliser, et avec d'excellentes propriétés de convergence.

En effet, la méthode du gradient d'information riemannien ne demande aucun calibrage de la part de l'utilisateur (précisément, la sélection des pas d'optimisation se fait de manière automatique). En même temps, elle est rapidement convergente et asymptotiquement efficace.

Cela étant dit, cette méthode a un champ d'application limité, comme elle ne peut être appliquée que si la métrique d'information de Fisher est connue sous forme explicite. Par exemple, elle ne permet d'estimer les lois à contours elliptiques que dans le cas où le paramètre de forme est déjà connu (dans ce cas, la métrique d'information de Fisher a été donnée sous forme explicite dans [8]).

Dans [90], notre objectif était de trouver une méthode capable de remplacer la méthode du gradient d'information riemannien, pour l'estimation des lois à contours elliptiques "complètes", avec un paramètre de forme inconnu. Au lieu de chercher l'expression de la métrique d'information de Fisher, qui devenait de plus en plus compliquée, nous avons choisi de la remplacer par une autre métrique, plus simple et plus facile à calculer. C'est ainsi que nous avons introduit la CIM, qui est une approximation diagonale par blocs de la métrique d'information de Fisher.

Intuitivement, les méthodes CIG sont liées à la CIM de la même manière que la méthode du gradient d'information riemannien est liée à la métrique d'information de Fisher. Cela permet de retenir au moins une partie des propriétés de convergence de la méthode du gradient d'information riemannien, tout en élargissant son champ d'application aux lois à contours elliptiques avec un paramètre de forme inconnu.

La méthode CIG pour l'estimation des lois à contours elliptiques a deux versions, une version hors ligne et une version en ligne. La version en ligne est particulièrement intéressante, dans les applications aux jeux de données réels de haute dimension ou de grande taille. Par exemple, elle est capable de réaliser une transformation de couleurs, entre deux images qui font plus de 10^6 pixel, en 21 secondes, seulement. Pour cette même transformation, les méthodes existantes, dans l'état de l'art, mettent jusqu'à 2 heures, pour une performance comparable. En fait, les méthodes CIG en ligne ont même donné d'excellents résultats sur des images de haute définition, bien plus grandes.

Nous avons effectué des expériences numériques détaillées, pour comparer les méthodes CIG aux méthodes déjà existantes pour l'estimation des lois à contours elliptiques.

Ces expériences ont montré que la méthode CIG en ligne est similaire à la méthode des moments (de [82]), par sa faible complexité calculatoire et son temps de calcul réduit, mais aussi similaire à la méthode du point fixe (de [50, 59, 84]), grâce à sa performance supérieure. La méthode CIG en ligne réussit donc à réunir les meilleures qualités des méthodes existantes, à la fois en termes de temps de calcul et de performance.

Dans notre dernier article [89], nous avons étendu les méthodes CIG, de l'estimation des lois à contours elliptiques à l'estimation de leurs mélanges. Nous n'avons considéré que la méthode CIG en ligne, comme les lois de mélange sont habituellement utilisées pour modéliser des jeux de données plus compliqués, de grande taille. En revanche, nous avons encore développé deux versions de la méthode CIG en ligne, la première avec des pas d'optimisation décroissants, et la deuxième avec des pas d'optimisation adaptatifs. Nous avons appelé ces deux versions CIG-DS et CIG-AS (pour decreasing step-size et adaptive step-size, respectivement).

La méthode CIG-AS implémente une sélection adaptative des pas d'optimisation. De plus, on peut montrer qu'elle a un taux de convergence très rapide, puisqu'elle réalise ce qu'on appelle un taux de convergence linéaire (ce qui signifie que l'erreur d'optimisation décroît avec une vitesse exponentielle [2]). La méthode CIG-AS est très intéressante, d'un point de vue pratique aussi bien que théorique.

Cependant, dans les expériences avec des données simulées ou réelles, nous avons implémenté une méthode hybride, entre CIG-DS et CIG-AS. En effet, CIG-DS est préférable dans la phase préliminaire de l'optimisation, alors que CIG-AS est préférable dans la phase finale, à l'approche d'un point cible. De manière impressionnante, cette méthode hybride avait une meilleure performance que les méthodes existantes dans l'état de l'art, telles que la méthode du gradient stochastique euclidien et la méthode e.m. [37, 57]. De plus, cette méthode hybride avait un temps de calcul deux à trois fois plus faible que les méthodes de l'état de l'art.

En résumé, les méthodes CIG ont plusieurs avantages sur les méthodes existantes pour l'estimation des lois à contours elliptiques et de leurs mélanges.

- les méthodes CIG en ligne n'utilisent qu'une seule donnée ou mini-batch (de taille constante), pour chaque itération. Cela leur permet de prendre en compte des jeux de données de haute dimension ou de grande taille, avec un temps de calcul assez réduit. A l'inverse, la majorité des méthodes existantes sont des méthodes hors ligne (point fixe ou e.m. [50, 59]), qui doivent utiliser des ressources en temps et mémoire beaucoup plus importantes, pour traiter des jeux de données plus grands.

- la méthode CIG-AS implémente une sélection adaptative, complètement automatique, des pas d'optimisation. Cela permet d'éviter une phase de calibrage avec sélection manuelle des pas, habituellement longue et pénible. La majorité des méthodes d'estimation en ligne sont très sensibles aux pas d'optimisation, et doivent faire appel à une phase de calibrage (voir la discussion dans [80]).

- les méthodes CIG en ligne garantissent une convergence rapide et une précision élevée des estimateurs, même en comparaison avec des méthodes hors ligne. Les autres méthodes en ligne décrites dans l'état de l'art (notamment [38, 57]) restent difficiles à mettre en place lorsque les paramètres de forme des lois à contours elliptiques sont inconnus. Dans ce cas, ou bien elles manquent de précision, ou bien elles ont besoin d'utiliser des mini-batch d'une taille croissante.

Organisation de la thèse

La thèse contient sept chapitres, dont le contenu s'organise de la façon suivante.

- le chapitre 1 est un chapitre introductif. Il rappelle la définition des lois à contours elliptiques et de leurs mélanges, et présente de façon générale les différentes approches à leur estimation. Ensuite, il décrit la contribution et l'organisation de la thèse.

- le chapitre 2 contient une revue des travaux existants dans la littérature, sur la géométrie des lois à contours elliptiques, ainsi que sur l'estimation de ces lois et de leurs mélanges. Les sections 2.3 et 2.4 présentent les méthodes existantes pour l'estimation des lois à contours elliptiques et de leurs mélanges, respectivement. A la fin du chapitre, ces méthodes sont regroupées dans les tableaux 2.1 et 2.2, sous une forme synthétique.

- le chapitre 3 contient la contribution de la thèse à la géométrie des lois à contours elliptiques et de leurs mélanges : l'introduction d'une nouvelle métrique Riemannienne, appelée CIM (component-wise information metric), dans les sections 3.3 et 3.4. Ces deux sections donnent l'expression de la CIM, du gradient riemannien qui lui est associé (le CIG), et de la rétraction à utiliser avec ce gradient, pour l'estimation des lois à contours elliptiques et de leurs mélanges, respectivement. A la fin du chapitre, ces expressions sont réunies dans le tableau 3.5. Elles seront fondamentales pour les nouvelles méthodes d'estimation, introduites dans les chapitres 5 et 6.

- le chapitre 4 introduit la méthode du gradient d'information riemannien, basée sur notre premier article en revue [88]. Cette méthode est applicable au cas particulier des lois à contours elliptiques avec un paramètre de forme connu. Ses propriétés de convergence sont énoncées dans les Propositions 1 à 5. Celles-ci montrent que la méthode du gradient riemannien d'information converge rapidement, et qu'elle est asymptotiquement efficace, tout en ayant une sélection automatique des pas d'optimisation.

- le chapitre 5 introduit la méthode CIG, pour l'estimation des lois à contours elliptiques complètes, issue de notre article en revue [90]. La section 5.2 décrit la méthode CIG hors ligne (Algorithme 1), et donne des conditions suffisantes pour sa convergence, dans la proposition 6. La section 5.3 décrit la méthode CIG en ligne (Algorithme 2), ainsi que ces propriétés théoriques (taux de convergence, normalité asymptotique), dans les propositions 7 à 9.

- le chapitre 6 introduit la méthode CIG en ligne pour l'estimation des mélanges de lois à contours elliptiques, issue de notre article en revue [89]. La section 6.2 décrit la version à pas d'optimisation décroissants de cette méthode (Algorithme 3), et la section 6.3 décrit la version à pas d'optimisation adaptatifs. La convergence et le taux de convergence de ces deux versions sont donnés par les propositions 11 et 12.

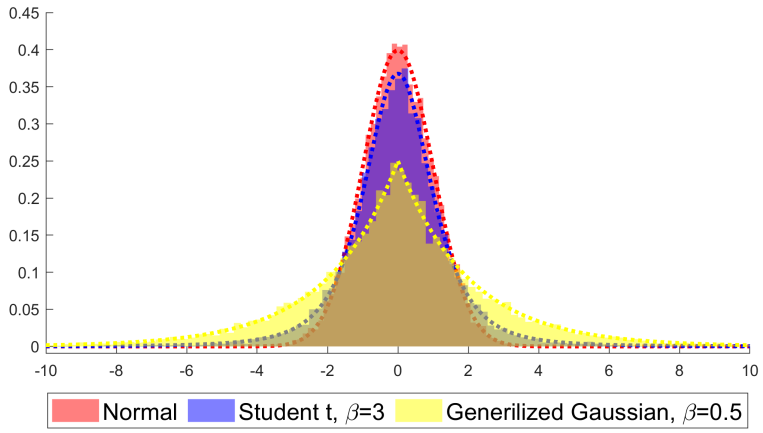
- le chapitre 7 met en évidence les propriétés des méthodes CIG, à travers des applications à des données simulées et réelles. Il compare aussi les méthodes CIG aux méthodes existantes dans la littérature, en termes de leur performance et de leur temps de calcul. Dans ce chapitre, la section 7.2 présente les applications aux données simulées, alors que la section 7.3 s'adresse aux données réelles.

Chapter 1

Introduction

1.1 ECD and their mixtures

Elliptically-contoured distributions (ECD) are a far-reaching extension of multivariate Gaussian distributions. They go by this name because, when an ECD has a probability density function, its contours (level surfaces) are multi-dimensional ellipsoids.



The common centre of these ellipsoids is determined by the location parameter (or expectation) $\mu \in \mathbb{R}^m$, while the squares of the principal axes are proportional to the eigenvalues of the scatter matrix $\Sigma \in \mathcal{P}_m$ (here, $\mathcal{P}_m$ denotes the space of $m \times m$ symmetric positive-definite matrices). An additional shape parameter $\beta \in \mathbb{R}_+$ (For some distributions it is also called degree of freedom or scale parameter) determines the factor for this proportionality.

Let X be an m-dimensional random vector following an ECD model. Its probability

density function (if it exists) takes on the form

$$p(x|\theta) = c(\beta)\frac{1}{\sqrt{\det(\Sigma)}}g_\beta(\delta) \tag{1.1}$$

where $\theta = (\mu, \Sigma, \beta)$, the constant $c(\beta)$ is a normalizing factor, and $\delta = (x-\mu)^\dagger\Sigma^{-1}(x-\mu)$. Here, $\dagger$ denotes the transpose of the matrices. The density generator function g_β depends on the specific sub-family of ECD distributions.

ECD were originally introduced in [46] and investigated in [17, 27, 31]. They inherit a lot of the nice properties of multivariate Gaussian distributions, and also make up for some of their deficiencies. Compared with multivariate Gaussian distribution, ECD are more flexible and powerful (e.g. they allow for heavy tails), and contain many widely-used sub-families of statistical distributions (elliptical Gamma, Pearson type II, elliptical multivariate logistic distributions, ...).

In terms of applications, the most popular sub-families of ECD are Multivariate Generalized Gaussian Distributions (MGGD), and Multivariate Student t-Distributions [34, 47, 69]. These are location-scale distributions, and are further parameterised by a shape parameter, or a degrees of freedom parameter. MGGD are used in image processing, as models for wavelet and curvelet coefficients, and as models for three-channel colour vectors, in image denoising, context-based image retrieval, image thresholding, texture classification, and image quality assessment [6, 12, 21, 35, 70, 81]. MGGD are also used in video coding and denoising, radar signal processing, and biomedical signal processing [24, 26]. Some applications of Student t-distributions are presented in [50], involving image denoising. In radar imaging, the Student t-distribution is related to the so-called $\mathcal{G}^0$ model within the family of spherically invariant random vectors (SIRVs), which is largely exploited in the context of SAR or PolSAR imaging, for tasks such as despeckling, classification, segmentation or detection [10, 20, 29, 32].

Mixtures of ECD generalise mixtures of multivariate Gaussians (we will use the abbreviation MECD). An MECD is a convex combination of a finite number of ECD. Its density function is given by

$$f(x|\boldsymbol{\theta}) = \sum_{k=1}^{K} w_k\, p(x|\theta_k) \tag{1.2}$$

Where K is the number of mixture components (or order of the mixture). The parameters w_k are called mixture weights and must satisfy $w_k \in (0,1)$ and $\sum_{k=1}^{K} w_k = 1$. The function $p(x|\theta_k)$ is the k-th component of the mixture, with its parameters $\theta_k = (\mu_k, \Sigma_k, \beta_k)$, given as in equation (1.1). We will use $\boldsymbol{\theta}$ to denote the ordered set of all the parameters of this mixture model $\boldsymbol{\theta} = (w_k, \mu_k, \Sigma_k, \beta_k; k = 1, \cdots, K)$.

MECD are capable of modelling general, real-world probability distributions. For example, they have been recognized as generalizations of Gaussian mixture models, for data with heavy tails or outliers [30, 41]. Mixtures of MGGD have been applied to object or action recognition in videos [57]. Moreover, mixtures of Laplace distributions have been applied to image denoising [76]. Mixtures of Multivariate Student T-distributions have been used for robust modelling, clustering and classification of data with outliers [5, 38, 52, 60]. Based on MECD model, a new clustering algorithm for PolSAR images segmentation is introduced in [66]

Because ECD and MECD have been successful in real-world signal and image processing applications, much attention has been devoted to developing effective methods for estimating their parameters. The estimation of ECD is an important problem that has

received extensive attention. On the other hand, the estimation of MECD is currently an emerging problem, which is significantly harder than the estimation of ECD.

1.2 The estimation problem

There already exist several well-established methods for ECD estimation, while a few methods for MECD estimation have been recently proposed in the literature. Practically all of these methods are based on maximum-likelihood estimation (MLE), but one notable exception is the method of moments (MM), which can sometimes be used for ECD estimation.

The method of moments was introduced for MGGD estimation, in [82]. As its name indicates, it is based on the idea of matching first, second, and fourth order moments of an unknown ECD with their empirical estimates.

The main advantage of this method is its low computational cost. However, it is much less accurate than maximum-likelihood estimation, and cannot be applied at all, to ECD which do not have higher-order moments (*e.g.*, higher order moments are infinite), such as Student T-distributions [33, 18].

Maximum-likelihood estimation focuses on maximising the log-likelihood function[1]. The maximum-likelihood estimate, based on a dataset $\mathcal{X} = \{x_1, \cdots, x_T\}$, drawn from an unknown ECD density $p(x|\theta^*)$, is a global maximiser of the log-likelihood function

$$L(\theta; \mathcal{X}) = \sum_{t=1}^{T} \ell_p(\theta; x_t) \tag{1.3}$$

where $\ell_p(\theta; x) = \log p(x|\theta)$ for the ECD density $p(x|\theta)$ in (1.1), and the maximum is taken over all $\theta = (\mu, \Sigma, \beta)$.

Similarly, if the dataset $\mathcal{X} = \{x_1, \cdots, x_T\}$ is drawn from an unknown MECD density $f(x|\boldsymbol{\theta}^*)$, then the maximum-likelihood estimate is a global maximiser of the log-likelihood function

$$L(\boldsymbol{\theta}; \mathcal{X}) = \sum_{t=1}^{T} \ell_f(\boldsymbol{\theta}; x_t) \tag{1.4}$$

where $\ell_f(\boldsymbol{\theta}; x) = \log f(x|\boldsymbol{\theta})$ for the MECD density $f(x|\boldsymbol{\theta})$ in (1.2), and the maximum is taken over all $\boldsymbol{\theta} = (w_k, \mu_k, \Sigma_k, \beta_k; k = 1, \cdots, K)$ — recall K is the number of mixture components, in (1.2).

The advantage of maximum-likelihood estimation is that the maximum-likelihood estimate is known to converge to the true parameter θ^* or $\boldsymbol{\theta}^*$, and to be asymptotically efficient (roughly speaking [44, 78], it enjoys the best possible asymptotic performance). On the other hand, it is not immediately clear how to compute maximum-likelihood estimates, and special methods must be developed for this purpose.

For ECD and MECD estimation, these methods can be divided into offline and online, and also into Euclidean and Riemannian methods. An iterative method for computing maximum-likelihood estimates uses available data to update (at each new iteration) an approximation $\theta^{(n)}$, and obtain a new and improved $\theta^{(n+1)}$.

The method is called offline or batch if it uses all of the available data for each update. It is called online or stochastic if it uses only one datapoint or one mini-batch of datapoints,

[1] In later chapters of this thesis, we will consider an alternative formulation, which focuses on minimising the Kullback-Leibler divergence, rather than maximising the log-likelihood.

for each update. Offline methods are suitable for dealing with small or moderate-sized datasets, while online methods are suitable for high-dimensional or large-scale datasets.

Furthermore, the distinction between Euclidean and Riemannian methods is the following. The parameters of ECD and MECD are subject to certain non-linear constraints (such as scatter matrices being positive-definite). Euclidean methods enforce these constraints using *ad hoc* techniques (such as adding a positive number to the scatter matrix, to make sure it is positive-definite [38]), while Riemannian methods appeal to the intrinsic Riemannian geometry of the parameters.

Usually, ECD are better at modelling moderate-sized datasets, while MECD are better with large-scale datasets. This is the reason why most currently existing methods for ECD estimation are offline methods, perhaps the most popular among them being the fixed-point method (FP) [59, 50, 84]. For now, existing methods for MECD estimation are mostly based on expectation-maximization (EM), which is an offline method [59, 37]. Very recently, a few works have considered online methods [57, 38]. To our knowledge, there still do not exist any general Riemannian online methods for MECD estimation.

Within this context, the present thesis will introduce new Riemannian and online estimation methods both for ECD and MECD, designed to overcome the major difficulties which we have observed with currently existing methods.

1.3 Contribution of this thesis

The contribution made in the present thesis is based on our three journal papers [88, 90, 89]. Above all, it consists in a new family of methods, for the estimation of ECD and MECD, called CIG methods. The abbreviation CIG stands for component-wise information gradient, and will be explained below.

The present section will discuss some of the main features of CIG methods. These features can be summarised as follows.

- the CIG methods include the CIG online estimation method. For each iteration, this only requires one datapoint or one mini-batch of datapoints. This makes it able to process high-dimensional or large-scale datasets, within short amounts of time.
- the CIG methods include the CIG adaptive step-size method. This implements a fully automatic selection of optimisation step-sizes (learning rates). This makes it easy to use, since it avoids the cumbersome task of manual selection of step-sizes.
- the CIG methods are able to guarantee fast rates of convergence, and high estimation accuracies, even in comparison to offline methods, which process the entire dataset at each iteration.

The importance of these features can be appreciated, in applications to real data. For example, in image editing, the CIG online method can be used to perform a colour transformation on a pair of images, with over 10^6 pixels, in only 21 seconds. For this same colour transformation, other state-of-the-art methods require two hours to do the same job, with a comparable performance. In fact, the CIG online method will be seen to produce excellent results, in applications with much larger, full HD images.

Our motivation, for introducing the CIG methods, is the fact that the Fisher information metric (FIM) of ECD and MECD models can be very difficult to compute. This metric does have a tractable, closed-form expression, but only in the case of ECD models with a known and fixed shape parameter.

In this particular case, the closed-form expression of the FIM was given in [8]. In our journal paper [88], we used it to introduce the Riemannian information gradient method. This is an online method, with automatic selection of step-sizes, which has, in a certain sense, the "best" convergence properties : fast rate of convergence and asymptotic efficiency.

The Riemannian information gradient method cannot be applied to complete ECD models (with unknown location and shape parameters), and cannot be applied to MECD models. This is because the FIM is not known, in a tractable closed form, for these models. In fact, for MGGD, there does exist an analytic expression of the FIM, but it is overly complicated [83].

In the journal paper [90], our aim was to somehow extend the Riemannian information gradient method to complete ECD. Instead of pursuing increasingly complicated expressions of the FIM, we decided to replace it with another metric, which we called the component-wise information metric (CIM). Concretely, the CIM is a block-diagonal approximation of the FIM, and can be computed in a tractable closed form, even for complete ECD.

Roughly speaking, the CIG method for ECD estimation is based on the CIM, in the same way that the Riemannian information gradient method is based on the FIM. The CIG method for ECD estimation has two versions, CIG offline and CIG online.

The CIG method shows comparable, and sometimes superior performance, to that of state-of-the-art methods, such as the MM (method of moments) and FP (fixed-point) methods [82, 59, 50]. However, the main advantage lies with its CIG online version, which requires significantly shorter amounts of time, in order to converge. Precisely, this CIG online version is similar to the more elementary MM method, for its short convergence time, and similar to the more sophisticated FP method, for its improved performance.

In the journal paper [89], we extended the CIG method, from the estimation of ECD models to the estimation of MECD models. We only considered the online version, because MECD are high-dimensional, and usually used to model more complex, large-scale datasets. However, we developed two sub-versions of the online version, one with decreasing step-sizes, and the other one with adaptive step-sizes, which we called the CIG-DS an CIG-AS methods, respectively.

The CIG-AS method implements an adaptive, fully automatic selection of step-sizes, and is theoretically shown to achieve a so-called linear rate of convergence (the term "linear convergence" means the optimisation error decreases exponentially fast [2]). This makes the CIG-AS method quite attractive, both from a practical and a theoretical perspective.

In computer experiments, with simulated or real data, we implemented a "hybrid method", combining CIG-DS and CIG-AS. The CIG-DS method is preferable in the early stages of optimisation, and the CIG-AS method in the later stages. Impressively, this hybrid method had better performance than state-of-the-art methods, such as Euclidean stochastic gradient, expectation-maximization with fixed-point method (EMFP) [57], and proximal alternating linearized minimization (inertial stochastic PALM) [38]. At the same time, the computation time it requires in order to converge is two to three times less than any other existing method.

The reason why CIG-DS and CIG-AS outperform existing methods for MECD estimation is because they were designed to overcome the most problematic issues, observed with these methods. Most important among these issues were the following,

- offline methods (such as EMFP [59, 37, 57]), when applied to large-scale datasets, require very extensive computational resources (time and memory), making them

quite impractical.

- existing online methods (described in [57, 38]) are difficult to implement when the shape parameters of ECD are unknown. In this case, they are either inefficient, or they require increasingly large mini-batch sizes.

- in addition, online methods involve the particularly delicate task of selecting step-sizes, which critically influence their rate of convergence [80].

The first of these issues is overcome thanks to the fact that CIG-DS and CIG-AS are online methods. The second issue is overcome thanks to the fact that CIG-DS and CIG-AS use the CIM to "pre-calibrate" stochastic gradients, which greatly improves their efficiency. Finally, the last issue is resolved thanks to the adaptive step-size selection, included in CIG-AS.

In addition to the above-mentioned contributions, based on the journal papers [88, 90, 89], the CIG online method for ECD estimation was applied to change detection in multivariate image time series, in our additional journal paper [13].

1.4 Organization of the thesis

In addition to the current introductory chapter, the present thesis includes Chapters 2 to 7, as well as Appendices A to C. The contents of these chapters and appendices may be summarised as follows.

- Chapter 2 provides a detailed discussion of existing literature on the geometry of ECD, and on the estimation of ECD and MECD. In particular, Sections 2.3 and 2.4 discuss state-of-the-art methods for ECD and MECD estimation, respectively. These methods are finally summarised in Tables 2.1 and 2.2.

- Chapter 3 lays out our contribution to the geometry of ECD and MECD : the introduction of the component-wise information metric (CIM). Sections 3.3 and 3.4 define the CIM for ECD and MECD, respectively. Each one of these sections gives expressions of the CIM, and of the associated information gradient and retraction maps. These expressions are summarised in Table 3.5. For Chapters 5 and 6, it should be kept in mind that the information gradient associated to the CIM is called the component-wise information gradient (CIG).

- Chapter 4 introduces the Riemannian information gradient method, based on our journal paper [88]. This method can be applied, in the particular case of ECD models with known shape parameter. Its main properties are stated in Propositions 1 to 5. These propositions show that the Riemannian information gradient method can achieve a fast rate of convergence, as well as asymptotic efficiency, with an automatic choice of step-sizes.

- Chapter 5 introduces the CIG method, for the estimation of complete ECD models (with unknown location and shape parameters). Section 5.2 presents the offline version of this method (Algorithm 1), and states its convergence in Proposition 6. Section 5.3 presents the online version of this method (Algorithm 2), and states its convergence, rate of convergence, and asymptotic normality, in Propositions 7 to 9. Section 5.4 studies the global convergence properties of the CIG method, for MGGD

and Student t-distribution models, with known shape parameters. These properties are summarised in Tables 5.1 and 5.2.

- Chapter 6 extends the CIG method, from the estimation of ECD models to the estimation of MECD models. Section 6.2 introduces the decreasing step-size version of the CIG method for MECD estimation (Algorithm 3), while Section 6.3 is concerned with the adaptive step-size version (Algorithm 5). The convergence and rate of convergence of these two methods is given by propositions 11 and 12, respectively.

- Chapter 7 illustrates, through applications to simulated and real data, the various properties of CIG methods, introduced in Chapters 5 and 6. It also compares CIG methods to existing state-of-the-art methods, in terms of performance and computation time. In this chapter, Section 7.2 presents computer experiments with simulated data, while Section 7.3 deals with applications to real data, such as image editing and texture segmentation.

Appendices A to C are devoted to the proofs of various propositions and corollaries, stated in Chapters 4 to 6.

Chapter 2

State of the art

2.1 Introduction

This chapter gives a detailed presentation of the existing estimation methods for ECD and MECD models. Some of these methods rely on the Riemannian geometry of ECD models, which will therefore also be presented in this chapter.

The problem of estimating an ECD model is typically formulated as a mathematical optimisation problem, such as maximising the log-likelihood function (MLE estimation), or minimising the Kullback-Leibler divergence. This optimisation problem is then addressed using some kind of iterative method, such as the fixed-point method or gradient descent method.

To formulate these iterative methods (*e.g.* to define and compute the gradient), or to study their convergence (*e.g.* to determine whether a fixed-point iteration is contractive or not), it is first of all necessary to equip the parameter space with a geometry. The most classical choice is just that of a Euclidean geometry.

However, the Euclidean geometry has its limitations in applications. First of all, the parameters of the ECD model are subject to non-linear constraints (such as the scatter matrix being positive-definite). Using the Euclidean geometry means these constraints should be checked and then re-enforced at each iteration, leading to additional computations and time consumption, or even to numerical instability. Second, in Euclidean geometry, the convexity of the cost function (the negative log-likelihood or the Kullback-

Leibler divergence) cannot be guaranteed, and global convergence of gradient descent is therefore difficult to obtain.

These issues are the main motivation for the introduction of the Riemannian geometry of ECD. Existing work on this geometry is presented in the following Section 2.2. The state of the art on estimation of ECD and MECD is then presented in Sections 2.3 (for ECD) and 2.4 (for MECD).

2.2 Geometry of ECD

Estimation methods for ECD and MECD can be classified into classical Euclidean methods and recent, more refined Riemannian methods. Riemannian methods take into account the intrinsic geometry of the parameter space of ECD and MECD models, and provide a better understanding of the structure of the optimisation problems involved in estimating these models.

For example, Riemannian methods automatically preserve fundamental non-linear constraints, such as positive-definiteness of the scatter matrix. Moreover, even when the negative log-likelihood or the Kullback-Leibler divergence is not convex, with respect to the scatter matrix, in the Euclidean sense, it is often convex in the Riemannian sense (geodesically convex) [85].

Thus, the introduction of a Riemannian metric leads to new Riemannian methods which have both practical and theoretical advantages. Mostly, existing works have focused on deriving a Riemannian metric for the scatter matrix, without considering the other parameters (location and shape parameters). These works consider two possibilities for the Riemannian metric of the scatter matrix.

The first, older metric is the affine-invariant metric introduce by [61]. The second is the Fisher information metric which was derived in [8]. In fact, the Fisher information metric is intimately related with the statistical properties of ECD models.

The Fisher information metric was first introduced by Rao [64]. It was later made popular by Amari [3]. In this thesis, we will be interested in the Fisher information metric, because it allows dramatic improvements in the estimation of ECD and MECD. In the context of maximum-likelihood estimation, the gradient of the log-likelihood function, computed with respect to this metric, is the *information gradient* (sometimes also called the natural gradient). Estimation methods based on the information gradient are easy to implement and have excellent convergence properties, including fast rate of convergence and asymptotic efficiency. These properties would be very hard, or even impossible, to obtain without using the information gradient.

To begin, let us recall the definition and properties of the affine-invariant metric. The Fisher information metric will be discussed in Subsection 2.2.2.

2.2.1 Affine-invariant metric

Recall that an ECD has three parameters, namely the location parameter $\mu \in \mathbb{R}^m$, the scatter matrix $\Sigma \in \mathcal{P}_m$, and the shape parameter $\beta \in (0, \infty)$. Out of these three, due to the non-linear geometry of $\mathcal{P}_m$, the estimation of Σ is the most difficult. Therefore, existing contributions have mostly focused on the Riemannian geometry of $\mathcal{P}_m$.

Geometrically, $\mathcal{P}_m$ is a cone, sitting inside the vector space of $m \times m$ symmetric matrices. An affine-invariant Riemannian metric on this cone was first introduced by [73]. For any point $\Sigma \in \mathcal{P}_m$ and any tangent vectors U, V in the tangent space $T_\Sigma \mathcal{P}_m$

(these will always be real $m \times m$ symmetric matrices), this affine-invariant metric is given by

$$\langle U, V \rangle_{\Sigma}^{\uparrow} = \mathrm{tr}(\Sigma^{-1} U \Sigma^{-1} V) \tag{2.1}$$

Here, instead of the conventional symbol "$\langle \cdot, \cdot \rangle$", we added a symbol "$\uparrow$" in order to distinguish it from the Fisher information metric that will appear in the following section. In the same way, the norm of this metric is denoted by $\|U\|_{\Sigma}^{\uparrow} = \sqrt{\langle U, U \rangle_{\Sigma}^{\uparrow}}$.

The geodesics of the affine-invariant metric (2.1) are given by the exponential map [61]

$$\mathrm{Exp}_{\Sigma}(U) = \Sigma^{\frac{1}{2}} \exp(\Sigma^{-\frac{1}{2}} U \Sigma^{-\frac{1}{2}}) \Sigma^{\frac{1}{2}} = \Sigma \exp(\Sigma^{-1} U) \tag{2.2}$$

The geodesic is the generalization of the straight line, which is defined by a starting point Σ and a starting velocity U. For the starting velocity U, there exists a unique geodesic curve $\gamma : [0, 1] \to \mathcal{P}_m, t \mapsto \gamma(t)$ such that $\gamma(0) = \Sigma$, $\gamma'(0) = U$ and $\mathrm{Exp}_{\Sigma}(U) = \gamma(1)$. The introduction of the exponential map allows us to compute this geodesic. That is to say, the non linear constraints are respected by using this exponential map. Its inverse is the Riemannian logarithm

$$\log_{\Sigma}(U) = \Sigma^{\frac{1}{2}} \log(\Sigma^{-\frac{1}{2}} U \Sigma^{-\frac{1}{2}}) \Sigma^{\frac{1}{2}} = \Sigma \log(\Sigma^{-1} U) \tag{2.3}$$

Another important concept is the Riemannian distance. For any pair of points Σ and $T = \mathrm{Exp}_{\Sigma}(U)$ in $\mathcal{P}_m$, we define the Riemannian distance $d(\Sigma, T)$ to be the infimum of the lengths of all admissible curves from Σ to T. The Riemannian distance corresponding to the metric (2.1) has the following expression,

$$\mathrm{d}_{\uparrow}^2(\Sigma, T) = \left(\|\log_{\Sigma}(T)\|_{\Sigma}^{\uparrow} \right)^2 = \mathrm{tr}\left[\log(\Sigma^{-1} T) \right]^2 \qquad \Sigma, T \in \mathcal{P}_m \tag{2.4}$$

Here, all matrix functions, such as matrix power, exponetial and logarithm, are symmetric matrix functions [40].

Now consider the log-likelihood function $\ell_p(\theta; x) = \log p(x|\theta)$ where $p(x|\theta)$ is given by (1.1). Using the first order of Tyler's development, the Riemannian gradient of $\ell(\Sigma)$ is defined by the unique vector field that satisfies $\langle \nabla_{\Sigma}^{\uparrow} \ell_p, U \rangle = \mathrm{d}\ell_p(\Sigma)[U]$, where d is the differential operator. The Riemannian gradient of $\ell_p(\theta; x)$ with respect to Σ, computed using the affine-invariant metric (2.1) has the following expression

$$\nabla_{\Sigma}^{\uparrow} \ell_p(\theta; x) = -\frac{1}{2}\Sigma - \frac{\partial h_{\beta}(\delta)}{\partial \delta}(x - \mu)(x - \mu)^{\dagger} \tag{2.5}$$

where $h_{\beta}(\delta) = \log g_{\beta}(\delta)$ for the density generator function g_{β} in Equation (1.1).

The above Riemannian metric, exponential map, distance, and Riemannian gradient, can be used to define the geodesic convexity (g-convexity, for short) with respect to Σ. We say that a function f is geodesically convex, if for any geodesic curve $\gamma(t)$, the composite function $f \circ \gamma(t)$ is a convex function in the classical sense.

Definition 1 *If Θ is a Riemannian manifold, a function $f : \Theta \mapsto \mathbb{R}$ is said to be geodesically α-strongly convex if for any $\theta, \theta' \in \Theta$,*

$$f(\theta') \geqslant f(\theta) + \left\langle \nabla_{\theta} f(\theta), \mathrm{Exp}_{\theta}^{-1}(\theta') \right\rangle_{\theta} + \frac{\alpha}{2} d^2(\theta, \theta')$$

where $\langle \cdot, \cdot \rangle_{\theta}$ denotes the Riemannian metric on Θ. The vector $\nabla_{\theta} f(\theta)$ means the Riemannian gradient of f, computed using this metric, Exp *and $d(\cdot, \cdot)$ mean the Riemannian exponential map and distance (again, with respect to the same Riemannian metric). For example, when $\theta = (\Sigma)$, the metric $\langle \cdot, \cdot \rangle_{\theta}$ can be the affine-invariant metric (2.1), with the gradient, exponential and distance given by (2.2), (2.4) and (2.5).*

Specific to the ECD distribution family, the geodesic convexity of the log-likelihood function $\ell_p(\theta; x)$, with respect to Σ, has been studied by several authors [74, 75, 87].

2.2.2 Fisher information metric

In addition to the affine-invariant metric (2.1), it is very interesting to consider the Fisher information metric of the ECD model, with respect to Σ, for fixed μ and β.

The Fisher information metric, as the affine-invariant metric, is defined by a scalar product $\langle U, V\rangle^* = \sum_{i,j} \mathcal{I}^*(\Sigma) U^i V^j$. The matrix $[\mathcal{I}^*(\Sigma)]_{i,j}$ is the Fisher information matrix of ECD model, which is a symmetric positive definite matrix and given by its definition

$$[\mathcal{I}^*(\Sigma)]_{i,j} = -\mathbb{E}\left[\frac{\partial^2 \ell(\Sigma)}{\partial \Sigma^i \partial \Sigma^j}\right]$$

The analytical expression is given as [8]

$$\langle U, V\rangle^*_\Sigma = I_{\Sigma,1}\mathrm{tr}(\Sigma^{-1}U\Sigma^{-1}V) + I_{\Sigma,2}\mathrm{tr}(\Sigma^{-1}U)\mathrm{tr}(\Sigma^{-1}V) \tag{2.6}$$

where $*$ is the notation for Fisher information metric, not the conjugate transpose. The information coefficients $I_{\Sigma,1}$ and $I_{\Sigma,2}$ are defined by

$$I_{\Sigma,1} = \frac{2\mathcal{A}}{m(m+2)} \qquad I_{\Sigma,2} = \frac{\mathcal{A}}{m(m+2)} - \frac{1}{4} \qquad \mathcal{A} = \mathbb{E}\left[\left(\frac{\partial h_\beta(\delta)}{\partial \delta}\delta\right)^2\right] \tag{2.7}$$

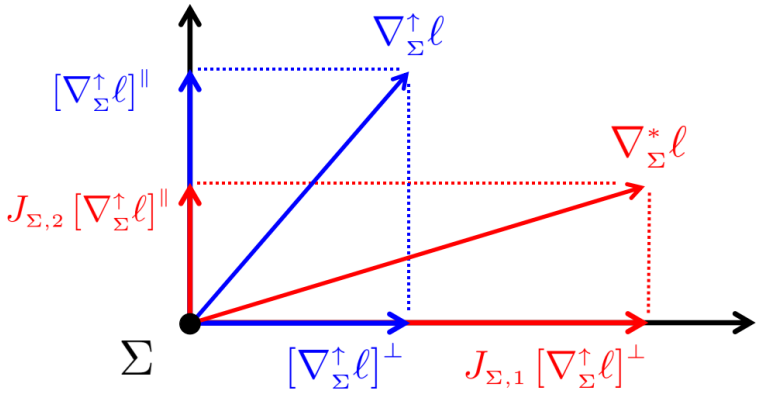
with $h_\beta(\delta) = \log g_\beta(\delta)$ as in equation (2.5).

This Fisher information metric shares the same exponential map and logarithmic map with the affine-invariant metric in (2.1). That is, both of these metrics have their exponential map given by (2.2) and logarithmic map given by (2.3). Similar to the affine-invariant metric, according to $d_*(\Sigma, T) = \|\mathrm{Log}_\Sigma(T)\|^*$, the Fisher information metric can also derive a Riemannien distance, that is analytically expressed as

$$d^2_*(\Sigma, T) = I_{\Sigma,1}\mathrm{tr}\left[\log(\Sigma^{-1}T)\right]^2 + I_{\Sigma,2}\mathrm{tr}^2\left[\log(\Sigma^{-1}T)\right] \qquad \Sigma, T \in \mathcal{P}_m \tag{2.8}$$

From definition (2.6) of the Fisher information metric, it is possible to compute the *information gradient* with respect to Σ. This turns out to be

$$\nabla^*_\Sigma \ell_p(\theta; x) \;=\; J_{\Sigma,1}\left[\nabla^\uparrow_\Sigma \ell_p(\theta; x)\right]^\perp + J_{\Sigma,2}\left[\nabla^\uparrow_\Sigma \ell_p(\theta; x)\right]^\parallel \tag{2.9}$$



Here, the vector $\nabla^\uparrow_\Sigma \ell_p(\theta; x)$ denotes the affine-invariant Riemannian gradient (2.5). Moreover, the coefficients $J_{\Sigma,1}$ and $J_{\Sigma,2}$ are given by

$$J_{\Sigma,1} = \frac{1}{I_{\Sigma,1}} \qquad J_{\Sigma,2} = \frac{1}{I_{\Sigma,1} + mI_{\Sigma,2}} \tag{2.10}$$

in terms of $I_{\Sigma,1}$ and $I_{\Sigma,2}$ defined in (2.7), and the symbols $\perp$ and $\|$ denote the following orthogonal decomposition of $\nabla^{\uparrow}_{\Sigma}\ell_p(\theta;x)$,

$$[\nabla^{\uparrow}_{\Sigma}\ell_p(\theta;x)]^{\|} = \frac{1}{m}\mathrm{tr}\left(\Sigma^{-1}\nabla^{\uparrow}_{\Sigma}\ell_p(\theta;x)\right)\Sigma \tag{2.11a}$$

$$[\nabla^{\uparrow}_{\Sigma}\ell_p(\theta;x)]^{\perp} = \nabla^{\uparrow}_{\Sigma}\ell_p(\theta;x) - [\nabla^{\uparrow}_{\Sigma}\ell_p(\theta;x)]^{\|} \tag{2.11b}$$

The gradient (2.9) is our first example of an information gradient. The information gradient is a central theme in this thesis, and we attempt to use it directly, for gradient descent, whenever possible. This is because the information gradient is "pre-calibrated", thanks to the presence of coefficients such as $J_{\Sigma,1}$ and $J_{\Sigma,2}$ in (2.9), which greatly simplify the task of choosing step-sizes.

2.3 Estimation of ECD

Typically, single ECD models (that is, one ECD rather than a mixture) are not used for modeling large-scale or high-dimensional datasets. Therefore, existing approaches to the estimation of ECD have focused on offline methods, where there is a finite number of datapoints. For a given dataset $\mathcal{X} = \{x_1, \cdots, x_T\}$, the objective function of MLE is

$$L(\theta; \mathcal{X}) = \sum_{t=1}^{T} \ell_p(\theta; x_t) = T \log c(\beta) - \frac{T}{2} \log\det(\Sigma) + \sum_{t=1}^{T} h_\beta(\delta_t) \tag{2.12}$$

where $\delta_t = (x_t - \mu)^\dagger \Sigma^{-1}(x_t - \mu)$ and $h_\beta(\delta) = \log g_\beta(\delta)$, in the notation of (1.1). The methods of maximizing this objective function can be classified into two categories, the first contains Euclidean (or classical) methods, and the second contains the methods based on Riemannian geometry.

2.3.1 Euclidean methods

In the context of Euclidean geometry, if the dataset is supposed to be centred, the MLE of (Σ, β) for MGGD was studied in [59, 84]. Also for MGGD, under the framework of block majorization-minimization, the MLE of all the three parameters (μ, Σ, β) is considered in [84]. For Student t-distributions, with the degree of freedom parameter being fixed, the MLE of (μ, Σ) is given in [50], according to a new parameterization of the couple (μ, Σ). Here is a more detailed description.

The estimation of Σ: In all the three works just mentioned ([59, 50, 84]), the fixed-point (FP) equation of Σ is obtained by differentiating the log-likelihood $L(\theta; \mathcal{X})$ with respect to Σ. This follows from

$$\frac{\partial L(\theta; \mathcal{X})}{\partial \Sigma} = -\frac{1}{2}\Sigma^{-1} - \frac{1}{T}\sum_{t=1}^{T} \frac{\partial h_\beta(\delta_t)}{\partial \delta_t} \Sigma^{-1}(x_t - \mu)(x_t - \mu)^\dagger \Sigma^{-1}$$

Setting this derivative to zero results in the FP (Fixed-Point method) equation

$$\Sigma = F(\Sigma) \text{ where } F(\Sigma) = -\frac{2}{T}\sum_{t=1}^{T} \frac{\partial h_\beta(\delta_t)}{\partial \delta_t}(x_t - \mu)(x_t - \mu)^\dagger \tag{2.13}$$

The sequence generated by the fixed-point iteration $\Sigma^{(n+1)} = F(\Sigma^{(n)})$ converges to a stationary point of $L(\Sigma)$. Note that, in [59], the scatter matrix Σ is normalized according to $\mathrm{tr}(\Sigma) = m$ and $\Sigma = mM$. Also in this same work [59], for MGGD models, the existence and uniqueness of the estimator of Σ is proved, whenever $\beta \in (0, 1)$.

The estimation of (μ, Σ): When both μ and Σ are unknown, the MLE can be achieved in two ways. The first one is implemented according to a new parameterization

$$\mathcal{S} = \begin{bmatrix} \Sigma + \mu\mu^\dagger & \mu \\ \mu^\dagger & 1 \end{bmatrix} \quad \text{therefore} \quad \mathcal{S} \in \mathcal{P}_{m+1} \tag{2.14}$$

If the new random vector y is given by $y^\dagger = (x^\dagger, 1)$, then the log-likelihood function can be reformulated as

$$\tilde{L}(\mathcal{S}) = -\frac{T}{2}\log\det(\mathcal{S}) - \sum_{t=1}^{T} \tilde{h}_\beta(\delta_{y_t}) \qquad \delta_{y_t} = y_t^\dagger \mathcal{S}^{-1} y_t \tag{2.15}$$

where the function $\tilde{h}_\beta$ is proportional to the original h_β. Therefore, the FP equation for $\mathcal{S}$ can be obtained by replacing Σ with $\mathcal{S}$ and then h_β with $\tilde{h}_\beta$ in (2.13). In [50], for Student t-distributions, the maximization of this new function $\tilde{L}(\mathcal{S})$ was proven to be equivalent to the maximization of $L(\mu, \Sigma)$. In addition, the existence and uniqueness of the MLE were also proved in [50].

Another idea is to find the MLE of μ and Σ through an alternating iteration. This idea is realized, in the case of MGGD, in [84]. The derivative of L with respect to μ is (recall that, here, $\theta = (\mu, \Sigma)$)

$$\frac{\partial L(\theta)}{\partial \mu} = -2 \sum_{t=1}^{T} \frac{\partial h_\beta(\delta_t)}{\partial \delta_t} \Sigma^{-1}(x_t - \mu)$$

its FP equation is derived by setting $\frac{\partial L(\theta)}{\partial \mu} = 0$

$$\mu = \frac{\sum_{t=1}^{T} \frac{\partial h_\beta(\delta_t)}{\partial \delta_t} x_t}{\sum_{t=1}^{T} \frac{\partial h_\beta(\delta_t)}{\partial \delta_t}} \tag{2.16}$$

In [84], equations (2.13) and (2.16) are used in an alternating manner to iteratively update estimates of μ and Σ, leading to the MLE of (μ, Σ). Global convergence of this method is guaranteed under the block majorization-minimization framework (BMM framework) [65].

The estimation of (μ, Σ, β): For the MGGD model, the estimation of the shape parameter β was achieved by solving a nonlinear equation [59, 84]. Let the derivative of L with respect to β equal to 0

$$\frac{\partial L(\theta)}{\partial \beta} = T \frac{\partial \log c(\beta)}{\partial \beta} + \sum_{t=1}^{T} \frac{\partial h_\beta(\delta_t)}{\partial \beta} = 0 \tag{2.17}$$

The update formula for β is then

$$\beta = \text{ solution of equation (2.17)} \tag{2.18}$$

The Newton-Raphson method is employed to solve equation (2.17) in [59]. In [84], the FP equations (2.13), (2.16) and (2.18) are applied in an alternating manner to find MLE estimates of the complete ECD model. Note that the negative log-likelihood function $-L(\beta; \mu^*, \Sigma^*)$ is proved to be strongly convex for any fixed $\mu = \mu^*$ and $\Sigma = \Sigma^*$. Therefore, under the framework of BMM, this alternating procedure converges to a stationary point of the log-likelihood $L(\theta; \mathcal{X})$ [84].

2.3.2 Riemannian methods

In the context of Riemannian geometry, all currently existing methods focus only on the MLE of the scatter matrix Σ, with the other parameters considered to be fixed [14, 74, 75, 87]. The gradient of the MLE objective function (2.12), with respect to the affine-invariant metric (2.1), can be found from (2.5)

$$\nabla^{\uparrow}_{\Sigma} L(\theta) = -\frac{T}{2}\Sigma - \sum_{t=1}^{T} \frac{\partial h_\beta(\delta_t)}{\partial \delta_t} (x_t - \mu)(x_t - \mu)^\dagger \tag{2.19}$$

Setting this gradient to zero, the same equation as (2.13) can be obtained. This FP equation is used for MLE of MGGD in [87], and of general ECD in [74]. For MGGD, the global g-convexity of the log-likelihood function is also studied in [87]. When the shape parameter β is fixed and $\beta \in [\frac{1}{2}, \infty)$, the log-likelihood function is globally geodesically convex, in the sense of Definition 1. Note that this is a different range of β than the one found in [59] using an Euclidean method.

A Riemannian-averaged fixed-point method was introduced in [14], which overcame the problem of instability of the FP iteration for larger values of the shape parameter. Precisely, this method implements successive Riemannian averages of fixed-point iterates. Recall the definition of the Riemannian average of $\Sigma, T \in \mathcal{P}_m$. For $t \in (0,1)$, the Riemannian average with ratio t of Σ and T is

$$\Sigma \#_t T = \Sigma^{\frac{1}{2}} (\Sigma^{-\frac{1}{2}} T \Sigma^{-\frac{1}{2}})^t \Sigma^{\frac{1}{2}}$$

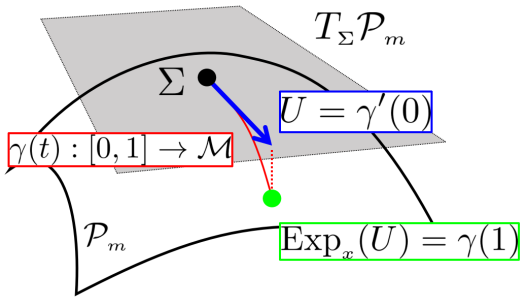
The Riemannian-averaged algorithm is defined as follows. When $\Sigma^{(n)}$ is given, instead of defining $\Sigma^{(n+1)}$ by $F(\Sigma^{(n)})$ according to (2.13), let

$$\Sigma^{(n+1)} = \Sigma^{(n)} \#_{t_n} F(\Sigma^{(n)}), \qquad t_n = \frac{1}{n+1}$$

Thanks to the stability of the Riemannian average, this method can give an efficient estimate of Σ for any value of the shape parameter β.

In addition to the fixed-point method, many Riemannian line-search methods are studied in [75]. The line-search optimization methods on Riemannian manifolds are all based on the following general iterative formula

$$\Sigma^{(n+1)} = \mathrm{Exp}_{\Sigma^{(n)}}(\eta^{(n)} U(\Sigma^{(n)}, \mathcal{X})) \tag{2.20}$$



where $\eta^{(n)} > 0$ is a step-size and $U(\Sigma^{(n)}, \mathcal{X})$ denotes a vector in $T_{\Sigma^{(n)}} \mathcal{P}_m$, which refers to the search direction (for example, $U(\Sigma^{(n)}, \mathcal{X}) = \mathrm{grad}_\Sigma L(\theta)$, defined in (2.19)).

With this general iterative formula, the most common line-search optimization methods, such as steepest descent, BFGS, and conjugate gradient methods, are compared with the FP methods, in [75]. The conclusion is that the Riemannian line-search methods and the FP method each have their own advantages. Although the Riemannian line-search methods have excellent performance, the FP algorithm is still competitive in many scenarios.

2.4 Estimation of MECD

Mixtures of ECD (MECD) are quite useful in fitting more complicated, large-scale datasets, of possibly higher dimension. For this reason, most existing methods for the estimation of MECD pay closer attention to online algorithms, which are time-saving and use up less memory. As in section 2.3, existing methods are classified into Euclidean and Riemannian.

2.4.1 Euclidean methods

In the Euclidean framework, the estimation of mixture models is classically addressed using Expectation-Maximization methods (EM) [23]. In these methods, a latent variable, $(Z_t^k; k = 1, \ldots, K)$ is introduced to indicate the membership of each datapoint x_t (then, $Z_t^k = 1$ if x_t belongs to the kth mixture component, and $Z_t^k = 0$ otherwise). The E (expectation) step computes the marginal log-likelihood of observed data. The M (maximization) step maximizes this marginal log-likelihood. These two steps are repeated until convergence.

To state this in a precise way, recall the MECD density (1.2) and its parameters $\boldsymbol{\theta} = (w_k, \mu_k, \Sigma_k, \beta_k; k = 1, \cdots, K)$. In the Expectation step, using the current estimates of the parameters

$$\boldsymbol{\theta}^{(n)} = (w_k^{(n)}, \mu_k^{(n)}, \Sigma_k^{(n)}, \beta_k^{(n)}; k = 1, \cdots, K)$$

the conditional expectation of the Z_t^k is determined by Bayes theorem, which gives "membership probabilities"

$$o_k(x_t|\boldsymbol{\theta}^{(n)}) = \frac{w_k^{(n)} p(x_t|\theta_k^{(n)})}{\sum_{k=1}^K w_k^{(n)} p(x_t|\theta_k^{(n)})} \tag{2.21}$$

Then, the function $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(n)})$ is defined to be the conditional expectation of the complete log-likelihood function

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(n)}) = \sum_{k=1}^K \sum_{t=1}^T o_k(x_t|\boldsymbol{\theta}^{(n)}) \log(w_k) + \sum_{k=1}^K \sum_{t=1}^T o_k(x_t|\boldsymbol{\theta}^{(n)}) \log p(x_t|\theta_k)$$

In the Maximization step, a Lagrange multiplier is added, for the weights w, in order to guarantee $\sum_{k=1}^K w_k = 1$. After differentiating the function $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(n)})$ with respect to w, the update formula for w is found to be

$$w_k^{(n+1)} = \frac{1}{T} \sum_{t=1}^T o_k(x_t|\boldsymbol{\theta}^{(n)})$$

In [57], the update formula for the other parameters $(\mu_k^{(n)}, \Sigma_k^{(n)}, \beta_k^{(n)})$, is performed by repeating the same operation as in equations (2.13), (2.16) and (2.18), for each sub-distribution (*i.e.* each mixture component). The resulting method is called the EM with fixed point method (EM-FP). In the same work [57], it was implemented for mixtures of MGGD, and then compared to the Euclidean stochastic gradient method.

For this Euclidean stochastic gradient method, the well-known logit parameterization is applied to ensure the constraints $\sum_{k=1}^K w_k = 1$. This parameterization is given by

$$\begin{aligned} w_k &= \frac{\exp(\pi_k)}{1 + \sum_{j=1}^{K-1} \exp(\pi_j)}, \text{ for } k \in \{1, \cdots, K-1\} \\ \text{and } w_K &= \frac{1}{1 + \sum_{j=1}^{K-1} \exp(\pi_j)} \end{aligned} \tag{2.22}$$

The logit parameterization yields the reparameterized log-likelihood function of the mixture model

$$\ell_f(\boldsymbol{\theta}; x) = \log \left[\sum_{k=1}^{K-1} \frac{\exp(\pi_k)}{1 + \sum_{j=1}^{K-1} \exp(\pi_j)} p(x|\theta_k) + \frac{1}{1 + \sum_{j=1}^{K-1} \exp(\pi_j)} p(x|\theta_K) \right]$$

The Euclidean gradient with respect to the parameters π_k is given by

$$\text{grad}_{\pi_k} \ell_f(\boldsymbol{\theta}; x) = o_k - \frac{\exp(\pi_k)}{1 + \sum_{j=1}^{K-1} \exp(\pi_j)}$$

$$\text{with } o_k = \frac{\exp(\pi_k)}{1 + \sum_{j=1}^{K-1} \exp(\pi_j)} \frac{p(x|\theta_k)}{f(x|\boldsymbol{\theta})}$$

$$f(x|\boldsymbol{\theta}) = \sum_{k=1}^{K-1} \frac{\exp(\pi_k)}{1 + \sum_{j=1}^{K-1} \exp(\pi_j)} p(x|\theta_k) + \frac{1}{1 + \sum_{j=1}^{K-1} \exp(\pi_j)} p(x|\theta_K)$$

The stochastic gradient update formula for the π_k is then

$$\pi_k^{(n+1)} = \pi_k^{(n)} + \eta^{(n+1)} \text{grad}_{\pi_k} \ell_f(\boldsymbol{\theta}^{(n)}; x_{n+1}) \tag{2.23}$$

Where the coefficient $\eta^{(n+1)}$ denotes a step-size, and the vector x_{n+1} is a new data point (it can also be replaced by a mini-batch). From (2.23), the new weights $w_k^{(n+1)}$ can be obtained according to (2.22).

The gradients with respect to the other parameters are

$$\text{grad}_{\mu_k} \ell_f(\boldsymbol{\theta}; x) = -2 o_k \frac{\partial h_{\beta_k}(\delta_k)}{\partial \delta_k} \Sigma_k^{-1} (x - \mu_k) \tag{2.24a}$$

$$\text{grad}_{\Sigma_k} \ell_f(\boldsymbol{\theta}; x) = -o_k \left[\frac{1}{2} \Sigma_k + \frac{\partial h_{\beta_k}(\delta_k)}{\partial \delta_k} (x - \mu_k)(x - \mu_k)^\dagger \right] \tag{2.24b}$$

$$\text{grad}_{\beta_k} \ell_f(\boldsymbol{\theta}; x) = o_k \left(\frac{\partial \log c(\beta_k)}{\partial \beta_k} + \frac{\partial h_{\beta_k}(\delta_k)}{\partial \beta_k} \right) \tag{2.24c}$$

where $h_{\beta_k}(\delta_k) = \log g_{\beta_k}(\delta_k)$ as in (1.1). Then, the Euclidean stochastic gradient updates for $\theta_k = (\mu_k, \Sigma_k, \beta_k)$ are

$$\mu_k^{(n+1)} = \mu_k^{(n)} + \eta^{(n+1)} \text{grad}_{\mu_k} \ell_f(\boldsymbol{\theta}^{(n)}; x_{n+1}) \tag{2.25a}$$

$$\Sigma_k^{(n+1)} = \Sigma_k^{(n)} + \eta^{(n+1)} \text{grad}_{\Sigma_k} \ell_f(\boldsymbol{\theta}^{(n)}; x_{n+1}) \tag{2.25b}$$

$$\beta_k^{(n+1)} = \beta_k^{(n)} + \eta^{(n+1)} \text{grad}_{\beta_k} \ell_f(\boldsymbol{\theta}^{(n)}; x_{n+1}) \tag{2.25c}$$

In [57], the gradients (2.24) are used in the standard formulation of a stochastic gradient descent, (2.25). However, in [38], they are used in the framework of an online proximal alternating linearized minimisation method (PALM), applied to the estimation of mixtures of Student T-distributions. In this work, the cost function is reshaped by adding the following surjective mappings to keep $\beta_k \in (0, \infty)$ and $\Sigma_k \in \mathcal{P}_m$ after each iteration.

$$\varphi_{\beta_k}(\beta_k) = \beta_k^2 + \epsilon \qquad \varphi_{\Sigma_k}(\Sigma_k) = \Sigma_k^\dagger \Sigma_k + \epsilon I_m \qquad \text{with } \epsilon > 0$$

In addition, an inertial gradient is applied to this PALM method to accelerate the iterative process, with a hyper parameter $\rho_k \in [0, 1]$, for each sub-component θ_k of $\boldsymbol{\theta}$

$$\tilde{\theta}_k^{(n)} = \theta_k^{(n)} + \rho_k (\theta_k^{(n)} - \theta_k^{(n)})$$
$$\theta_k^{(n+1)} = \tilde{\theta}_k^{(n)} + \eta^{(n)} \text{grad}_{\theta_k} \ell_f(\boldsymbol{\theta}; x)$$

2.4.2 Riemannian methods

In the Riemannian context, the MLE of a mixture of multivariate Gaussian distributions was studied in [42]. This work proposed an offline batch gradient method, and also an online mini-batch gradient method. The expectation μ_k and covariance matrix Σ_k of Gaussian model are integrated into a single parameter according to (2.14). The geometry of the new integrated matrix $\mathcal{S}_k$ is considered based on the affine-invariant gradients (2.5) and the exponetial map (2.2). The affine-invariant gradient of the reformulated $\tilde{\ell}_f(\boldsymbol{\theta}; x)$ with respect to $\mathcal{S}_k$ is

$$\nabla^{\uparrow}_{\mathcal{S}_k} \tilde{\ell}_f(\boldsymbol{\theta}; x) = -o_k \left[\frac{1}{2} \mathcal{S}_k + \frac{\partial \tilde{h}_{\beta_k}(\delta_k(y))}{\partial \delta_k(y)} y y^{\dagger} \right] \tag{2.26}$$

where $y^{\dagger} = (x, 1)^{\dagger}$, and $\delta_k(y) = y^{\dagger} \mathcal{S}_k^{-1} y$. The o_k are the membership probabilities, given in (2.21).

Moreover, the weights of the Gaussian mixture model have the same reformulation as in the in Euclidean method (2.22). Then, an alternating iteration scheme is adopted between the weights and the new parameter $(\mathcal{S}_k)_{1 \leqslant k \leqslant K}$. The resulting Riemannian online method (stochastic gradient method) has the following general form.

1. Update $\pi_k^{(n)}$ according to (2.23)
2. Reparametrize $\mathcal{S}_k^{(n)}$ according to (2.14) with current $(\mu_k^{(n)}, \Sigma_k^{(n)})$
3. Compute the Riemannian gradient $\nabla^{\uparrow}_{\mathcal{S}_k} \tilde{\ell}_f(\boldsymbol{\theta}^{(n)}; x_{n+1})$ using (2.26)
4. Update $\mathcal{S}_k^{(n)}$ according to the exponential map (2.2)

$$\mathcal{S}_k^{(n+1)} = \mathrm{Exp}_{\mathcal{S}_k^{(n)}} \left(\eta^{(n+1)} \nabla^{\uparrow}_{\mathcal{S}_k} \tilde{\ell}_f(\boldsymbol{\theta}^{(n)}; x_{n+1}) \right)$$

   Possibly, the new datapoint x_{n+1} can be replaced with a mini-batch.

5. Decompose $\mathcal{S}_k^{(n+1)} \to (\mu_k^{(n+1)}, \Sigma_k^{(n+1)})$

In the experiments of [42], the Riemannian gradient method, offline or stochastic with mini-batch, is compared with other standard methods, such as EM-FP, conjugate gradient and BFGS. The Riemannian stochastic gradient method shows remarkable convergence behavior, making it a potential candidate for large scale mixture learning.

In addition to mixtures of multivariate Gaussians, an offline estimation method for MECD models with a fixed shape parameter was proposed in [51]. This method is very similar to [42], also employing the reparametrization (2.14). Then, it implements a Riemannian conjugate gradient rule, into the alternating iterative scheme of the form just described.

2.5 Conclusion

We summarize the existing estimation methods of ECD and MECD, reviewed in Sections 2.3 and 2.4, in the following two tables.

ECD	
Euclidean	Riemannian
Pascal, Frédéric, et al. [59] "Parameter estimation for **MGGD**s." 2013 Method: **FP** Unknown parameters: (Σ, β) Other results: MLE of Σ exists and is unique for fixed β in $(0, 1)$	Boukouvalas, Zois, et al. [14] "A new Riemannian-averaged FP algorithm for **MGGD** parameter estimation." 2015 Method: **Riemannian-averaged FP** Unknown parameters: (Σ) Other results: More stable than FP
Laus, Friederike, and Gabriele Steidl. [50] "Multivariate myriad filters based on parameter estimation of **Student t** distributions." 2019 Method: **Alternate FP** Unknown parameters: (μ, Σ) Other results: MLE of (μ, Σ) exists and is unique	Zhang, Teng, Ami Wiesel, and Maria Sabrina Greco. [87] "**MGGD**: Convexity and graphical models." 2013 Method: **FP** Unknown parameters: (Σ) Other results: MLE of Σ exists and is unique for fixed β in $(\frac{1}{2}, \infty)$ Wiesel Ami [85]. "Geodesic convexity and covariance estimation." 2012
Wang, Bin, et al. [84] "Globally convergent algorithms for learning **MGGD**." 2021 Method: **Alternate FP** Unknown parameters: (μ, Σ, β) Other results: log-likelihood function is marginally convex with respect to β	Sra, Suvrit, and Reshad Hosseini. [74] "Geometric optimisation on positive -definite matrices for **ECD**s." 2013 Method: **FP** Unknown parameters: (Σ)
	Sra, Suvrit, and Reshad Hosseini. [75] "Conic geometric optimization on the manifold of positive-definite matrices." 2015 Method: **FP**, **steepest gradient**, **BFGS**, **conjugate gradient** Unknown parameters: (Σ)
Note that all the methods above are offline.	

Table 2.1: State of the art: existing methods for ECD estimation

In the following chapter, we will introduce a new geometry for ECD and MECD models. This will be based on the original concept of component-wise information metric (CIM). Later on, in Chapters 5 and 6, the CIM will be used to define a new online estimation method, which we have called the component-wise information gradient method (CIG). This new method is one of the main contributions of the present thesis. It will be shown to have several significant advantages, in comparison with existing estimation methods, summarized here in Tables 2.1 and 2.2.

MECD	
Euclidean	Riemannian
Najar, Fatma, et al. [57] "Online recognition via a finite mixture of **MGGD**." 2020 Method: **EM-FP**, **SGD** Unknown parameters: (w, μ, Σ, β)	Hosseini, Reshad, and Suvrit Sra. [42] "An alternative to EM for **Gaussian** mixture models: batch and stochastic Riemannian optimization." 2020 Method:**EM-FP**,**steepest gradient**, **BFGS**,**conjugate gradient** Unknown Parameters: (w, μ, Σ)
Hertrich, Johannes, and Gabriele Steidl. [38] "Inertial Stochastic PALM and its Application for Learning **Student T** Mixture Models." 2020 Method: **Inertial PALM with SGD** Unknown parameters: (w, μ, Σ, β)	Li, Shengxi, Zeyang Yu, and Danilo Mandic. [51] "A universal framework for learning the **elliptical** mixture model." 2020 Method: **conjugate gradient** Unknown parameters: (w, μ, Σ)
Roizman Violeta, Matthieu Jonckheere, and Frédéric Pascal [67]. "A flexible EM -like clustering algorithm for noisy data." 2019	

Table 2.2: State of the art: existing methods for MECD estimation

Chapter 3

The component-wise information metric

3.1 Introduction

Our contribution to the geometry of ECD and MECD lies in the introduction of the component-wise information metric (CIM). Our objective is to propose a new online estimation algorithm for ECD and their mixtures (that is, MECD). For ECD, if the shape parameter is known, efficient online estimation can be carried out using the Fisher information metric. However, if the shape parameter is unknown, the Fisher information metric does not have a closed-form expression. We specifically introduce the CIM as a computationally advantageous alternative to the Fisher information metric (FIM), in this case, and also in the more difficult case of MECD.

The aim of an online estimation algorithm is to find the true parameter θ^*. Here, this is formulated as an optimisation problem on a Riemannian manifold, which is the parameter space Θ of an ECD, or $\boldsymbol{\Theta}$ of an MECD model. This is the problem of minimising the Kullback-Leibler (or KL) divergence. For ECD models, the KL divergence is given by

$$D(\theta) = \int p(x|\theta^*) \log\left(\frac{p(x|\theta^*)}{p(x|\theta)}\right) \mathrm{d}x = \mathbb{E}_{\theta^*}\left[\log p(x|\theta^*)\right] - \mathbb{E}_{\theta^*}\left[\log p(x|\theta)\right] \tag{3.1}$$

for $\theta \in \Theta$, where $p(x|\theta)$ is given by (1.1) and θ^* is supposed to be the true parameter of the ECD model. Note that θ^* is the unique global minimum of $D(\theta)$. Another way of formulating (3.1) is to say that any online estimation algorithm searches for the minimum

$$\arg\min_{\theta\in\Theta} -\mathbb{E}_{\theta^*}\left[\ell_p(\theta;x)\right] \qquad \text{with } \ell_p(\theta;x) = \log p(x|\theta) \tag{3.2}$$

For MECD, the density function p and parameter θ from (1.1) are replaced by the density f and the parameter $\boldsymbol{\theta}$ from (1.2), respectively. The KL divergence is given by

$$D(\boldsymbol{\theta}) = \int f(x|\boldsymbol{\theta}^*) \log\left(\frac{f(x|\boldsymbol{\theta}^*)}{f(x|\boldsymbol{\theta})}\right) \mathrm{d}x = \mathbb{E}_{\boldsymbol{\theta}^*}\left[\log f(x|\boldsymbol{\theta}^*)\right] - \mathbb{E}_{\boldsymbol{\theta}^*}\left[\log f(x|\boldsymbol{\theta})\right] \tag{3.3}$$

for $\boldsymbol{\theta} \in \boldsymbol{\Theta}$. Then, online algorithms aim to search for the minimum

$$\arg\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} -\mathbb{E}_{\boldsymbol{\theta}^*}\left[\ell_f(\boldsymbol{\theta}; x)\right] \qquad \text{with } \ell_f(\boldsymbol{\theta}; x) = \log f(x|\boldsymbol{\theta}) \tag{3.4}$$

Here, unlike the ECD case, the true parameter $\boldsymbol{\theta}^*$ is a global minimum of $D(\boldsymbol{\theta})$, but it is not unique (this is because it is possible to permute mixture components).

The target problems (3.2) and (3.4) are optimisation problems on the Riemannian manifolds Θ and $\boldsymbol{\Theta}$, respectively. The general update rule, for offline (batch) gradient optimisation methods on Riemannian manifolds, takes on the following form [2]

$$\theta^{(n+1)} = \mathrm{Ret}_{\theta^{(n)}}\left(\eta^{(n)} U(\theta^{(n)}; \mathcal{X})\right) \tag{3.5}$$

with $U(\theta^{(n)}; \mathcal{X})$ the direction of descent. On the other hand, the general form of an update rule for online (stochastic) gradient optimisation methods is the following

$$\theta^{(n+1)} = \mathrm{Ret}_{\theta^{(n)}}\left(\eta^{(n+1)} U(\theta^{(n)}; \mathcal{X}_{n+1})\right) \tag{3.6}$$

Here, the smooth mapping Ret_θ from the tangent space $T_\theta\Theta$ to Θ is required to be a retraction, in the sense that it verifies

$$\begin{aligned} \mathrm{Ret}_\theta(0_\theta) &= \theta \\ \mathrm{dRet}_\theta(0_\theta) &= \mathrm{Id}_{T_\theta\Theta} \end{aligned} \tag{3.7}$$

where 0_θ denotes the zero element in $T_\theta\Theta$, $\mathrm{dRet}_\theta(\cdot)$ means the differential of retraction map $\mathrm{Ret}_\theta : T_\theta\Theta \to \Theta$, and $\mathrm{Id}_{T_\theta\Theta}$ denotes the identity mapping on $T_\theta\Theta$. The positive scalars $\eta^{(n)}$ are step-sizes.

For offline methods (3.5), each vector $U(\theta^{(n)}; \mathcal{X})$ depends on the entire dataset $\mathcal{X}$. For online methods (3.6), each vector $U(\theta^{(n)}; \mathcal{X}_{n+1})$ depends only on a new mini-batch $\mathcal{X}_{n+1}$, which may reduce to a single datapoint x_{n+1}. Ideally, $U(\theta^{(n)}; \mathcal{X})$ or $U(\theta^{(n)}; \mathcal{X}_{n+1})$ is the Riemannian information gradient, which can be computed, assuming the Fisher information metric is known.

However, in general, ECD and MECD models do not admit any tractable expression of the Fisher information metric. Therefore, there is no tractable means of evaluating the Riemannian information gradient.

In the present chapter, several specific situations are discussed. The Fisher information metric for $\theta = (\Sigma)$ (with known μ and β) has already been given in Subsection 2.2.2. In the following, Section 3.2 gives the Fisher information metric for $\theta = (\mu, \Sigma)$ (with known β), and introduces a retraction map Ret for this case $\theta = (\mu, \Sigma)$.

Sections 3.3 and 3.4, respectively, consider the most general cases, of complete ECD models and MECD models, where the shape parameter β is unknown. In these cases, the parameter space does not admit any tractable expression of the Fisher information metric [83]. Sections 3.3 and 3.4 introduce the component-wise information metric (CIM), as a computationally advantageous alternative to the Fisher information metric (FIM).

3.2 Geometry of (μ,Σ)

Fisher information metric of (μ, Σ)**:** The Fisher information metric for the case $\theta = (\Sigma)$ (with known and fixed μ and β), has already been discussed, in Subsection 2.2.2.

Furthermore, when the location parameter μ is also unknown, but the shape parameter β is still known, the Fisher information metric for $\theta = (\mu, \Sigma)$ can be computed explicitly. In fact, this metric takes on a simple form (this can be seen as a "direct product")

$$\langle U_\theta, V_\theta \rangle^*_\theta = \langle U_\mu, V_\mu \rangle^*_\mu + \langle U_\Sigma, V_\Sigma \rangle^*_\Sigma \tag{3.8}$$

where $\langle U_\mu, V_\mu \rangle^*_\mu$ and $\langle U_\Sigma, V_\Sigma \rangle^*_\Sigma$ are Fisher information metrics in $\mathbb{R}^m$ and $\mathcal{P}_m$ respectively. Here, each tangent vector $U_\theta, V_\theta \in T_\theta\Theta$ is of the form $U_\theta = (U_\mu, U_\Sigma)^\dagger$ and $V_\theta = (V_\mu, V_\Sigma)^\dagger$. Moreover, the second term $\langle U_\Sigma, V_\Sigma \rangle_\Sigma$ is given by the same formula as in (2.6). For the first term, we have computed it to be [90],

$$\langle U_\mu, V_\mu \rangle^*_\mu = I_\mu \, U_\mu^\dagger \, \Sigma^{-1} \, V_\mu \tag{3.9}$$

where the information coefficient I_μ is given by

$$I_\mu = -\frac{4}{m} \mathbb{E}\left[\frac{\partial^2 \, h(\delta, \beta)}{\partial \, \delta^2} \, \delta \right] - 2\mathbb{E}\left[\frac{\partial \, h(\delta, \beta)}{\partial \, \delta} \right] \tag{3.10}$$

Adding up (2.6) and (3.9), one gets the full Fisher information metic for $\theta = (\mu, \Sigma)$,

$$\langle U_\theta, V_\theta \rangle_\theta = I_\mu \, U_\mu^\dagger \, \Sigma^{-1} \, V_\mu + I_{\Sigma,1} \mathrm{tr}(\Sigma^{-1} U_\Sigma \Sigma^{-1} V_\Sigma) + I_{\Sigma,2} \mathrm{tr}(\Sigma^{-1} U_\Sigma) \mathrm{tr}(\Sigma^{-1} V_\Sigma) \tag{3.11}$$

where the information constants $I_{\Sigma,1}$ and $I_{\Sigma,2}$ were given in equation (2.7). The Fisher information metric (3.11) generalises the one studied in the multivariate Gaussian case [9].

Retraction map of (μ, Σ)**:** The Fisher information metric (3.11) on the product manifold $\Theta = \mathbb{R}^m \times \mathcal{P}_m$, does not admit a tractable expression, for its Riemannian exponential map. However, when it is restricted to the subspace $\mathcal{P}_m$, its exponential map is well-known, and was given in equation (2.2). In addition, the exponential map on the subspace $\mathbb{R}^m$ (a Euclidean space) reduces to vector addition. Accordingly, we proposed to use a retraction map which is defined by the product of these two exponentials [90],

$$\begin{array}{rccc} \mathrm{Ret}_\theta : & T_\theta\Theta & \longrightarrow & \Theta \\ & U_\theta = \begin{pmatrix} U_\mu \\ U_\Sigma \end{pmatrix} & \longmapsto & \begin{pmatrix} \mathrm{Exp}_\mu(U_\mu) \\ \mathrm{Exp}_\Sigma(U_\Sigma) \end{pmatrix} = \begin{pmatrix} \mu + U_\mu \\ \Sigma \exp(\Sigma^{-1} U_\Sigma) \end{pmatrix} \end{array} \tag{3.12}$$

Both of the exponential maps Exp_μ and Exp_Σ verify the properties (3.7). Therefore, their direct product (3.12) also verifies these properties, and is a well-defined retraction.

Information distance of (μ, Σ)**:** We also introduced a distance function on the product manifold $\Theta = \mathbb{R}^m \times \mathcal{P}_m$, defined as the sum of squares of the distances in the subspaces $\mathbb{R}^m$ and $\mathcal{P}_m$. This will be called the component-wise information distance (CID) [90].

For $\mathcal{P}_m$, the information distance is defined as in Equation (2.8). For $\mathbb{R}^m$, the information distance is proportional to the usual Euclidean distance. Now, the component-wise information distance is the following distance on Θ [90],

$$\begin{aligned} d^2_*(\theta_1, \theta_2) =& d^2_*(\mu_1, \mu_2) + d^2_*(\Sigma_1, \Sigma_2) \\ =& I_\mu \, (\mu_1 - \mu_2)^\dagger (\mu_1 - \mu_2) \\ & + I_{\Sigma,1} \mathrm{tr}\left[\log(\Sigma_1^{-1}\Sigma_2)\right]^2 + I_{\Sigma,2} \mathrm{tr}^2\left[\log(\Sigma_1^{-1}\Sigma_2)\right] \end{aligned} \tag{3.13}$$

where $\theta_1 = (\mu_1, \Sigma_1)$ and $\theta_2 = (\mu_2, \Sigma_2)$, with $\mu_1, \mu_2 \in \mathbb{R}^m$ and $\Sigma_1, \Sigma_2 \in \mathcal{P}_m$. The constant I_μ was given in (3.10), and the constants $I_{\Sigma,1}$ and $I_{\Sigma,2}$ were given in (2.7).

Information gradient of (μ, Σ): The information gradient associated to the Fisher information metric (3.11) is just the product of the information gradients, with respect to μ and Σ [90],

$$\nabla^*_\theta \ell_p(\theta; x) \;=\; \begin{pmatrix} \nabla^*_\mu \ell_p(\theta; x) \\ \nabla^*_\Sigma \ell_p(\theta; x) \end{pmatrix} = \begin{pmatrix} I_\mu^{-1}\, \Sigma\, \mathrm{grad}_\mu \ell_p(\theta; x) \\ J_{\Sigma,1}\; [\nabla^\uparrow_\Sigma \ell_p(\theta; x)]^\perp \;+\; J_{\Sigma,2}\; [\nabla^\uparrow_\Sigma \ell_p(\theta; x)]^\parallel \end{pmatrix} \tag{3.14}$$

where the coefficients $J_{\Sigma,1}$ and $J_{\Sigma,2}$ of $\nabla^*_\Sigma \ell_p$ were given in equation (2.10), and the information constant I_μ in (3.10). Here, the vector $\mathrm{grad}_\mu \ell_p(\theta; x)$ is the gradient in the classical Euclidean sense

$$\mathrm{grad}_\mu \ell_p(\theta; x) = -2 \frac{\partial h_\beta(\delta)}{\partial \delta} \Sigma^{-1}(x - \mu) \tag{3.15}$$

and the affine-invariant gradient $\nabla^\uparrow_\Sigma \ell_p(\theta; x)$ is given by Equation (2.5).

3.3 Geometry of (μ,Σ,β)

When the shape parameter β is also unknown, the Fisher information metric of the complete ECD model, with $\theta = (\mu, \Sigma, \beta)$, does not have a closed form expression, and it is therefore impossible to derive applicable expressions of the Riemannian exponential map, the information gradient, or any other useful geometric objects.

A similar situation arises in the field of artificial neural networks, where the Fisher information metric is too difficult to compute. The Fisher information metric is then replaced with a quasi-diagonal reduction [53]. Motivated by this idea, from the field of artificial intelligence, we introduce the component-wise information metric of ECD models [90].

Each one of the three parameters μ, Σ, and β has a closed form expression for its Fisher information metric. These closed form expressions give the three main diagonal blocks of the ECD model Fisher information metric. The other (off-diagonal) blocks being unknown, we just set them to zero. In this way, we obtain the component-wise information metric (CIM) of the ECD model, at $\theta = (\mu, \Sigma, \beta)$,

$$\mathcal{I}^{\odot}(\theta) = \begin{array}{c} \\ U_\mu \\ U_\Sigma \\ U_\beta \end{array} \begin{array}{c} \begin{array}{ccc} V_\mu & V_\Sigma & V_\beta \end{array} \\ \begin{bmatrix} \mathcal{I}_\mu & 0 & 0 \\ 0 & \mathcal{I}_\Sigma & 0 \\ 0 & 0 & \mathcal{I}_\beta \end{bmatrix} \end{array}$$

The blocks $\mathcal{I}_\mu$, $\mathcal{I}_\Sigma$ and $\mathcal{I}_\beta$ are respectively the Fisher information matrics of $\mathbb{R}^m$, $\mathcal{P}_m$ and $\mathbb{R}_+$ for ECD.

Component-wise information metric of (μ, Σ, β): The Fisher information metric for (μ, Σ) was given in Equation (3.11). The component-wise information metric is obtained by adding to (3.11) a further term for the shape parameter β. The CIM of the ECD model follows [90]

$$\begin{aligned} \langle U_\theta, V_\theta \rangle_\theta^{\odot} &= \langle U_\mu, V_\mu \rangle_\mu^* + \langle U_\Sigma, V_\Sigma \rangle_\Sigma^* + \langle U_\beta, V_\beta \rangle_\beta^* \\ &= I_\mu \, U_\mu^\dagger \, \Sigma^{-1} \, V_\mu + I_{\Sigma,1} \mathrm{tr}(\Sigma^{-1} U_\Sigma \Sigma^{-1} V_\Sigma) + I_{\Sigma,2} \mathrm{tr}(\Sigma^{-1} U_\Sigma) \mathrm{tr}(\Sigma^{-1} V_\Sigma) \\ &\quad + I_\beta \, U_\beta \, V_\beta \end{aligned} \tag{3.16}$$

where $\odot$ is the symbol for component-wise information metric, $U_\theta = (U_\mu, U_\Sigma, U_\beta)$ and $V_\theta = (V_\mu, V_\Sigma, V_\beta)$ are tangent vectors at the point $\theta = (\mu, \Sigma, \beta)$. The information constants $I_{\Sigma,1}$, $I_{\Sigma,2}$ and I_μ are given in (2.7) and (3.10), respectively. The information constant of β is

$$I_\beta = -\mathbb{E}\left[\frac{\partial^2 \log c(\beta)}{\partial \beta^2} + \frac{\partial^2 h_\beta(\delta)}{\partial \beta^2} \right] \tag{3.17}$$

Retraction map of (μ, Σ, β): We introduce a retraction map for (μ, Σ, β), using the same idea as in (3.12). Recall that the exponential map for μ is the vector addition, and the intrinsic exponential map on $\mathcal{P}_m$ was given in equation (2.2). Since β belongs to $\mathbb{R}_+$, a $1-$dimensional version of (2.2) is used as the exponential map for β.

Combining these three exponential maps, The retraction for the complete ECD model

is given as [90],

$$\begin{aligned} \mathrm{Ret}_\theta : \qquad T_\theta\Theta \qquad &\longrightarrow \qquad\qquad \Theta \\ U_\theta = \begin{pmatrix} U_\mu \\ U_\Sigma \\ U_\beta \end{pmatrix} \quad &\longmapsto \quad \begin{pmatrix} \mathrm{Exp}_\mu(U_\mu) \\ \mathrm{Exp}_\Sigma(U_\Sigma) \\ \mathrm{Exp}_\beta(U_\beta) \end{pmatrix} = \begin{pmatrix} \mu + U_\mu \\ \Sigma \exp(\Sigma^{-1} U_\Sigma) \\ \beta e^{U_\beta/\beta} \end{pmatrix} \end{aligned} \tag{3.18}$$

where U_θ is the tangent vector (in the following, U_θ will represent the direction of descent). All the three exponential maps Exp_μ, Exp_Σ and Exp_β verify the properties (3.7). Therefore, their direct product (3.18) also verifies these properties, and is a well-defined retraction.

Component-wise information distance of (μ, Σ, β): The information distance of (μ, Σ) was given in (3.13). In order to include β, which belongs to $(0, \infty)$, a one-dimensional version of Equation (2.4) is added to (3.13). The component-wise information distance for the complete model (with all three parameters μ, Σ and β), then has the expression [90]

$$\begin{aligned} d^2_\odot(\theta_1, \theta_2) =& d^2_*(\mu_1, \mu_2) + d^2_*(\Sigma_1, \Sigma_2) + d^2_*(\beta_1, \beta_2) \\ =& I_\mu \, (\mu_1 - \mu_2)^\dagger (\mu_1 - \mu_2) + I_{\Sigma,1} \mathrm{tr} \left[\log(\Sigma_1^{-1}\Sigma_2)\right]^2 + I_{\Sigma,2} \mathrm{tr}^2 \left[\log(\Sigma_1^{-1}\Sigma_2)\right] \\ &+ I_\beta \, \log^2 \left(\beta_1^{-1} \, \beta_2\right) \end{aligned} \tag{3.19}$$

Where the information constants I_μ, $I_{\Sigma,1}$, $I_{\Sigma,2}$ and I_β are given respectively in (3.10), (2.7) and (3.17).

Component-wise information gradient of (μ, Σ, β): Consider now the Component-wise Information Gradient (CIG for short), derived from the metric (3.16). The CIG is the unique vector field $\nabla^\odot_\theta \ell_p(\theta; x)$ on Θ which satisfies

$$\mathrm{d}\, \ell_p(\theta; x) \, [U_\theta] \; = \; \left\langle \nabla^\odot_\theta \ell_p(\theta; x), U_\theta \right\rangle^\odot_\theta \qquad \text{recall } \ell_p = \log p \tag{3.20}$$

where the scalar product on the right-hand side is the CIM (3.16), and $\mathrm{d}\ell_p$ is the differential of ℓ_p. Accordingly, the component-wise information gradient has the following form [90]

$$\nabla^\odot_\theta \ell_p(\theta; x) \; = \; \begin{pmatrix} \nabla^*_\mu \ell_p(\theta; x) \\ \nabla^*_\Sigma \ell_p(\theta; x) \\ \nabla^*_\beta \ell_p(\theta; x) \end{pmatrix} = \begin{pmatrix} I_\mu^{-1} \, \Sigma \, \mathrm{grad}_\mu \ell_p(\theta; x) \\ J_{\Sigma,1} \, \left[\nabla^\uparrow_\Sigma \ell_p(\theta; x)\right]^\perp \, + \, J_{\Sigma,2} \, \left[\nabla^\uparrow_\Sigma \ell_p(\theta; x)\right]^\parallel \\ I_\beta^{-1} \mathrm{grad}_\beta \ell_p(\theta; x) \end{pmatrix} \tag{3.21}$$

where the coefficients I_μ, $J_{\Sigma,1}$, $J_{\Sigma,2}$ and I_β are respectively given in (2.7), (3.10), and (3.17). For the third component in (3.21), the derivative of ℓ_p with respect to β is

$$\mathrm{grad}_\beta \ell_p(\theta; x) = \frac{\partial \log c(\beta)}{\partial \beta} + \frac{\partial h_\beta(\delta)}{\partial \beta} \tag{3.22}$$

3.4 Mixtures of ECD

A mixture of ECD is given by the probability density $f(x|\boldsymbol{\theta})$ in (1.2). This density is parameterised by $\boldsymbol{\theta} = (w_k, \theta_k; k = 1, \cdots, K)$ where $\theta_k = (\mu_k, \Sigma_k, \beta_k)$ are the parameters of mixture components $p(x|\theta_k)$, which are single ECD of the form (1.1).

In this thesis, the mixture weights $w = (w_k)_k$ are mapped to a point $r = (r_k)_k$ on the unit sphere $S^{K-1} \subset \mathbb{R}^K$, by setting $w_k = r_k^2$. In this way, the parameter space $\boldsymbol{\Theta}$ of the MECD model is a product manifold, with subspaces S^{K-1} (for the weights), and $\Theta_k = \mathbb{R}^m \times \mathcal{P}_m \times \mathbb{R}_+$ (for each single ECD mixture component). In other words,

$$\boldsymbol{\Theta} = S^{K-1} \times \Theta_1 \times \ldots \times \Theta_K \qquad \Theta_k = \mathbb{R}^m \times \mathcal{P}_m \times \mathbb{R}_+$$

This parameter space will be given the structure of a d-dimensional Riemannian manifold (the dimension d is given by $d = K(m(m+3)/2+1)$).

Ideally, one hopes to equip $\boldsymbol{\Theta}$ with the Fisher information metric, and derive the corresponding Riemannian information gradient and exponential map. However, as in the single ECD case (of the previous section), this Fisher information metric does not have a closed form expression, and the same is true of the corresponding exponential map and information gradient.

Therefore, as in the previous section, a component-wise information metric, of quasi-diagonal form, is introduced on $\boldsymbol{\Theta}$,

$$\mathcal{I}^{\odot}(\boldsymbol{\theta}) = \begin{array}{c} \\ U_r \\ U_{\mu_k} \\ U_{\Sigma_k} \\ U_{\beta_k} \end{array} \begin{array}{c} \begin{array}{cccc} \mathrm{d}r & V_{\mu_k} & V_{\Sigma_k} & V_{\beta_k} \end{array} \\ \begin{bmatrix} \mathcal{I}_r & 0 & 0 & 0 \\ 0 & \mathcal{I}_{\mu_k} & 0 & 0 \\ 0 & 0 & \mathcal{I}_{\Sigma_k} & 0 \\ 0 & 0 & 0 & \mathcal{I}_{\beta_k} \end{bmatrix} \end{array}$$

Its explicit expression is [89]

$$\langle U_{\boldsymbol{\theta}}, V_{\boldsymbol{\theta}} \rangle^{\odot}_{\boldsymbol{\theta}} = \langle U_r, V_r \rangle^*_r + \sum_{k=1}^{K} \langle U_{\mu_k}, V_{\mu_k} \rangle^*_{\mu_k} + \sum_{k=1}^{K} \langle U_{\Sigma_k}, V_{\Sigma_k} \rangle^*_{\Sigma_k} + \sum_{k=1}^{K} \langle U_{\beta_k}, V_{\beta_k} \rangle^*_{\beta_k} \tag{3.23}$$

where $U_{\boldsymbol{\theta}}$, $V_{\boldsymbol{\theta}}$ are tangent vectors, in the tangent space $T_{\boldsymbol{\theta}}\boldsymbol{\Theta}$, and all sub-metrics are the Fisher information metrics in the corresponding subspaces. Precisely, the Fisher information metric $\langle \cdot, \cdot \rangle^*_r$ on unit sphere coincides with the scalar product in Euclidean sens [4]. For each mixture component, the Fisher information metric in each subspace has the same form as in Equation (3.16).

$$\langle U_{\mu_k}, V_{\mu_k} \rangle^*_{\mu_k} \;=\; I_{\mu_k}\, U^{\dagger}_{\mu_k}\, \Sigma_k^{-1}\, V_{\mu_k} \tag{3.24a}$$

$$\langle U_{\Sigma_k}, V_{\Sigma_k} \rangle^*_{\Sigma_k} = I_{\Sigma_k,1} \mathrm{tr}(\Sigma_k^{-1} U_{\Sigma_k} \Sigma_k^{-1} V_{\Sigma_k}) + I_{\Sigma_k,2} \mathrm{tr}(\Sigma_k^{-1} U_{\Sigma_k}) \mathrm{tr}(\Sigma_k^{-1} V_{\Sigma_k}) \tag{3.24b}$$

$$\langle U_{\beta_k}, V_{\beta_k} \rangle^*_{\beta_k} \;=\; I_{\beta_k}\, U_{\beta_k}\, V_{\beta_k} \tag{3.24c}$$

The information constants in Equations (3.24) are also equal to the values given above.

$$I_{\Sigma_k,1} = \frac{2\mathcal{A}}{m(m+2)} \qquad I_{\Sigma_k,2} = \frac{\mathcal{A}}{m(m+2)} - \frac{1}{4} \qquad \mathcal{A} = \mathbb{E}\left[\left(\frac{\partial h_{\beta_k}(\delta_k)}{\partial \delta_k} \delta_k \right)^2 \right] \tag{3.25a}$$

$$I_{\mu_k} = -\frac{4}{m} \mathbb{E}\left[\frac{\partial^2 h_{\beta_k}(\delta_k)}{\partial\, \delta_k^2}\, \delta_k \right] - 2\mathbb{E}\left[\frac{\partial\, h_{\beta_k}(\delta_k)}{\partial\, \delta_k} \right] \tag{3.25b}$$

$$I_{\beta_k} \;=\; -\mathbb{E}\left[\frac{\partial^2 \,\log c(\beta_k)}{\partial\, \beta_k^2} \;+\; \frac{\partial^2\, h_{\beta_k}(\delta_k)}{\partial\, \beta_k^2}\right] \tag{3.25c}$$

where $\delta_k = (x-\mu_k)^\dagger \Sigma_k^{-1}(x-\mu_k)$.

To define a retraction on $\boldsymbol{\Theta}$, the retraction map in (3.18) is used for each component θ_k. For the weights, as given by the parameters $r = (r_k)_k \in S^{K-1}$, the intrinsic Riemannian exponential on the sphere $S^{K-1} \subset \mathbb{R}^K$ has an easy expression

$$\mathrm{Exp}_r\,(U_r) = \cos\left(\|U_r\|\right) r + \frac{\sin\left(\|U_r\|\right)}{\|U_r\|} U_r \tag{3.26}$$

The direct product of these maps is chosen as the retraction for MECD [89],

$$\begin{array}{rccc} \mathrm{Ret}_\theta: & T_\theta\Theta & \to & \Theta \\ & U_\theta = \begin{pmatrix} U_r \\ (U_{\mu_k})_k \\ (U_{\Sigma_k})_k \\ (U_{\beta_k})_k \end{pmatrix} & \mapsto & \begin{pmatrix} \mathrm{Exp}_r\,(U_r) \\ \left(\mathrm{Exp}_{\mu_k}\,(U_{\mu_k})\right)_k \\ \left(\mathrm{Exp}_{\Sigma_k}\,(U_{\Sigma_k})\right)_k \\ \left(\mathrm{Exp}_{\beta_k}\,(U_{\beta_k})\right)_k \end{pmatrix} \end{array} \tag{3.27}$$

here, for any component k, following (3.18),

$$\begin{pmatrix} \mathrm{Exp}_{\mu_k}(U_{\mu_k}) \\ \mathrm{Exp}_{\Sigma_k}(U_{\Sigma_k}) \\ \mathrm{Exp}_{\beta_k}(U_{\beta_k}) \end{pmatrix} = \begin{pmatrix} \mu_k + U_{\mu_k} \\ \Sigma_k \exp(\Sigma_k^{-1} U_{\Sigma_k}) \\ \beta_k e^{U_{\beta_k}/\beta_k} \end{pmatrix}$$

The component-wise information gradient (CIG), derived from the metric (3.23), is given as [89],

$$\nabla^{\odot}_{\boldsymbol{\theta}} \ell_f\,(\boldsymbol{\theta}; x) = \begin{pmatrix} \nabla^*_r \ell_f\,(\boldsymbol{\theta}; x) \\ \left(\nabla^*_{\mu_k} \ell_f\,(\boldsymbol{\theta}; x)\right)_k \\ \left(\nabla^*_{\Sigma_k} \ell_f\,(\boldsymbol{\theta}; x)\right)_k \\ \left(\nabla^*_{\beta_k} \ell_f\,(\boldsymbol{\theta}; x)\right)_k \end{pmatrix} \tag{3.28}$$

Each element $\nabla^*_i \ell_f\,(\boldsymbol{\theta}; x)$ in this column vector is the information gradient with respect to the corresponding i-th subspace. Concretely, the information gradient with respect to r is

$$\nabla^*_r \ell_f\,(\boldsymbol{\theta}; x) = \frac{\partial}{\partial r} \ell_f\,(\boldsymbol{\theta}; x) - \left\langle \frac{\partial}{\partial r} \ell_f\,(\boldsymbol{\theta}; x)\,, r \right\rangle \times r \tag{3.29}$$

where $\langle \cdot, \cdot \rangle$ denotes the scalar product in $\mathbb{R}^K$ (the dot product). The information gradients for μ_k, Σ_k and β_k are consistent with (3.21)

$$\begin{pmatrix} \nabla^*_{\mu_k} \ell_f(\boldsymbol{\theta}; x) \\ \nabla^*_{\Sigma_k} \ell_f(\boldsymbol{\theta}; x) \\ \nabla^*_{\beta_k} \ell_f(\boldsymbol{\theta}; x) \end{pmatrix} = \begin{pmatrix} I_{\mu_k}^{-1}\, \Sigma_k\, \mathrm{grad}_{\mu_k} \ell_f(\boldsymbol{\theta}; x) \\ J_{\Sigma_k,1}\, \left[\nabla^{\uparrow}_{\Sigma_k} \ell_f(\boldsymbol{\theta}; x)\right]^{\perp} \;+\; J_{\Sigma_k,2}\, \left[\nabla^{\uparrow}_{\Sigma_k} \ell_f(\boldsymbol{\theta}; x)\right]^{\parallel} \\ I_{\beta_k}^{-1} \mathrm{grad}_{\beta_k} \ell_f(\boldsymbol{\theta}; x) \end{pmatrix}$$

Here, all necessary information constants are defined in (3.25). The vectors $\mathrm{grad}_{\mu_k} \ell_f(\boldsymbol{\theta}; x)$, $\nabla^{\uparrow}_{\Sigma_k} \ell_f(\boldsymbol{\theta}; x)$ and $\mathrm{grad}_{\beta_k} \ell_f(\boldsymbol{\theta}; x)$ are defined in (2.24) and (2.5).

3.5 Conclusion

Our main contribution to the geometry of ECD and MECD is the introduction of the component-wise information metric (CIM), in Sections 3.3 and 3.4 of the above. The CIM coincides with the Fisher information metric (FIM) whenever the shape parameter is known (when $\theta = (\Sigma)$ or $\theta = (\mu, \Sigma)$). When the shape parameter is unknown (when $\theta = (\mu, \Sigma, \beta)$), the CIM has a straightforward "direct product" structure, while the FIM does not admit any closed-form expression.

The following Table 3.5 details the geometric objects introduced in the present chapter, for each special case of ECD or MECD models.

Model	Summary
ECD with $\theta = (\Sigma)$	• the CIM coincides with the FIM (2.6) • the information gradient is given by (2.9) • Exp is tractable, given by (2.2)
ECD with $\theta = (\mu, \Sigma)$	• the CIM coincides with the FIM (3.11) • the information gradient is given by (3.14) • use Ret given by (3.12), instead of intractable Exp
ECD with $\theta = (\mu, \Sigma, \beta)$	• the CIM is different from the FIM, and given by (3.16) • the component-wise information gradient is given by (3.21) • use Ret given by (3.18), instead of intractable Exp
MECD with $\boldsymbol{\theta} = (w, \theta_1, \cdots, \theta_K)$	• the CIM is different from the FIM, and given by (3.23) • the component-wise information gradient is given by (3.28) • use Ret given by (3.27), instead of intractable Exp

Table 3.1: The CIM and related geometric objects for ECD and MECD

The above table shows that there are certain special cases where the CIM coincides with the FIM, which is known in closed form. In such cases, the Riemannian information gradient can be computed explicitly, and replaced into the general update rules (3.5) or (3.6). The following Chapter 4 will be exactly concerned with such cases, and studies Riemannian online estimation of a general statistical model, whose Fisher information metric is known. More difficult cases, where the CIM is used, because the FIM is unknown in closed form, will be the subject of subsequent chapters 5 and 6.

Chapter 4

Riemannian information gradient method

4.1 Introduction

The present chapter reflects our work, published in the journal paper [88]. Its central theme is the application of the Riemannian information gradient method, to online (also called recursive or stochastic) estimation of statistical models parameterised on Riemannian manifolds.

For these statistical models, the information gradient method can be applied, under the assumption that the Fisher information metric is known in closed form. Then, the information gradient can be computed directly from the Fisher information metric. The information gradient method greatly simplifies the task of selecting step-sizes. Moreover, this method guarantees fast convergence, asymptotic normality, and asymptotic efficiency of online estimation.

Unlike other chapters of this thesis, the present chapter does not focus solely on ECD models, but considers general statistical models, whose parameters belong to a Riemannian manifold. After stating general results on the Riemannian information gradient method, it returns to certain special cases of ECD, in order to illustrate these results.

The focus will be on an online update rule which falls under the general form (3.6). The aim of this update rule is to minimise a statistical divergence function $D(\theta)$ (such as the Kullback-Leibler divergence), defined on some Riemannian manifold Θ. This minimisation is equivalent to finding the true value θ^* of the statistical parameter θ, as already discussed in Section 3.1, before equation (3.1).

Specifically, the general online update rule considered in the following is (4.1a). The Riemannian information gradient method is a special case of this general rule, which will be given by (4.5). The main problem considered in the present chapter is how to

choose the step-sizes in (4.1a), in order to obtain fast and asymptotically efficient online estimates. In solving this problem, several original results are introduced.

First, under mild assumptions on the divergence function, it is proved that, with an adequate choice of step-sizes, the update rule computes online estimates which achieve a fast rate of convergence.

Second, the asymptotic normality of these online estimates is proved, by employing a novel linearisation technique (see Appendix A.1.4).

Third, it is proved that the Riemannian information gradient method is asymptotically efficient, in the sense that it achieves an optimal asymptotic rate of convergence.

These results, while relatively familiar in the Euclidean context (for example, see [25, 58]), were formulated and proved for the first time, in a Riemannian context, in our paper [88]. They will be illustrated with a numerical application to the online estimation of the scatter matrix of an ECD model, with known location and shape parameters.

The mathematical problem, considered in the present chapter, is formulated in Section 4.2. This involves a parameterised statistical model P of probability distributions P_θ, where the statistical parameter θ belongs to a Riemannian manifold Θ. Given independent observations, with distribution P_{θ^*} for some $\theta^* \in \Theta$, the aim is to estimate the unknown true parameter θ^*.

In principle, this is done by minimising a statistical divergence function $D(\theta)$, which measures the dissimilarity between P_θ and P_{θ^*}. Taking advantage of the observations, there are two approaches to minimising $D(\theta)$: stochastic minimisation, which leads to online estimation, and empirical minimisation, which leads to classical techniques, such as maximum-likelihood estimation [15, 16].

The original results, obtained in the present chapter are stated in Section 4.3. In particular, these are Propositions 2, 4, and 5. Overall, these propositions show that online estimation, which requires less computational resources than maximum-likelihood estimation, can still achieve the same optimal performance, characterised by asymptotic efficiency. Recall that asymptotic efficiency means the asymptotic Cramér-Rao lower bound is achieved [44, 78].

To summarise these propositions, consider a sequence of online estimates $\theta^{(n)}$, computed using a Riemannian online update rule of the form (3.6). Informally, under assumptions which guarantee that θ^* is an attractive local minimum of $D(\theta)$, and that the updates are neither too noisy, nor too unstable, in the neighborhood of θ^*,

• Proposition 2 states that, with an adequate choice of step-sizes, the $\theta^{(n)}$ achieve a fast rate of convergence to θ^*. Precisely, the expectation of the squared Riemannian distance between $\theta^{(n)}$ and θ^* is $O(n^{-1})$. This is called a fast rate, because it is the best achievable, for any step-sizes which are proportional to n^{-q} with $q \in (1/2, 1]$ [7, 25].

• Proposition 4 states that the distribution of the $\theta^{(n)}$ becomes asymptotically normal, centred at θ^*, when n grows increasingly large, and also characterises the corresponding asymptotic covariance matrix.

• Proposition 5 is concerned with the Riemannian information gradient method. If the Riemannian manifold Θ is equipped with the Fisher information metric of the statistical model P, then Riemannian information gradient descent computes online estimates $\theta^{(n)}$ which are asymptotically efficient. This is illustrated, with a numerical application to the online estimation of the scatter matrix of an ECD model, in Section 4.

The end-result of Proposition 5 is that the Riemannian information gradient method, which uses the Fisher information metric, achieves asymptotic efficiency. In [77], an alternative route to asymptotic efficiency is proposed, using the averaged Riemannian stochastic gradient method. This method does not require any prior knowledge of the

Fisher information metric, but has an additional computational cost, which comes from computing on-line Riemannian averages.

The proofs of Propositions 2, 4, and 5 are detailed in Appendices A.1, A.2 and A.3.

4.2 Problem statement

Let $P = (P, \Theta, X)$ be a statistical model, with parameter space Θ and sample space X. To each $\theta \in \Theta$, the model P associates a probability distribution P_θ on X. Here, we consider the case where Θ is a Riemannian manifold, and X is any measurable space.

The Riemannian metric of Θ will be denoted $\langle \cdot, \cdot \rangle$, with its Riemannian distance $d(\cdot, \cdot)$. In general, the metric $\langle \cdot, \cdot \rangle$ is not the information metric of the model P.

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a complete probability space, and $(x_n \,; n = 1, 2, \cdots)$ be i.i.d. random variables on Ω, with values in X. While the distribution of x_n is unknown, it is assumed to belong to the model P. That is, $\mathbb{P} \circ x_n^{-1} = P_{\theta^*}$ for some $\theta^* \in \Theta$, which is the true parameter.

Consider the following problem : how to obtain fast, asymptotically efficient, online estimates $\theta^{(n)}$ of the true parameter θ^*, based on observations of the random variables x_n? The present work proposes to solve this problem through a detailed study of the decreasing-step-size online update rule,

$$\theta^{(n+1)} = \mathrm{Exp}_{\theta^{(n)}} \left(\eta^{(n+1)} \, U(\theta^{(n)}, \mathcal{X}_{mb}^{new}) \right) \qquad n = 0, 1, \cdots \tag{4.1a}$$

starting from an initial guess $\theta^{(0)}$. Here, $\mathcal{X}_{mb}^{new}$ could be just one sample or a mini-batch. When it is one sample, it is also denoted as x_n. This update rule is similar to the one in [11], and falls under the general form (3.6). Instead of the general retraction Ret in (3.6), it uses the Riemannian exponential Exp of the metric $\langle \cdot, \cdot \rangle$ of Θ. In the following, the step-sizes $\eta^{(n)}$ are strictly positive, decreasing, and verify the usual conditions for stochastic approximation [48, 58]

$$\sum \eta^{(n)} \, = \, \infty \qquad \sum (\eta^{(n)})^2 \, < \, \infty \tag{4.1b}$$

Moreover, $U(\theta, x)$ is a continuous vector field on Θ for each $x \in X$, which generalises the classical concept of score statistic [39, 44].

A priori knowledge about the model P is injected into the update rule (4.1a) using a divergence function $D(\theta)$. As defined in [3], this is a positive function, equal to zero if and only if $P_\theta = P_{\theta^*}$, and with positive-definite Hessian at $\theta = \theta^*$. Since one expects that minimising $D(\theta)$ will lead to estimating θ^*, it is natural to require that

$$\mathbb{E}_{\theta^*} U(\theta, x) \, = \, -\nabla D(\theta) \tag{4.1c}$$

In other words, that $U(\theta, x)$ is an unbiased estimator of minus the Riemannian gradient of $D(\theta)$. With $U(\theta, x)$ given by (4.1c), the update rule (4.1a) is a Riemannian stochastic gradient method, of the form considered in [11, 77, 86].

In practice, a suitable choice of $D(\theta)$ is often the Kullback-Leibler divergence [72],

$$D(\theta) \, = \, - \, \mathbb{E}_{\theta^*} \left[\ell(\theta; x) \right] \qquad \ell(\theta; x) \, = \, \log \left(\frac{p_\theta(x)}{p_{\theta^*}(x)} \right) \tag{4.2}$$

where it is assumed that P_θ and P_{θ^*} have common support, with probability density functions $p_\theta(x)$ and $p_{\theta^*}(x)$, respectively. Indeed, if $D(\theta)$ is chosen to be the Kullback-Leibler divergence, then (4.1c) is satisfied by

$$U(\theta, x) = \nabla \ell(\theta; x) \tag{4.3}$$

which, in many practical situations, can be evaluated directly.

Riemannian information gradient method:
When the Fisher information metric $\langle \cdot, \cdot \rangle^*$ of the model P is known, it is natural to equip Θ with this Fisher information metric. Then, the Riemannian gradient on the right-hand side of equation (4.3) coincides with the Riemannian information gradient,

$$U(\theta, x) = \nabla^* \ell(\theta; x) \tag{4.4}$$

If this expression of $U(\theta, x)$ is replaced into equation (4.1a), this equation becomes

$$\theta^{(n+1)} = \mathrm{Exp}_{\theta^{(n)}} \left(\eta^{(n+1)} \nabla^*_\theta \ell(\theta^{(n)}; \mathcal{X}^{new}_{mb}) \right) \tag{4.5}$$

which will be called the Riemannian information gradient method. ■

Before stating the main results, in the following Section 4.3, it may be helpful to recall some general background, on online estimation [58]. For simplicity, let $D(\theta)$ be the Kullback-Leibler divergence (4.2). The problem of estimating the true parameter θ^* is equivalent to the problem of finding a global minimum of $D(\theta)$. Of course, this problem cannot be tackled directly, since $D(\theta)$ cannot be computed without knowledge of θ^*. There exist two routes, around this difficulty.

The first route is empirical minimisation, which replaces the expectation in (4.2) with an empirical mean over observed data. Given the first N observations, instead of minimising $D(\theta)$, one minimises the empirical divergence $\hat{D}(\theta)$,

$$\hat{D}(\theta) \; = \; - \frac{1}{N} \sum_{n=1}^{N} \ell(\theta; x_n) \tag{4.6}$$

where $\ell(\theta; x)$ is the likelihood function of (4.2). Now, given the minus sign ahead of the sum in (4.6), it is clear that minimising $\hat{D}(\theta)$ amounts to maximising the sum of log-likelihoods. Thus, this lead to the method of maximum-likelihood estimation.

It is well-known that maximum-likelihood estimation, under general regularity conditions, is asymptotically efficient [44]. Roughly, this means the maximum-likelihood estimator has the least possible asymptotic variance, equal to the inverse of the Fisher information matrix. On the other hand, as the number N of observations grows very large, it can be especially difficult to deal with the empirical divergence $\hat{D}(\theta)$ of (4.6). In the process of searching for the minimum of $\hat{D}(\theta)$, each evaluation of this function, or of its derivatives, will involve a massive number of operations, ultimately becoming unpractical.

Aiming to avoid this difficulty, the second route, that applies to online estimation, is based on observation-driven updates, following the general scheme of (4.1a). In this scheme, each new online estimate $\theta^{(n+1)}$ is computed using only one new observation x_{n+1}, or eventually a mini-batch. Therefore, as the number of observations grows very large, the overall computational effort required by online estimation, remains the same.

The main results in the following section show that online estimation can achieve the same asymptotic performance as maximum-likelihood estimation, as the number N of observations grows large. As a word of caution, it should be said that online estimation does not, in general, fare better than maximum-likelihood estimation for moderate values of the number N of observations, and models with a small number of parameters. However, it is a very desirable substitute to maximum-likelihood estimation for models with a large number of parameters, which typically require a very large number of observations, in order to be estimated correctly.

4.3 Main results

The motivation of the following Propositions 1 to 5 is to provide general conditions, which guarantee that the update rule (4.1a) computes fast, asymptotically efficient, online estimates $\theta^{(n)}$ of the true parameter θ^*. In the statement of these propositions, it is implicitly assumed that conditions (4.1b) and (4.1c) are verified. Moreover, the following assumptions are considered.

- (**d1**) the divergence function $D(\theta)$ has an isolated stationary point at $\theta = \theta^*$, and Lipschitz gradient in a neighborhood of this point.
- (**d2**) this stationary point is moreover attractive : $D(\theta)$ is twice differentiable at $\theta = \theta^*$, with positive-definite Hessian at this point.
- (**u1**) in a neighborhood of $\theta = \theta^*$, the function $V(\theta) = \mathbb{E}_{\theta^*} \|U(\theta, x)\|_\theta^2$ is uniformly bounded.
- (**u2**) in a neighborhood of $\theta = \theta^*$, the function $R(\theta) = \mathbb{E}_{\theta^*} \|U(\theta, x)\|_\theta^4$ is uniformly bounded.

For Assumption (d1), the definition of a Lipschitz vector field on a Riemannian manifold may be found in [56]. For Assumptions (u1) and (u2), $\|\cdot\|$ denotes the Riemannian norm.

Assumptions (u1) and (u2) are so-called moment control assumptions. They imply that the noisy nature of the observations x_n does not cause the iterates $\theta^{(n)}$ to diverge, and are also crucial to proving the asymptotic normality of these iterates.

Let Θ^* be a neighborhood of θ^* which verifies (d1), (u1), and (u2). Without loss of generality, it is assumed that Θ^* is compact and convex (see the definition of convexity in [62, 72]). Then, Θ^* admits a system of normal coordinates $(\theta^\alpha; \alpha = 1, \cdots, d)$ with origin at θ^*. With respect to these coordinates, denote the components of $U(\theta^*, x)$ by $u^\alpha(\theta^*)$ and let $\Sigma^* = (\Sigma^*_{\alpha\beta})$,

$$\Sigma^*_{\alpha\beta} \;=\; \mathbb{E}_{\theta^*} \left[u^\alpha(\theta^*)\, u^\beta(\theta^*) \right] \tag{4.7}$$

When (d2) is verified, denote the components of the Hessian of $D(\theta)$ at $\theta = \theta^*$ by $H = (H_{\alpha\beta})$,

$$H_{\alpha\beta} \;=\; \left. \frac{\partial^2 D}{\partial\theta^\alpha \partial\theta^\beta} \right|_{\theta^\alpha = 0} \tag{4.8}$$

Then, the matrix $H = (H_{\alpha\beta})$ is positive-definite [2]. Denote by $\lambda > 0$ its smallest eigenvalue.

Propositions 1 to 5 require the condition that the estimates $\theta^{(n)}$ are stable, which means that all the $\theta^{(n)}$ lie in Θ^*, almost surely. The need for this condition is discussed in Remark 3. Note that, if $\theta^{(n)}$ lies in Θ^*, then $\theta^{(n)}$ is determined by its normal coordinates $(\theta^{(n)})^\alpha$.

Proposition 1 (consistency) *assume (**d1**) and (**u1**) are verified, and the estimates $\theta^{(n)}$ are stable. Then, $\lim \theta^{(n)} = \theta^*$ almost surely.*

Proposition 2 (mean-square rate) *assume (**d1**), (**d2**) and (**u1**) are verified, the estimates $\theta^{(n)}$ are stable, and $\eta^{(n)} = \frac{a}{n}$ where $2\lambda a > 1$. Then*

$$\mathbb{E}\, d^2(\theta^{(n)}, \theta^*) \;=\; \mathcal{O}\left(n^{-1}\right) \tag{4.9}$$

Proposition 3 (almost-sure rate) *assume the conditions of Proposition 2 are verified. Then,*

$$d^2(\theta^{(n)}, \theta^*) \,=\, o(n^{-p}) \text{ for } p \in (0,1) \qquad \text{almost surely} \tag{4.10}$$

Proposition 4 (asymptotic normality) *assume the conditions of Proposition 2, as well as assumption (**u2**), are verified. Then, the distribution of the re-scaled coordinates $[n^{1/2}\,(\theta^{(n)})^\alpha]$ converges to a centred d-variate normal distribution, with covariance matrix Σ given by Lyapunov's equation*

$$A\,\Sigma \,+\, \Sigma\,A \,=\, -a^2\,\Sigma^* \tag{4.11}$$

where $A = (A_{\alpha\beta})$ with $A_{\alpha\beta} \,=\, \frac{1}{2}\delta_{\alpha\beta} \,-\, aH_{\alpha\beta}$ (here, δ denotes Kronecker's delta).

Proposition 5 (asymptotic efficiency) *assume the Riemannian metric $\langle\cdot,\cdot\rangle$ of Θ coincides with the information metric of the model P, and let $D(\theta)$ be the Kullback-Leibler divergence (4.2). Further, assume (**d1**), (**d2**), (**u1**) and (**u2**) are verified, the online estimates $\theta^{(n)}$ are stable, and $\eta^{(n)} = \frac{a}{n}$ where $2a > 1$. Then,*

(i) the rates of convergence (4.9) and (4.10) hold true.

(ii) if $a = 1$, the distribution of the re-scaled coordinates $[n^{1/2}\,(\theta^{(n)})^\alpha]$ converges to a centred d-variate normal distribution, with covariance matrix Σ^*.

(iii) if $a = 1$, and $U(\theta, x)$ is given by (4.4), then Σ^* is the identity matrix, and the online estimates $\theta^{(n)}$ are asymptotically efficient.

(iv) the following rates of convergence also hold

$$\mathbb{E}\,D(\theta^{(n)}) \,=\, \mathcal{O}(n^{-1}) \tag{4.12a}$$

$$D(\theta^{(n)}) \,=\, o(n^{-p}) \text{ for } p \in (0,1) \qquad \text{almost surely} \tag{4.12b}$$

Remark 1 *assumptions (**d2**), (**u1**) and (**u2**) do not depend on the Riemannian metric $\langle\cdot,\cdot\rangle$ of Θ. Precisely, if they are verified for one Riemannian metric on Θ, then they are verified for any Riemannian metric on Θ. Moreover, if the function $D(\theta)$ is C^2, then the same is true for assumption (**d1**). In this case, Propositions 1 to 5 apply for any Riemannian metric on Θ, so that the choice of the metric $\langle\cdot,\cdot\rangle$ is a purely practical matter, to be decided according to applications.*

Remark 2 *the conclusion of Proposition 1 continues to hold, if (4.1c) is replaced by*

$$\mathbb{E}_{\theta^*}\,\langle U(\theta, x), \nabla D(\theta)\rangle < 0 \text{ for } \theta \neq \theta^* \tag{4.13}$$

Then, it is even possible to preserve Propositions 2, 3, and 4, provided assumption (d2) is replaced by the assumption that the mean vector field, $X(\theta) \,=\, \mathbb{E}_{\theta^*} U(\theta, x)$, has an attractive stationary point at $\theta = \theta^*$. This generalisation of Propositions 1 to 4 can be achieved following essentially the same approach as laid out in Section 4.3. However, in the present work, it will not be carried out in detail.

Remark 3 *the condition that the online estimates $\theta^{(n)}$ are stable is standard in all prior work on stochastic optimisation in manifolds [11, 77, 86]. In practice, this condition can be enforced through replacing (4.1a) by a so-called projected or truncated update rule. This is identical to (4.1a), except that $\theta^{(n)}$ is projected back onto the neighborhood Θ^* of θ^*, whenever it falls outside of this neighborhood [48, 58]. On the other hand, if the $\theta^{(n)}$ are not required to be stable, but (d1) and (u1) are replaced by global assumptions,*

*(**d1'**) $D(\theta)$ has compact level sets and globally Lipschitz gradient.*

*(**u1'**) $V(\theta) \leqslant C\,(1 + D(\theta))$ for some constant C and for all $\theta \in \Theta$.*

then, applying the same arguments as in the proof of Proposition 1, it follows that the $\theta^{(n)}$ converge to the set of stationary points of $D(\theta)$, almost surely.

Remark 4 *from (ii) and (iii) of Proposition 5, it follows that the distribution of $n\,d^2(\theta^{(n)}, \theta^*)$ converges to a χ^2-distribution with d degrees of freedom. This provides a practical means of confirming the asymptotic efficiency of the online estimates $\theta^{(n)}$.*

4.4 Application to scatter-matrix estimation

Here, the conclusion of Proposition 5 is illustrated, by applying the update rule (4.1a) to the estimation of the scatter matrix of an ECD, with fixed location and shape parameters.

Precisely, in the notation of Section 4.2, let $\Theta = \mathcal{P}_m$ the space of $m \times m$ positive-definite matrices, and $X = \mathbb{R}^m$. For each $\theta \in \Theta$, let P_θ be the ECD distribution with density $p(x|\theta)$ given by (1.1), with location parameter $\mu = 0$, scatter matrix $\Sigma = \theta$, and known shape parameter $\beta = \beta^*$,

$$p(x|\theta) = c(\beta^*)\frac{1}{\sqrt{\det(\theta)}} g_{\beta^*}(\delta) \tag{4.14}$$

where $\delta = x^\dagger \theta^{-1} x$. In the following, the familiar notation $h_\beta(\delta) = \log g_\beta(\delta)$ is also used.

In addition, let $(x^{(n)}; n = 1, 2, \cdots)$ be i.i.d. random vectors in $\mathbb{R}^m$, with distribution P_{θ^*}, where $\theta^* \in \Theta$ is the true scatter matrix. The standard approach to estimating the true scatter matrix θ^* is based on maximum-likelihood estimation [59, 75]. An original approach, based on online estimation, is now introduced using the update rule (4.1a).

As in Proposition 5, the parameter space $\Theta = \mathcal{P}_m$ is equiped with the Fisher information metric (2.6). Here, this can be written as follows

$$\langle U, V\rangle_\theta^* \;=\; I_{\theta,1}\,\mathrm{tr}\left(\theta^{-1}U\theta^{-1}V\right) \;+\; I_{\theta,2}\,\mathrm{tr}\left(\theta^{-1}U\right)\mathrm{tr}\left(\theta^{-1}V\right) \tag{4.15}$$

for $U, V \in T_\theta\Theta$, where the information coefficients $I_{\theta,1}$ and $I_{\theta,2}$ are given by (2.7),

$$I_{\theta,1} = \frac{2\mathcal{A}}{m(m+2)} \qquad I_{\theta,2} = \frac{\mathcal{A}}{m(m+2)} - \frac{1}{4} \qquad \mathcal{A} = \mathbb{E}_\theta\left[\left(\frac{\partial h_{\beta^*}(\delta)}{\partial \delta}\delta\right)^2\right] \tag{4.16}$$

Note that these coefficients $I_{\theta,1}$ and $I_{\theta,2}$ do not depend on θ, but only on β^*.

The Fisher information metric with respect to θ being given by (4.15), it becomes possible to specify the update rule (4.1a). Indeed, recall from Section 2.2 that the exponential map of the metric (4.15) is given by (2.2),

$$\mathrm{Exp}_\theta(u) \;=\; \theta\exp\left(\theta^{-1}u\right) \tag{4.17a}$$

Then, according to (ii) of Proposition 5, consider the step-sizes

$$\eta^{(n)} = \frac{1}{n} \tag{4.17b}$$

Finally, according to (iii) of Proposition 5, let $U(\theta, x)$ be equal to the Riemannian information gradient,

$$U(\theta, x) = \nabla^*\ell(\theta; x) \tag{4.17c}$$

This is the gradient of $\ell(\theta; x) = \log p(x|\theta)$ with respect to the Fisher information metric (4.15), and can be computed from (2.9). Now, replacing (4.17) into (4.1a) defines an original update-rule (*i.e.* an original algorithm) for online estimation of the true scatter matrix θ^*.

Figures 4.1 and 4.2 below display numerical results from an application to MGGD, which correspond to $h(\delta) = -\frac{\delta^\beta}{2}$ in (4.14) and $\mathcal{A} = \frac{m}{2}\left(\frac{m}{2} + \beta\right)$ in (4.16) [8, 28].

These figures were generated from 10^3 Monte Carlo runs of the algorithm defined by (4.1a) and (4.17), with random initialisations, for the specific values $\beta = 4$ and $m = 7$. Essentially the same numerical results could be observed for any $\beta \leqslant 9$ and $m \leqslant 50$.

Figures 4.1 and 4.2 are concerned with the Riemannian information distance $d_*(\theta^{(n)}, \theta^*)$ induced by the Fisher information metric (4.15). This distance is given by (2.8)

$$d_*^2(\theta, \theta^*) = I_{\theta,1} \operatorname{tr}\left[\log\left(\theta^{-1}\theta^*\right)\right]^2 + I_{\theta,2} \operatorname{tr}^2\left[\log\left(\theta^{-1}\theta^*\right)\right] \qquad \theta\,, \theta^* \in \Theta \tag{4.18}$$

Figure 4.1 confirms the fast rate of convergence (4.9), stated in (i) of Proposition 5. On a log-log scale, it shows the empirical mean $\mathbb{E}_{MC} d_*^2(\theta^{(n)}, \theta^*)$ over Monte Carlo runs, as a function of n. This decreases with a constant negative slope equal to -1, starting roughly at $\log n = 4$.

Figure 4.2 confirms the asymptotic efficiency of the online estimates $\theta^{(n)}$, stated in (iii) of Proposition 5, using Remark 4.

Figure 4.2 shows a kernel density estimate of $n\, d_*^2(\theta^{(n)}, \theta^*)$ where $n = 10^5$ (solid blue curve). This agrees with a $\mathcal{X}^2$-distribution with 28 degrees of freedom (dotted red curve), where $d = 28$ is indeed the dimension of the parameter space $\Theta = \mathcal{P}_m$ for $m = 7$.

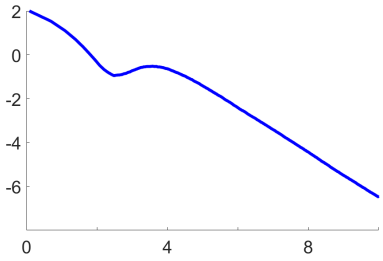


Figure 4.1: Riemannian information gradient: Fast rate of convergence

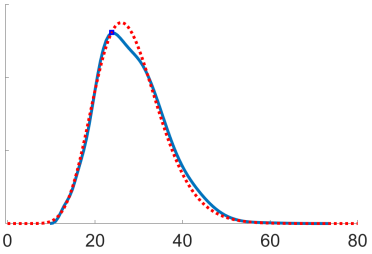


Figure 4.2: Riemannian information gradient: Asymptotic efficiency

4.5 Conclusion

This chapter introduced the Riemannian information gradient method, in the form of the update rule (4.5), applying it to online estimation of statistical models parameterised on Riemannian manifolds. The Riemannian information gradient method leads to a fast rate of convergence, and even to asymptotic efficiency. However, it can only be applied if the Fisher information metric is known.

The main results obtained in this chapter describe the convergence behavior of online estimates $\theta^{(n)}$, computed using the the general decreasing-step-size online update rule (4.1a), under Assumptions (d1), (d2), (u1), and (u2), formulated at the beginning of Section 4.3.

- Proposition 1 states that the $\theta^{(n)}$ converge almost surely to the true parameter θ^*, under Assumptions (d1) and (u1). The rate of almost sure convergence is stated in Proposition 3, with the additional Assumption (d2).

- Proposition 2 states the mean-square rate of convergence of the $\theta^{(n)}$. If the step-sizes $\eta^{(n)}$ are chosen correctly, this becomes a fast rate $\mathcal{O}(n^{-1})$. This proposition requires Assumptions (d1), (d2), and (u1).

- Propositions 4 states that the asymptotic distribution of the $\theta^{(n)}$ is a normal distribution, centred at θ^*, and characterises its covariance matrix. This proposition requires Assumptions (d1), (d2), (u1), and (u2).

- Proposition 5 states that, when the Fisher information metric is used to implement the update rule (4.1a) (this corresponds to the Riemannian information gradient method), there is an automatic choice of step-sizes ($\eta^{(n)} = 1/n$) which guarantees the $\theta^{(n)}$ are asymptotically efficient.

Section 4.4 applies Proposition 5 to the estimation of the scatter matrix of an ECD, with known location and shape parameters. It is impossible to extend this application to the case of a general ECD model (with unknown shape parameter), since the Fisher information metric is unknown, in this case. In order to overcome this difficulty, the following Chapter 5 will consider the component-wise information gradient method (CIG method).

Chapter 5

Estimation of ECD with CIG

5.1 Introduction

The main body of this chapter is based on our article [90], currently under review.

As shown in the previous chapter, the Fisher information metric and the related information gradient have significant advantages. The Riemannian information gradient method has a fast rate of convergence, and achieves asymptotic efficiency, with a simple (universal) choice of step-sizes ($\eta^{(n)} = 1/n$).

For a complete ECD model, with unknown parameter $\theta = (\mu, \Sigma, \beta)$, including location and shape parameters, the Fisher information metric does not have a closed form expression, so the Riemannian information gradient method cannot be applied.

Here, the Component-wise Information Metric (CIM), introduced in Chapter 3, will be used instead of the Fisher information metric (FIM). The gradient with respect to the CIM is the component-wise information gradient (CIG), which will now be used to define the CIG method for estimation of the complete ECD model.

The CIM is a quasi-diagonal reduction of the FIM. Thus, the CIG method retains some of the good properties of the Riemannian information gradient method (fast convergence and asymptotic efficiency, as seen in Proposition 5 of the previous Chapter 4), but not all of them. For example, the CIG method can still achieve a fast rate of convergence when estimating the true parameter $\theta^* = (\mu^*, \Sigma^*, \beta^*)$, but will fail to achieve asymptotic efficiency (theoretical results about the CIG method are given in Section 5.3).

The CIG method has an offline version, which can be applied to moderate-sized datasets, and an online version, more suited for recursive computations, which may be needed in applications such as change detection [13]. Both of these methods compute updated estimates $(\mu^{(n+1)}, \Sigma^{(n+1)}, \beta^{(n+1)})$ from current estimates $(\mu^{(n)}, \Sigma^{(n)}, \beta^{(n)})$, based

on the alternating scheme

$$\begin{aligned} &\text{step 1}: \mu^{(n+1)} \leftarrow (\mu^{(n)}, \Sigma^{(n)}, \beta^{(n)}) \\ &\text{step 2}: \Sigma^{(n+1)} \leftarrow (\mu^{(n+1)}, \Sigma^{(n)}, \beta^{(n)}) \\ &\text{step 3}: \beta^{(n+1)} \leftarrow (\mu^{(n+1)}, \Sigma^{(n+1)}, \beta^{(n)}) \end{aligned} \tag{5.1}$$

where each sub-parameter μ, Σ, or β is updated separately, in its own turn.

In the following, Section 5.2 starts with offline version of the CIG method. The online version and its theoretical convergence properties are introduced in Section 5.3. The question of the geodesic convexity of the Kullback-Leibler divergence $D(\theta)$, in the case of ECD estimation (recall Equation (3.1)), is studied in Section 5.4.

5.2 CIG offline

The CIG offline method is a second-order deterministic gradient method, somewhat similar to a Newton method. In the Newton method, the direction of descent is found by solving the Newton equation [2]. In the CIG offline method, the Hessian in the Newton equation is approximated by the component-wise information metric (or matrix) $\mathcal{I}^{\odot}(\theta)$, from Section 3.3.

For the CIG offline method, the choice of update direction depends on the complete dataset. The cost function (3.1) is reformulated, by replacing the KL divergence, with the empirical average of $-\ell_p(\theta; x)$, as in equation (4.6). This empirical average will be denoted by $\hat{D}(\theta)$.

If the current estimate is $\theta^{(n)} = (\mu^{(n)}, \Sigma^{(n)}, \beta^{(n)})$, the direction of update is given by one of the three components (depending on which sub-parameter is being updated)

$$\nabla^*_{\mu} \hat{D}(\theta^{(n)}) = -\frac{1}{T} \sum_{t=1}^{T} \nabla^*_{\mu} \ell_p(\theta^{(n)}; x_t) \tag{5.2a}$$

$$\nabla^*_{\Sigma} \hat{D}(\theta^{(n)}) = -\frac{1}{T} \sum_{t=1}^{T} \nabla^*_{\Sigma} \ell_p(\theta^{(n)}; x_t) \tag{5.2b}$$

$$\nabla^*_{\beta} \hat{D}(\theta^{(n)}) = -\frac{1}{T} \sum_{t=1}^{T} \nabla^*_{\beta} \ell_p(\theta^{(n)}; x_t) \tag{5.2c}$$

where the vectors $\nabla^*_{\mu}\ell_p$, $\nabla^*_{\Sigma}\ell_p$ and $\nabla^*_{\beta}\ell_p$ under the empirical averages are given in (3.21). Using the expressions in (5.2), the CIG offline algorithm can now be stated as follows.

Algorithm 1 CIG offline algorithm for ECD estimation

Input: A finite dataset $\mathcal{X}$, an initialization $\theta^{(0)} = (\mu^{(0)}, \Sigma^{(0)}, \beta^{(0)})$;
Output: The estimate $\hat{\theta}$;

1: **for** $n = 0, 1, 2 \cdots, N$ **do**
2: **Update μ:**
3: $\theta_{current} \leftarrow (\mu^{(n)}, \Sigma^{(n)}, \beta^{(n)})$;
4: Compute $\nabla^*_{\mu} \hat{D}(\theta_{current})$;
5: $\eta_{\mu} \leftarrow$ Armijo-Goldstein$(\mathcal{X}, \theta_{current})$;
6: $\mu^{(n+1)} \leftarrow \text{Exp}_{\mu^{(n)}} \left(-\eta_{\mu} \nabla^*_{\mu} \hat{D}(\theta_{current}) \right)$;

Update Σ:
$\theta_{current} \leftarrow (\mu^{(n+1)}, \Sigma^{(n)}, \beta^{(n)})$;
Compute $\nabla^*_\Sigma \hat{D}(\theta_{current})$;
$\eta_\Sigma \leftarrow$ Armijo-Goldstein$(\mathcal{X}, \theta_{current})$;
$\Sigma^{(n+1)} \leftarrow \mathrm{Exp}_{\Sigma^{(n)}}\left(-\eta_\Sigma \nabla^*_\Sigma \hat{D}(\theta_{current})\right)$;

Update β:
$\theta_{current} \leftarrow (\mu^{(n+1)}, \Sigma^{(n+1)}, \beta^{(n)})$;
Compute $\nabla^*_\beta \hat{D}(\theta_{current})$;
$\eta_\beta \leftarrow$ Armijo-Goldstein$(\mathcal{X}, \theta_{current})$;
$\beta^{(n+1)} \leftarrow \mathrm{Exp}_{\beta^{(n)}}\left(-\eta_\beta \nabla^*_\beta \hat{D}(\theta_{current})\right)$;
end for
$\hat{\theta} \leftarrow (\mu^{(N+1)}), \Sigma^{(N+1)}, \beta^{(N+1)})$;

In this algorithm, the three exponential maps Exp_μ, Exp_Σ and Exp_β are defined in (3.18), and the gradients are computed based on (5.2). The constant η denotes the step-size, which is selected according to the Armijo-Goldstein rule, thanks to a backtracking procedure (Definition 4.2.2. in [2]). The following Proposition 6 states the convergence of Algorithm 1.

In this proposition, θ^* denotes a stationary point of the cost function $\hat{D}(\theta)$ (defined as in (4.6)), and Θ^* a compact neighborhood of θ^*.

Proposition 6 *Assume that θ^* is the unique stationary point of $\hat{D}(\theta)$ in Θ^* and let $(\theta^{(n)})_{n\geqslant 0}$ be a sequence generated by Algorithm 1. If the $(\theta^{(n)})_{n\geqslant 0}$ remain within Θ^*, then $\lim_{n\to\infty} \theta^{(n)} = \theta^*$.*

The proof is given in Appendix B.1.

For the cases $\theta = (\Sigma)$ or $\theta = (\mu, \Sigma)$, the CIM coincides with the FIM (recall Table 3.5). However, it is well know that, near the true value θ^*, the Hessian of the function $\hat{D}(\theta)$ is approximated by the FIM [3]. Therefore, in these two cases, one should expect the $\theta^{(n)}$ to converge to θ^* with a superlinear rate of convergence, just like the Newton method does (Theorem 6.3.2 in [2]).

Precisely, if $\theta = (\Sigma)$ or $\theta = (\mu, \Sigma)$, with a fixed shape parameter β^*, then, under the assumptions of Proposition 6, one should expect Algorithm 1 to generate a sequence $(\theta^{(n)})_{n\geqslant 0}$ converging superlinearly to θ^*. Here, this will not be proved mathematically, but will be observed experimentally in Subsection 7.2.1 (see Figure 7.1).

5.3 CIG Online

The CIG online method is a stochastic gradient method, entirely based on the geometry of the component-wise information metric (CIM), described in Section 3.3. This method aims to minimise the original cost function in Equation (3.1).

In the CIG online method, for the current estimate $\theta^{(n)} = (\mu^{(n)}, \Sigma^{(n)}, \beta^{(n)})$, its correspond stochastic information gradients are given in (3.21). Accordingly, the expected direction of descent is is equal to 0 at any stationary point θ^* of $D(\theta)$ in (3.1).

As in the classic stochastic gradient descent method, the step-size $\eta^{(n)} = \frac{a}{n}$ is strictly positive, decreasing, and verifies the usual conditions (4.1b).

The choice of the coefficient a requires some attention. Whenever the true shape parameter β^* is known, optimal performance is obtained by taking $a = 1$ (see also Corollary

1, below). However, if β^* is unknown, then a has to be chosen manually, by trial and error. This will be discussed in connection with Proposition 8 and with Figure 7.2 in Section 7.2.1.

The CIG online algorithm can now be stated as follows.

Algorithm 2 CIG online algorithm for ECD estimation

Input: A dataset $\mathcal{X}$, an initialization $\theta^{(0)} = (\mu^{(0)}, \Sigma^{(0)}, \beta^{(0)})$, the coefficient $a > 0$;
Output: The estimate $\hat{\theta}$;

for $n = 0, 1, 2, \cdots, N$ **do**
  $\eta^{(n+1)} \leftarrow \frac{a}{n+1}$;
  Update μ:
  $\theta_{current} \leftarrow (\mu^{(n)}, \Sigma^{(n)}, \beta^{(n)})$;
  Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;
  Compute $\nabla_{\mu}^{*}\ell_p(\theta_{current}; \mathcal{X}_{mb}^{new})$;
  $\mu^{(n+1)} \leftarrow \mathrm{Exp}_{\mu^{(n)}}\left(\eta^{(n+1)}\nabla_{\mu}^{*}\ell_p(\theta_{current}; \mathcal{X}_{mb}^{new})\right)$;

  Update Σ:
  $\theta_{current} \leftarrow (\mu^{(n+1)}, \Sigma^{(n)}, \beta^{(n)})$;
  Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;
  Compute $\nabla_{\Sigma}^{*}\ell_p(\theta_{current}; \mathcal{X}_{mb}^{new})$;
  $\Sigma^{(n+1)} \leftarrow \mathrm{Exp}_{\Sigma^{(n)}}\left(\eta^{(n+1)}\nabla_{\Sigma}^{*}\ell_p(\theta_{current}; \mathcal{X}_{mb}^{new})\right)$;

  Update β:
  $\theta_{current} \leftarrow (\mu^{(n+1)}, \Sigma^{(n+1)}, \beta^{(n)})$;
  Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;
  Compute $\nabla_{\beta}^{*}\ell_p(\theta_{current}; \mathcal{X}_{mb}^{new})$;
  $\beta^{(n+1)} \leftarrow \mathrm{Exp}_{\beta^{(n)}}\left(\eta^{(n+1)}\nabla_{\beta}^{*}\ell_p(\theta_{current}; \mathcal{X}_{mb}^{new})\right)$;
end for
$\hat{\theta} \leftarrow (\mu_{N+1}, \Sigma_{N+1}, \beta_{N+1})$;

Here, the three exponential maps Exp_{μ}, Exp_{Σ} and Exp_{β} are defined in Equation (3.18), and the stochastic gradients are computed based on (3.21). In each update, $\mathcal{X}_{mb}^{new}$ could be just one sample, or a mini-batch. If $\mathcal{X}_{mb}^{new}$ is a mini-batch (with more than one sample), then an empirical average over this mini-batch should be introduced, for the stochastic gradients.

The following proposition 7 states the convergence of Algorithm 2. In this proposition, θ^* is a stationary point of $D(\theta)$ in (3.1), and Θ^* is a compact neighborhood of θ^*, which verify the following assumption.

A1. θ^* is the unique stationary point of $D(\theta)$ in Θ^*. Moreover, the second derivatives $\nabla_{\mu}^{*^2}D(\theta^*)$, $\nabla_{\Sigma}^{*^2}D(\theta^*)$, $\nabla_{\beta}^{*^2}D(\theta^*)$ are all positive-definite.

Proposition 7 *Assume that A1 is verified and that the estimates $(\theta^{(n)})_{n\geqslant 0}$ generated by Algorithm 2 remain within Θ^*. Then, $\lim \theta^{(n)} = \theta^*$ almost surely.*

The proof of this convergence is given in Appendix B.2.

In addition to almost-sure convergence, stated in Proposition 7, we have also studied the convergence rate and asymptotic normality of Algorithm 2. Recall Θ^* is a neighborhood of θ^* which satisfies the conditions in Proposition 7. This neighborhood Θ^* admits a

system of normal coordinates $(\theta^i; i = 1, \cdots, d)$ with origin at θ^*, where d is the dimension of the parameter space Θ, $d = \frac{m(m+1)}{2} + m + 1$. Since $D(\theta)$ has an isolated stationary point at $\theta = \theta^*$, its Hessian at this point $\theta = \theta^*$ can be expressed in normal coordinates

$$\mathcal{H}_{ij} = \left.\frac{\partial^2 D}{\partial\theta^i \partial\theta^j}\right|_{\theta^i=0} \tag{5.3}$$

When θ^* is a local minimum of $D(\theta)$, the matrix $\mathcal{H} = (\mathcal{H}_{ij})$ is positive-definite [2]. With these notations, the rate of convergence is given by the following Proposition 8.

Recall that the step-sizes in Algorithm 2 are of the form $\eta^{(n)} = a/n$, where the constant a is chosen by the user.

Proposition 8 *Under the assumptions of Proposition 7, if $a > \frac{1}{2\lambda}$, where $\lambda > 0$ is the smallest eigenvalue of $\mathcal{H}$,*

$$\mathbb{E}\left[d_\odot^2(\theta^*, \theta^{(n)})\right] = \mathcal{O}(n^{-1}) \tag{5.4}$$

Here, $d_\odot(\cdot,\cdot)$ stands for the CID in (3.19), and the "big O" notation means that there exist $K > 0$ and $n_0 > 0$ such that

$$\forall n \geqslant n_0 \qquad \mathbb{E}[d_\odot^2(\theta^*, \theta^{(n)})] \leqslant \frac{K}{n}$$

In terms of the normal coordinates (θ^i), let the component-wise information gradient $\nabla_\theta^\odot \ell_p(\theta^*; x)$ at the point $\theta = \theta^*$, given by (3.21), have components $(u^i(\theta^*))$. Let $\mathcal{G}^* = (\mathcal{G}^*_{ij})$, be the matrix

$$\mathcal{G}^*_{ij} = \mathbb{E}_{\theta^*}\left[u^i(\theta^*)u^j(\theta^*)\right] \tag{5.5}$$

Then, the following proposition gives the asymptotic normality of the CIG online algorithm.

Proposition 9 (asymptotic normality) *Under the assumptions of Propositions 7 and 8, the distribution of the re-scaled coordinates $(n^{\frac{1}{2}}\theta^i)_{i\in\{1,\cdots,d\}}$ converges to a centred $d-$variate normal distribution, where d is the dimension of Θ, with covariance matrix $\mathcal{G}$ given by the following Lyabunov equation*

$$A\mathcal{G} + \mathcal{G}A = -a^2\mathcal{G}^* \tag{5.6}$$

Here, $A = (A_{ij})$ with $A_{ij} = \frac{1}{2}\delta_{ij} - a\mathcal{H}_{ij}$ (δ denotes Kronecker's delta).

The proofs of Propositions 8 and 9 and are discussed in Appendix B.3.

For the cases $\theta = (\Sigma)$ or $\theta = (\mu, \Sigma)$, the CIM coincides with the FIM (recall Table 3.5). Therefore the component-wise information distance (the CID (3.19)) also coincides with the "true" information distance, which is the distance associated with the FIM. Thus, in these cases, Assumptions (d1) to (u1) in Proposition 5 of the previous Chapter 4 are satisfied, and the following corollary may be obtained [88].

Corollary 1 *For the ECD model, parameterised by $\theta = (\Sigma)$ or $\theta = (\mu, \Sigma)$, with a fixed β^*, the CIM (3.16) coincides with the Fisher information metric.*

1. *the rate in equation (5.4) holds, whenever $a > 1/2$.*
2. *if $a = 1$ the distribution of the re-scaled coordinates $(n^{1/2}\theta^i)$ converges to a centred d-variate normal distribution, with covariance matrix equal to the identity $\mathcal{G}^* = I_d$, and the recursive estimates $\theta^{(n)}$ are asymptotically efficient.*

Note that, Item 2) of Corollary 1 implies that the distribution of $nd_*^2(\theta^*, \theta^{(n)})$ converges to a χ^2-distribution with d degrees of freedom.

$$nd_*^2(\theta^*, \theta^{(n)}) \xrightarrow{dist} \chi^2\left(\frac{m(m+1)}{2}\right) \text{ for } \theta = (\Sigma)$$

$$nd_*^2(\theta^*, \theta^{(n)}) \xrightarrow{dist} \chi^2\left(\frac{m(m+1)}{2} + m\right) \text{ for } \theta = (\mu, \Sigma)$$

This provides a practical means of confirming the asymptotic normality of the estimators $\theta^{(n)}$.

5.4 Global convergence analysis

This section studies the global convergence of the CIG algorithm, in both its offline and online versions. The term "global convergence" means that the iterates $(\theta^{(n)})_{n\geq 0}$, generated by Algorithm 1 or Algorithm 2, always converge to the true parameter value θ^*, whatever the initial guess $\theta^{(0)}$.

We will mainly consider the two most well-known sub-families of ECD, the Multivariate Generalized Gaussian Distribution (MGGD) and Multivariate Student t-Distribution, and only in the cases $\theta = (\Sigma)$ and $\theta = (\mu, \Sigma)$. The main results are stated in the following two tables. In these tables, the shape parameter β is considered fixed and known.

	MGGD
$\theta = (\Sigma)$	global convergence holds for $\beta > 0$
$\theta = (\mu, \Sigma)$	global convergence holds for $\beta > 0$

Table 5.1: Global convergence analysis: MGGD

	Student
$\theta = (\Sigma)$	global convergence holds for $\beta > 0$
$\theta = (\mu, \Sigma)$	global convergence holds $\beta > 0$

Table 5.2: Global convergence analysis: Student T

For the cases indicated in Tables 5.1 and 5.2, the cost function ($\hat{D}(\theta)$ for the offline version, and $D(\theta)$ for the online version) has a unique stationary point at the true parameter θ^*, which is the global minimizer. This follows from the strict g-convexity (*i.e.* geodesic convexity of this cost function), which will now be considered.

First, for the case of $\theta = (\Sigma)$ with known μ^* and β^*, let

$$\mathfrak{f}(\delta, \beta) = \frac{1}{g_\beta(\delta)} \tag{5.7}$$

then, for the MGGD model

$$\mathfrak{f}(\delta, \beta) = \exp\left(\frac{1}{2}\delta^\beta\right) \qquad \beta > 0 \tag{5.8a}$$

and for the Student t model,

$$\mathfrak{f}(\delta,\beta)=\left(1+\frac{\delta}{\beta}\right)^{\frac{\beta+m}{2}} \qquad \beta>0 \tag{5.8b}$$

The following proposition introduces a sufficient condition for the Kullback-Leibler divergence $D(\Sigma)$ and its empirical approximation $\hat{D}(\Sigma)$ to be geodesically strictly convex.

Proposition 10 *Assume that the function $\mathfrak{f}:\mathbb{R}_+\to\mathbb{R}_+$ in (5.7) verifies the following condition : for any function $\varphi:\mathbb{R}\to\mathbb{R}_+$*

$$\varphi \text{ is strictly log-convex } \Rightarrow \mathfrak{f}\circ\varphi \text{ is strictly log-convex} \tag{5.9}$$

Then, the Kullback-Leibler divergence $D(\Sigma)$ and the empirical divergence $\hat{D}(\Sigma)$ are geodesically strictly convex.

In particular, when the conclusion of Proposition 10 holds, the unique global minimum of $D(\Sigma)$ is at the true Σ^*. Moreover, Σ^* is also the unique stationary point of $D(\Sigma)$.

Proposition 10 directly yields the following corollary, for the MGGD and Student T models. By plugging (5.8a) and (5.8b) into (5.9), it is easy to obtain the following. Compared with [87], this proposition can give a wider range of shape (or degree of freedom) parameters, for geodesic convexity.

Corollary 2 *the Kullback-Leibler divergence $D(\Sigma)$ and its empirical approximation $\hat{D}(\theta)$ are geodesically strictly convex, with unique global minimum (and unique stationary point), in both of the following cases.*

1. *the dataset $\mathcal{X}$ is distributed according to an MGGD model, with scatter matrix Σ^* and with shape parameter $\beta>0$.*
2. *the dataset $\mathcal{X}$ is distributed according to a Student T model, with scatter matrix Σ^* and degree of freedom $\beta>0$.*

Thus, when Σ is unknown and β satisfies the conditions of Corollary 2, this corollary implies the global convergence of Algorithms 1 and 2. Precisely, these algorithms will always converge to the true value $\theta^*=(\Sigma^*)$ of the parameter $\theta=(\Sigma)$, whatever the initial guess $\theta^{(0)}$.
For the more complicated situation $\theta=(\mu,\Sigma)$, global convergence does not always hold. The cost function $D(\theta)$ is not geodesically convex, but may be reformulated, using a new matrix argument [50].

$$\mathcal{S}=\begin{bmatrix}\Sigma+\mu\mu^\dagger & \mu\\ \mu^\dagger & 1\end{bmatrix}$$

Precisely, if the new random vector y is given by

$$y^\dagger=\left(x^\dagger,1\right)^\dagger$$

then the cost function can be reformulated as

$$\tilde{D}(\theta)=-\frac{1}{2}\log\det(\mathcal{S})-\log\tilde{\mathfrak{f}}(\delta_y)$$

where

$$\delta_y=y^\dagger\mathcal{S}^{-1}y=(x-\mu)^\dagger\Sigma^{-1}(x-\mu)+1$$

Here, the function $\tilde{\mathfrak{f}}$ for MGGD is

$$\tilde{\mathfrak{f}}(\delta_y) = \exp\left[\frac{1}{2}\left(\delta_y - 1\right)^{\beta}\right]$$

and for Student T-distributions

$$\tilde{\mathfrak{f}}(\delta_y) = \left(1 - \frac{1}{\beta} + \frac{\delta_y}{\beta}\right)^{\frac{\beta+m}{2}} \tag{5.10}$$

In [50], the minimization of $\tilde{D}(\theta)$ was proven to be equivalent to the minimization of $D(\theta)$. Replacing the new function $\tilde{\mathfrak{f}}$ into (5.9), the following corollary is obtained.

Corollary 3 *The KL divergence $D(\mu, \Sigma)$, and the empirical divergence $\hat{D}(\mu, \Sigma)$, both have a unique global minimum (and unique stationary point) at (μ^*, Σ^*), in the following cases*

1. *the dataset $\mathcal{X}$ is distributed according to an MGGD model, with expectation and scatter matrix (μ^*, Σ^*) and with fixed shape parameter $\beta > 0$.*
2. *the dataset $\mathcal{X}$ is distributed according to a Student t model, with expectation and scatter matrix (μ^*, Σ^*) and with the fixed degree of freedom $\beta > 0$.*

For these two cases, global convergence is then guaranteed.

Finally, for the most complicated case $\theta = (\mu, \Sigma, \beta)$, the cost function is always non-convex. Moreover, we have verified experimentally that it has multiple stationary points in the parameter space $\Theta = \mathbb{R}^m \times \mathcal{P}_m \times \mathbb{R}_+$. Therefore, the iterates $(\theta^{(n)})_{n\geq 0}$ generated by Algorithms 1 and 2 cannot be guaranteed to converge to the true parameter θ^*, unless the initial guess $\theta^{(0)}$ is close enough to θ^*.

5.5 Conclusion

This chapter introduced the CIG methods, which has two versions, an offline (batch) version and an online (stochastic) version. Both versions aim to estimate the complete ECD model, where all the parameters are unknown, including the shape parameter β.

Propositions 6 and 7 state the convergence of the offline and online CIG methods, respectively. For Proposition 7, the assumption A1 is required.

The online CIG method retains some of the good properties of the Riemannian information gradient method, seen in the previous Chapter 4. For example, Propositions 8 and 9 state the fast convergence and asymptotic normality of this method, provided the step-size $\eta^{(n+1)} = \frac{a}{n+1}$ is chosen correctly, with $a > \frac{1}{2\lambda}$ as explain in Proposition 8.

In general, when the shape parameter β is unknown, this choice of step-size is carried-out manually, and may be very time consuming. On the other hand, when β is known, Corollary 1, after Proposition 9, shows that it is possible to recover all of the good properties of the Riemannian information gradient method, by choosing $\eta^{(n+1)} = \frac{1}{n+1}$. In particular, the online CIG method even turns out to be asymptotically efficient.

In fact, when β is known, there is also a notable improvement in the performance of the offline CIG method, which turns out to have a super-linear rate of convergence.

For the MGGD and Student T models, the global convergence of the CIG methods was considered in Section 5.4, and it summarized in Tables 5.1 and 5.2. For the cases

indicated in these tables, the CIG method converges to the true parameter $\theta^* = (\mu^*, \Sigma^*)$ independently of the initial guess $\theta^{(0)}$.

All of the results obtained in this chapter will be illustrated by computor experiments in Chapter 7. The following Chapter 6 will extend the online CIG method to the estimation of mixtures of ECD (MECD).

Chapter 6

Online estimation of MECD

6.1 Introduction

The main body of this chapter is based on our article [89], which is currently under review.

An MECD model is a mixture (*i.e.* a weighted sum) of K ECDs, where K is called the number of components. The dimension of its parameter space can be quite large, as it is more than K times the dimension of the parameter space of a single ECD model. In addition, MECD are typically used to model more complex, large-scale datasets. Accordingly, traditional offline methods may become impractical when applied to the estimation of MECD, due to lack of computational resources (time and memory).

Therefore, it seems more suitable to use online (stochastic) methods. However, Euclidean stochastic gradient methods may need increasingly large mini-batch sizes, to overcome instability [38]. Besides, choosing optimal step-sizes, for stochastic gradient methods, is a difficult task, especially for the estimation of MECD.

The present chapter proposes a different approach, by applying the CIG online method to the estimation of MECD. This extends the previous Chapter 5, which focused on estimating a single ECD, rather than a full mixture.

The geometric background, needed to introduce the CIG online method for MECD estimation, was given in Section 3.4 of Chapter 3. This includes the component-wise information metric (CIM) and component-wise information gradient (CIG) for MECD, respectively given by (3.23) and (3.28).

The CIG online method for the estimation of MECD has two versions, a decreasing step-size version (CIG-DS) and an adaptive step-size version (CIG-AS). The decreasing step-size version is a direct extension of the CIG online method of the previous chapter, Algorithm 2. It can be obtained by introducing the retraction map (3.27) and CIG (3.28) of MECD models, instead of the retraction map (3.18) and CIG (3.21) of ECD models.

This decreasing step-size version requires manual selection of the step-sizes, according to a rule similar to the one in Proposition 8 of the previous Chapter 5, for single ECD. This is not always practical for MECD.

The adaptive step-size version involves an automatic selection of the step-sizes, which guarantees a fast rate of convergence. Automatic selection of the step-sizes is very helpful in saving time, since a manual selection involves running the algorithm several times, selecting a suitable step-size by trial and error.

The CIG method turns out to have several significant advantages. First, being an online (i.e. stochastic) method, each iteration requires access to only one mini-batch, of a constant size, instead of the whole dataset (or mini-batch with increasing size). This considerably reduces the required time and memory usage. Second, when shape parameters are unknown, it still converges to accurate estimates, without increasing the size of mini-batches (compare to [38]). Third, the CIG-AS variant (with an adaptive step-size) avoids manual step-size selection.

In the following, Section 6.2 presents the CIG-DS method, as well as a short description of another method, which is an online backtracking method, called CIG-OB. In Section 6.3, the CIG-AS method is stated in detail. Here, the theoretical convergence properties, of the CIG-DS and CIG-AS methods, will be stated in Propositions 11 and 12, respectively. In the following Chapter 7, these properties will be evaluated by numerical experiments, in Section 7.2.2.

6.2 CIG-DS

The CIG-DS method uses the most common decreasing step-size, $\eta^{(n)} = \frac{a}{n}$ with $a > 0$, in order to verify the usual stochastic approximation Conditions (4.1b). The constant a should be selected in advance, before running the algorithm, (just as in Algorithm 2). The choice of this constant has a crucial influence on the rate of convergence.

Recall here the MECD density (1.2) is parameterised by $\boldsymbol{\theta} = (r, (\mu_k)_k, (\Sigma_k)_k, (\beta_k)_k)$ where $k = 1, \dots, K$, and $r = (r_k)_k$ is related to the mixture weights $w = (w_k)_k$ by $w_k = r_k^2$.

Algorithm 3 CIG-DS algorithm for MECD

Input: A dataset $\mathcal{X}$, an initialization $\boldsymbol{\theta}^{(0)} = \left(r^{(0)}, (\mu_k^{(0)})_k, (\Sigma_k^{(0)})_k, (\beta_k^{(0)})_k\right)$, a constant $a > 0$;

Output: The estimate $\hat{\boldsymbol{\theta}}$;

for $n = 0, 1, 2, \cdots, N$ **do**
  $\eta^{(n+1)} \leftarrow \frac{a}{n+1}$;
  Update r**:**
  $\boldsymbol{\theta}_{current} \leftarrow \left(r^{(n)}, (\mu_k^{(n)})_k, (\Sigma_k^{(n)})_k, (\beta_k^{(n)})_k\right)$;
  Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;
  Compute $\nabla_r^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})$;
  $r^{(n+1)} \leftarrow \mathrm{Exp}_{r^{(n)}}\left(\eta^{(n+1)} \nabla_r^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})\right)$;
  Update μ**:**
  $\boldsymbol{\theta}_{current} \leftarrow \left(r^{(n+1)}, (\mu_k^{(n)})_k, (\Sigma_k^{(n)})_k, (\beta_k^{(n)})_k\right)$;
  for $k = 1, 2, \cdots, K$ **do**
    Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;
    Compute $\nabla_{\mu_k}^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})$;
    $\mu_k^{(n+1)} \leftarrow \mathrm{Exp}_{\mu_k^{(n)}}\left(\eta^{(n+1)} \nabla_{\mu_k}^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})\right)$;
  end for

Update Σ:
$\boldsymbol{\theta}_{current} \leftarrow \left(r^{(n+1)}, (\mu_k^{(n+1)})_k, (\Sigma_k^{(n)})_k, (\beta_k^{(n)})_k\right)$;
for $k = 1, 2, \cdots, K$ **do**
  Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;
  Compute $\nabla^*_{\Sigma_k} \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})$;
  $\Sigma_k^{(n+1)} \leftarrow \mathrm{Exp}_{\Sigma_k^{(n)}} \left(\eta^{(n+1)} \nabla^*_{\Sigma_k} \ell_f \left(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new}\right)\right)$;
end for

Update β:
$\boldsymbol{\theta}_{current} \leftarrow \left(r^{(n+1)}, (\mu_k^{(n+1)})_k, (\Sigma_k^{(n+1)})_k, (\beta_k^{(n)})_k\right)$;
for $k = 1, 2, \cdots, K$ **do**
  Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;
  Compute $\nabla^*_{\beta_k} \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})$;
  $\beta_k^{(n+1)} \leftarrow \mathrm{Exp}_{\beta_k^{(n)}} \left(\eta^{(n+1)} \nabla^*_{\beta_k} \ell_f \left(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new}\right)\right)$;
end for
end for
$\hat{\boldsymbol{\theta}} \leftarrow \left(r^{(N+1)}, (\mu_k^{(N+1)})_k, (\Sigma_k^{(N+1)})_k, (\beta_k^{(N+1)})_k\right)$

In the above algorithm, all the exponential maps are given in (3.27), and all the gradients are computed using (3.28) and the formulas just after.

The following proposition gives the convergence and rate of convergence for Algorithm 3. In this proposition, $D(\boldsymbol{\theta})$ is the Kullback-Leibler divergence (3.3).

Proposition 11 (Convergence rate of CIG-DS) *Assume the Kullback-Leibler divergence function $D(\boldsymbol{\theta})$ has a stationary point at $\boldsymbol{\theta} = \boldsymbol{\theta}^*$, $\boldsymbol{\Theta}^* \subset \boldsymbol{\Theta}$ is a compact and convex neighborhood of $\boldsymbol{\theta}^*$, such that $\boldsymbol{\theta}^*$ is the unique stationary point of $D(\boldsymbol{\theta})$ in $\boldsymbol{\Theta}^*$, and the estimates $(\boldsymbol{\theta}^{(n)})_{n \geqslant 0}$, generated by Algorithm 3, remain within $\boldsymbol{\Theta}^*$. Then, for the sequence $(\boldsymbol{\theta}^{(n)})_{n \geqslant 0}$,*

$$\lim_{n \to \infty} \boldsymbol{\theta}^{(n)} = \boldsymbol{\theta}^*$$

If $a > \frac{1}{2\lambda}$, where $\lambda > 0$ is the smallest eigenvalue of the Hessian matrix of $D(\boldsymbol{\theta})$ at the point $\boldsymbol{\theta} = \boldsymbol{\theta}^*$, then

$$D(\boldsymbol{\theta}^{(n)}) - D(\boldsymbol{\theta}^*) = \mathcal{O}(n^{-1}) \tag{6.1}$$

Here, the "big O" notation means that there exist $C > 0$ and $n_0 > 0$ such that

$$\forall n \geqslant n_0 \qquad D(\boldsymbol{\theta}^{(n)}) - D(\boldsymbol{\theta}^*) \leqslant \frac{C}{n}$$

The proof of this proposition is given in Appendix C.1. In Algorithm 3, $\mathcal{X}_{mb}^{new}$ denotes a mini-batch of samples drawn at random from the complete dataset $\mathcal{X}$. Theoretically, the optimal value for a can be selected based on the condition $a > \frac{1}{2\lambda}$. In practice, it is very difficult to do so, and a blindly selected a may not guarantee the convergence rate in equation (6.1). One way of overcoming this issue is to introduce an online backtracking line-search technique [80]. When applied to the CIG method, this leads to the following CIG-OB (online backtracking) version.

Algorithm 4 CIG-OB algorithm for MECD

Input: A dataset $\mathcal{X}$, an initialization $\boldsymbol{\theta}^{(0)} = \left(r^{(0)}, (\mu_k^{(0)})_k, (\Sigma_k^{(0)})_k, (\beta_k^{(0)})_k\right)$;

Output: The estimate $\hat{\boldsymbol{\theta}}$;

for $n = 0, 1, 2 \cdots, N$ **do**
  Update r**:**
  $\boldsymbol{\theta}_{current} \leftarrow \left(r^{(n)}, (\mu_k^{(n)})_k, (\Sigma_k^{(n)})_k, (\beta_k^{(n)})_k\right)$;
  Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;
  Compute $\nabla_r^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})$;
  $\eta_r \leftarrow$ Armijo-Goldstein$(\mathcal{X}_{mb}^{new}, \boldsymbol{\theta}_{current})$;
  $r^{(n+1)} \leftarrow \mathrm{Exp}_{r^{(n)}}\left(\eta_r \nabla_r^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})\right)$;

  Update μ**:**
  $\boldsymbol{\theta}_{current} \leftarrow \left(r^{(n+1)}, (\mu_k^{(n)})_k, (\Sigma_k^{(n)})_k, (\beta_k^{(n)})_k\right)$;
  for $k = 1, 2, \cdots, K$ **do**
    Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;
    Compute $\nabla_{\mu_k}^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})$;
    $\eta_{\mu_k} \leftarrow$ Armijo-Goldstein$(\mathcal{X}_{mb}^{new}, \boldsymbol{\theta}_{current})$;
    $\mu_k^{(n+1)} \leftarrow \mathrm{Exp}_{\mu_k^{(n)}}\left(\eta_{\mu_k} \nabla_{\mu_k} \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})\right)$;
  end for

  Update Σ**:**
  $\boldsymbol{\theta}_{current} \leftarrow \left(r^{(n+1)}, (\mu_k^{(n+1)})_k, (\Sigma_k^{(n)})_k, (\beta_k^{(n)})_k\right)$;
  for $k = 1, 2, \cdots, K$ **do**
    Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;
    Compute $\nabla_{\Sigma_k}^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})$;
    $\eta_{\Sigma_k} \leftarrow$ Armijo-Goldstein$(\mathcal{X}_{mb}^{new}, \boldsymbol{\theta}_{current})$;
    $\Sigma_k^{(n+1)} \leftarrow \mathrm{Exp}_{\Sigma_k^{(n)}}\left(\eta_{\Sigma_k} \nabla_{\Sigma_k} \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})\right)$;
  end for

  Update β**:**
  $\boldsymbol{\theta}_{current} \leftarrow \left(r^{(n+1)}, (\mu_k^{(n+1)})_k, (\Sigma_k^{(n+1)})_k, (\beta_k^{(n)})_k\right)$;
  for $k = 1, 2, \cdots, K$ **do**
    Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;
    Compute $\nabla_{\beta_k}^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})$;
    $\eta_{\beta_k} \leftarrow$ Armijo-Goldstein$(\mathcal{X}_{mb}^{new}, \boldsymbol{\theta}_{current})$;
    $\beta_k^{(n+1)} \leftarrow \mathrm{Exp}_{\beta_k^{(n)}}\left(\eta_{\beta_k} \nabla_{\beta_k} \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})\right)$;
  end for
end for
$\hat{\boldsymbol{\theta}} \leftarrow \left(r^{(N+1)}, (\mu_k^{(N+1)})_k, (\Sigma_k^{(N+1)})_k, (\beta_k^{(N+1)})_k\right)$

In the algorithm above, Armijo-Goldstein$(\cdot, \cdot)$ means the Armijo-Goldstein backtracking criterion [2, 80], for which one needs to provide the current estimator and a mini-batch. The CIG-OB method inherits the properties of classic deterministic line-search methods. With respect to number of iterations, it has a fast rate of convergence, but the actual time consumption is more expensive than CIG-DS. Moreover, the size of the mini-batch significantly affects its accuracy, this will be discussed in detail, in Section 7.2.2.

6.3 CIG-AS

The CIG-AS method is similar in structure to the CIG-DS method. The main difference between the two methods is the following. In CIG-DS, the step-size $\eta^{(n)}$ is chosen in advance (for example, $\eta^{(n)} = \frac{a}{n}$). On the other hand, in CIG-AS, $\eta^{(n)}$ is computed based on the current estimate $\boldsymbol{\theta}^{(n)}$, in order to ensure a faster rate of convergence. In other words, CIG-AS involves an adaptive choice of the step-size $\eta^{(n)}$.

The idea of introducing an adaptive step-size is loosely based on [54], which deals with classical (Euclidean) gradient descent.

All the steps of CIG-AS are similar to algorithm 3, except the step-size is adaptive. This is reflected in the following algorithm (in this algorithm, the adaptive step-size $\eta^{(n)}$ is set in line 2).

Algorithm 5 CIG-AS algorithm for MECD

Input: A dataset $\mathcal{X}$, an initialization $\boldsymbol{\theta}^{(0)} = \Big(r^{(0)}, (\mu_k^{(0)})_k, (\Sigma_k^{(0)})_k, (\beta_k^{(0)})_k\Big)$;

Output: The estimate $\hat{\boldsymbol{\theta}}$;

1: **for** $n = 0, 1, 2 \cdots$ **do**

2: $\quad \eta^{(n+1)} \leftarrow \frac{\tau_{min}^{(n)}}{\rho' \, L \, \tau_{max}^{(n)}}$;

3: $\quad$ **Update** r**:**

4: $\quad \boldsymbol{\theta}_{current} \leftarrow \Big(r^{(n)}, (\mu_k^{(n)})_k, (\Sigma_k^{(n)})_k, (\beta_k^{(n)})_k\Big)$;

5: $\quad$ Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;

6: $\quad$ Compute $\nabla_r^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})$;

7: $\quad r^{(n+1)} \leftarrow \mathrm{Exp}_{r^{(n)}} \left(\eta^{(n+1)} \nabla_r^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})\right)$;

8: $\quad$ **Update** μ**:**

9: $\quad \boldsymbol{\theta}_{current} \leftarrow \Big(r^{(n+1)}, (\mu_k^{(n)})_k, (\Sigma_k^{(n)})_k, (\beta_k^{(n)})_k\Big)$;

10: $\quad$ **for** $k = 1, 2, \cdots, K$ **do**

11: $\quad\quad$ Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;

12: $\quad\quad$ Compute $\nabla_{\mu_k}^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})$;

13: $\quad\quad \mu_k^{(n+1)} \leftarrow \mathrm{Exp}_{\mu_k^{(n)}} \left(\eta^{(n+1)} \nabla_{\mu_k}^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})\right)$;

14: $\quad$ **end for**

15: $\quad$ **Update** Σ**:**

16: $\quad \boldsymbol{\theta}_{current} \leftarrow \Big(r^{(n+1)}, (\mu_k^{(n+1)})_k, (\Sigma_k^{(n)})_k, (\beta_k^{(n)})_k\Big)$;

17: $\quad$ **for** $k = 1, 2, \cdots, K$ **do**

18: $\quad\quad$ Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;

19: $\quad\quad$ Compute $\nabla_{\Sigma_k}^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})$;

20: $\quad\quad \Sigma_k^{(n+1)} \leftarrow \mathrm{Exp}_{\Sigma_k^{(n)}} \left(\eta^{(n+1)} \nabla_{\Sigma_k}^* \ell_f \left(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new}\right)\right)$;

21: $\quad$ **end for**

22: $\quad$ **Update** β**:**

23: $\quad \boldsymbol{\theta}_{current} \leftarrow \Big(r^{(n+1)}, (\mu_k^{(n+1)})_k, (\Sigma_k^{(n+1)})_k, (\beta_k^{(n)})_k\Big)$;

24: $\quad$ **for** $k = 1, 2, \cdots, K$ **do**

25: $\quad\quad$ Pick a new mini-batch $\mathcal{X}_{mb}^{new}$;

26: $\quad\quad$ Compute $\nabla_{\beta_k}^* \ell_f(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new})$;

27: $\quad\quad \beta_k^{(n+1)} \leftarrow \mathrm{Exp}_{\beta_k^{(n)}}\left(\eta^{(n+1)} \nabla^*_{\beta_k} \ell_f\left(\boldsymbol{\theta}_{current}; \mathcal{X}_{mb}^{new}\right)\right)$;
28: **end for**
29: **end for**
30: $\hat{\boldsymbol{\theta}} \leftarrow \left(r^{(N+1)}, (\mu_k^{(N+1)})_k, (\Sigma_k^{(N+1)})_k, (\beta_k^{(N+1)})_k\right)$

In the algorithm above,

$$\tau_{min}^{(n)} = \min\left\{1, \left(\frac{\lambda_{min,k}^{(n)}}{I_{\mu_k}^{(n)}}\right)_k, \left(J_{\Sigma_k,2}^{(n)}\right)_k, \left(\frac{1}{I_{\beta_k}^{(n)}}\right)_k\right\}$$

$$\tau_{max}^{(n)} = \max\left\{1, \left(\frac{\lambda_{max,k}^{(n)}}{I_{\mu_k}^{(n)}}\right)_k^2, \left(J_{\Sigma_k,1}^{(n)}\right)_k^2, \left(\frac{1}{I_{\beta_k}^{(n)}}\right)_k^2\right\}$$

where $\lambda_{min,k}$ is the smallest eigenvalue of Σ_k, and $\lambda_{max,k}$ is the biggest eigenvalue of Σ_k. Moreover, the coefficients I_{μ_k}, $J_{\Sigma_k,1}$, $J_{\Sigma_k,2}$, and I_{β_k} are computed from (2.10) and (3.25).

The following proposition gives the convergence of this algorithm. The meaning of strong geodesic convexity, for Condition (i), was explained in Definition 1, of Chapter 2. The definitions of Conditions (ii) and (iii) can be found in Equation (C.7) and Equation (C.20), of Appendix C.2.

The L-geodesically smooth assumption gives a sequence that upper bounds the error $\mathbb{E}_{\mathcal{X}_{new}}[D(\theta^{(n+1)})] - D(\theta^*)$. Then, this sequence is proved to be convergent to 0 by employing the α-geodesically strongly convexity and ρ-strong growth condition. The step-size which convient to this convergence condition is under the following form

$$\eta^{(n+1)} = \frac{\tau_{min}^{(n)}}{\rho' \, L \, \tau_{max}^{(n)}}$$

Proposition 12 (Linear convergence of CIG-AS) *Assume that $\boldsymbol{\theta}^*$ is the unique stationary point of the Kullback-Leibler divergence $D(\boldsymbol{\theta})$ in a compact convex neighborhood $\boldsymbol{\Theta}^*$. Assume that $D(\boldsymbol{\theta})$ satisfies the following properties on $\boldsymbol{\Theta}^*$,*

(i) α-geodesically strongly convex [86],

(ii) L-geodesically smooth [86],

(iii) satisfies the ρ-strong growth condition [79].

If $\boldsymbol{\theta}^{(0)}$ belongs to Θ^*, then the sequence generated by Algorithm 5 with the step-size $\eta^{(n)} = \frac{\tau_{min}^{(n)}}{\rho' \, L \, \tau_{max}^{(n)}}$ and a constant batch size b converges to $\boldsymbol{\theta}^*$, with the following rate

$$\mathbb{E}_{\mathcal{X}_{mb}}\left[D\left(\boldsymbol{\theta}^{(N)}\right)\right] - D\left(\boldsymbol{\theta}^*\right) \leqslant c(N)\,\left(D\left(\boldsymbol{\theta}^{(0)}\right) - D\left(\boldsymbol{\theta}^*\right)\right) \tag{6.2}$$

where

$$c(N) = \prod_{n=1}^{N}\left\{1 - \frac{\alpha\left[\tau_{min}^{(n)}\right]^2}{\rho' \, L \, \tau_{max}^{(n)}}\right\}$$

$$\rho' = \frac{(N-b)(\rho-1)}{(N-1)b} + 1$$

Remark that, c is a real positive scalar in $(0,1)$, not the normalizing constant. The constant b is the size of mini-batch, $\mathcal{X}_{mb}$ denotes the law for selection of mini-batch, and the constant N denotes the number of iterations.

The proposition above shows that the convergence rate is linear. Linear convergence means that the following definition holds [71].

Definition 2 *Suppose that the sequence $D(\boldsymbol{\theta}^{(N)})$ converges to the number $D(\boldsymbol{\theta}^*)$. The sequence is said to converge linearly to $D(\boldsymbol{\theta}^*)$ if there exist constants $c, A > 0$ such that*

$$\log\left(D(\boldsymbol{\theta}^{(N)}) - D(\boldsymbol{\theta}^*)\right) \leqslant -cN + A \tag{6.3}$$

Indeed, it can be shown that this definition holds, as a consequence of Proposition 12, simply by taking logarithms in Equation (6.2), and noting that $\log c(N) \leqslant -cN$, where

$$c = \min_{n \geqslant 1} \left\{ \log \left(\frac{1}{1 - \frac{\alpha \left[\tau_{min}^{(n)}\right]^2}{\rho' \, L \, \tau_{max}^{(n)}}} \right) \right\} > 0$$

The fact that $c > 0$ can be proved using the fact that $\theta^{(n)} \to \boldsymbol{\theta}^*$.

Note that, in practice, it may be difficult to select L and ρ. Too big a value of L and ρ will lead to a very slow convergence at early stages of the estimation. Therefore, in numerical experiments and real applications, we used the following step-size, a combination of the CIG-DS and CIG-AS step-sizes,

$$\eta^{(n)} = \max \left\{ \frac{\tau_{min}^{(n)}}{\rho' \, L \, \tau_{max}^{(n)}}, \frac{a}{n+1} \right\} \tag{6.4}$$

The use of this "hybrid" step-size selection rule will be illustrated in Subsection 7.2.2 of the following Chapter 7.

6.4 Conclusion

The present chapter introduced the CIG method for the estimation of MECD. Since MECD models are high-dimensional and typically used to model large datasets, only the online form of the CIG method was developed.

In particular, two new algorithms were proposed, CIG-DS (CIG with decreasing step-size) and CIG-AS (CIG with adaptive step-size), in Sections 6.2 and 6.3 respectively. The rate of convergence of the CIG-DS method was stated in Proposition 11, and the rate of convergence of the CIG-AS method was stated in Proposition 12.

The CIG-AS method carries out an adaptive step-size selection, avoiding the difficult and time-consuming manual step-size selection. It also achieves a faster, linear rate of convergence (in the sense of Definition 2).

The present chapter completes the theoretical part of this thesis. We have now extended CIG estimation methods to the estimation of complete MECD models. In the following chapter, we will present numerical experiments, with simulated and real data, in order to illustrate the performance of CIG methods, theoretically discussed in Chapters 5 and 6.

Chapter 7

Experiments and applications

7.1 Introduction

The present chapter is the final chapter, in the main body of this thesis. It aims to illustrate, through applications to simulated and real data, the various theoretical properties of the CIG estimation methods, for ECD and MECD, as introduced in Chapters 5 and 6.

Section 7.2 is devoted to experiments carried out with simulated data. In this section, Subsection 7.2.1 corresponds to the CIG methods of Chapter 5, applied to the estimation of ECD. Subsection 7.2.2 is devoted to the CIG methods of Chapter 6, applied to the estimation of MECD.

Section 7.3 is concerned with applications to real data. Subsection 7.3.1 applies the CIG offline and online methods, from Chapter 5, to colour transformation for image editing. Subsection 7.3.2 applies the CIG methods of Chapter 6 to texture segmentation.

The results of Section 7.2 confirm the theoretical results in Chapters 5 and 6. In particular, they verify the conclusions of Proposition 8, Corollary 1, and Propositions 11 and 12.

In addition, Section 7.2 demonstrates some properties which were not obtained theoretically in Chapters 5 and 6, such as the super-linear convergence rate of the CIG offline method. It also compares CIG methods to other, state-of-the-art methods, such as method of moments, fixed-point, expectation-maximization, and classical stochastic gradient [82, 59, 50, 57]. The comparison involves both estimation accuracy and computation time.

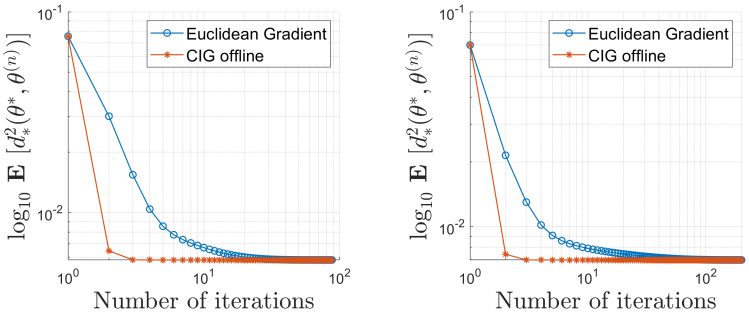
The applications of Section 7.3 showcase all the advantages of CIG online methods, when used on real data. For example (see Subsection 7.3.1), the CIG online method is able to perform the crucial estimation step, for a colour transformation application

to a pair of images, with over 10^6 pixels, in only 21 seconds. For this this application, other state-of-the-art methods require two hours to do the same job, with a comparable performance.

7.2 Computer experiments

7.2.1 ECD

This subsection presents a set of computer experiments, which confirm the theoretical results of Sections 5.2 and 5.3, and provide a detailed comparison of the CIG estimation methods, with the already existing MM and FP [82, 59]. For every experiment, 1000 Monte Carlo trials were carried out. For each trial, the dataset $\mathcal{X} = \{x_1, \cdots, x_T\}$ is independent and identically distributed, according to true parameters $(\mu^*, \Sigma^*, \beta^*)$. The dimension m of x_t is taken equal to 10. The true μ^* is randomly chosen from a multivariate normal distribution. The scatter Σ^* is defined as $\Sigma(i,j) = \rho^{|i-j|}$ for $i, j \in \{1, m\}$, and ρ uniformly distributed in $[0.2, 0.8]$. The shape parameter β^* is uniformly selected from the interval $[0.2, 5]$, for MGGD and for Student t.



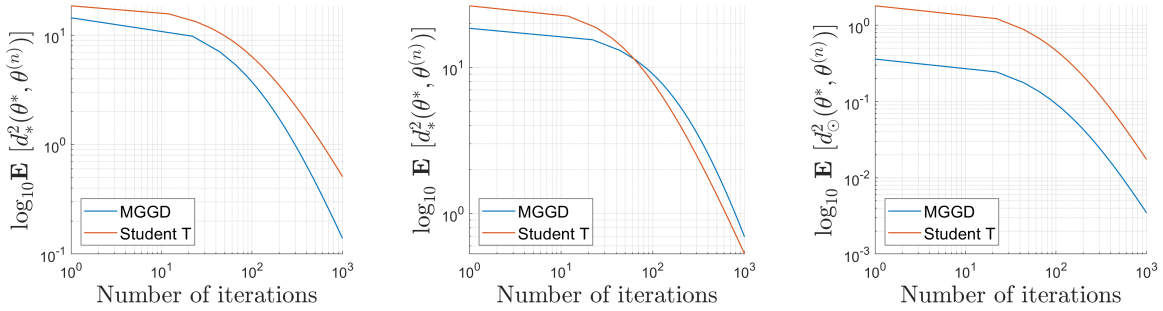
(a) The case $\theta = (\Sigma)$ (b) The case $\theta = (\mu, \Sigma)$

Figure 7.1: ECD: The superlinear convergence rate for CIG offline

The first experiment confirms the super-linear convergence rate of the CIG Offline method, for a dataset, distributed according to the MGGD model, which contains $N = 10^4$ samples. The initial value $\theta^{(0)}$ is defined as the MM estimate, using 10% of the entire dataset. Figure 7.1(a) presents the case of $\theta = (\Sigma)$ with known (μ^*, β^*). The CIG Offline method converges after only two iterations, and if the same accuracy needs to be achieved, the classic deterministic gradient method (not using the information gradient) requires at least 88 iterations. For the case of $\theta = (\mu, \Sigma)$ with known (β^*), things are similar. Figure 7.1(b) shows that CIG Offline method, after two iterations, achieves the same accuracy as the traditional deterministic gradient method, after 200 iterations. Here, the traditional deterministic gradient means the classical (Euclidean) deterministic gradient method, with step-sizes also selected according to the Armijo-Goldstein line search criteria [2].

In Figure 7.1, $d_*^2(\theta^*, \theta^{(n)})$ denotes the information distance, given by Equation (3.13). In Figure 7.2, $d_\odot^2(\theta^*, \theta^{(n)})$ denotes the component-wise information distance, given by Equation (3.19).

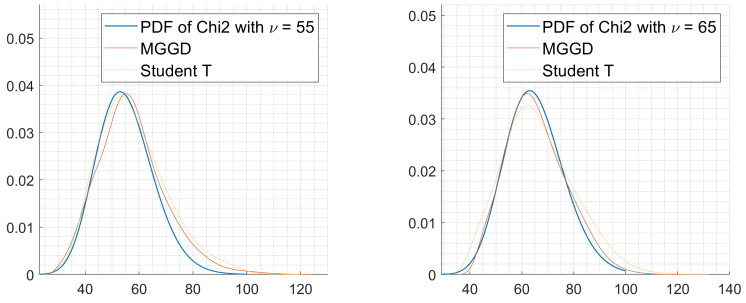
The second experiment demonstrates the convergence rate of the CIG online method. In this experiment, both MGGD and Student T datasets are used. The initialization $\theta^{(0)}$ is randomly chosen. Figures 7.2(a), 7.2(b), and 7.2(c) display the mean-square rate of convergence, stated theoretically in Proposition 8. In these log-log plots, the x-axis



(a) The case $\theta = (\Sigma)$ (b) The case $\theta = (\mu, \Sigma)$ (c) The case $\theta = (\mu, \Sigma, \beta)$

Figure 7.2: ECD: $\mathcal{O}(n^{-1})$ convergence rate for CIG Online method

and y-axis represent the number of iterations and $\mathbb{E}[d^2_{\odot}(\theta^*, \theta^{(n)})]$, respectively, where $\mathbb{E}$ denotes the Monte Carlo approximation of the expectation, obtained by averaging over the 1000 trials. As $\theta^{(n)}$ approaches the true value θ^*, the slope of each curve approaches -1, indicating the mean-square rate $\mathbb{E}[d^2_{\odot}(\theta^*, \theta^{(n)})] = \mathcal{O}(n^{-1})$ in Equation (5.4). Note that, for the cases of $\theta = (\Sigma)$ and $\theta = (\mu, \Sigma)$, the initialization $\theta^{(0)}$ can be chosen far away from $\theta^{(0)}$ (e.g. $d^2_{\odot}(\theta^*, \theta^{(0)}) > 10$). However, when $\theta = (\mu, \Sigma, \beta)$, the initialization should be in a neighbourhood of θ^* (this is according to the conditions stated in Proposition 8). For the results obtained in Figures 7.2(a) and 7.2(b) (that is to say, when β^* is known), the step-size coefficient a always equals 1, but for the case of unknown β, the step-size coefficient a is taken much larger, $a = 100$ (these choices correspond to the conditions of Proposition 8 and Corollary 1).

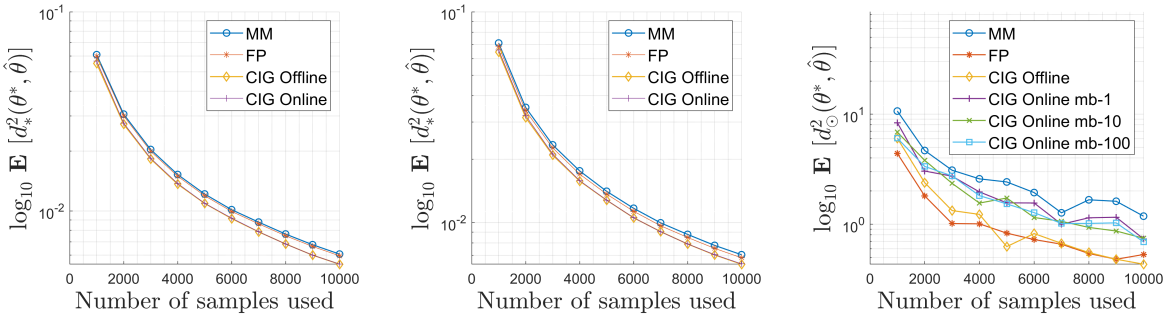


(a) The case $\theta = (\Sigma)$ (b) The case $\theta = (\mu, \Sigma)$

Figure 7.3: ECD: Asymptotic normality of CIG online method

For the case of $\theta = (\Sigma)$ and $\theta = (\mu, \Sigma)$, Figures 7.3(a) and 7.3(b) demonstrate the asymptotic normality of CIG online method, obtained theoretically in Corollary 1. The samples being vectors of dimension $m = 10$, the dashed blue curve is the probability density of a chi-squared distribution with 55 and 65 degrees of freedom, for Figures 7.3(a) and 7.3(b), respectively. The solid lines are the smoothed histograms of $Nd^2_*(\theta^*, \theta^{(N)})$ where $N = 10^5$. These "estimated p.d.f." coincide very closely with the theoretical chi-squared probability density, confirming the fact that $\theta^{(N)}$ is asymptotically normally distributed about θ^*.

In the third experiment, we compare the efficiency of the CIG offline and online methods with other common estimation methods, MM (method of moments) and FP (fixed-point method). In each trial, the dataset is generated from an MGGD model. For MGGD, the MM was given in [82], and the FP method in [59]. In Figures 7.4(a), 7.4(b) and 7.4(c),



(a) The case $\theta = (\Sigma)$ (b) The case $\theta = (\mu, \Sigma)$ (c) The case $\theta = (\mu, \Sigma, \beta)$

Figure 7.4: ECD: Efficiency comparison between MM, FP and CIG methods

the x-axis denotes the size of the dataset, and the y-axis denotes the expectation of the squared distance between θ^* and the final estimate $\hat{\theta}$. This expectation is approximated by the average of 10^3 Monte Carlo trials. For the cases $\theta = (\Sigma)$ and $\theta = (\mu, \Sigma)$, the CIG offline and online algorithms show a better accuracy. When $\theta = (\mu, \Sigma, \beta)$, the accuracy of the MLE methods is still significantly better than MM, and the accuracies of CIG offline and FP coincide. However, the accuracy of CIG online is not as good as as FP or CIG offline. This phenomenon may be explained theoretically. Indeed, when $\theta = (\mu, \Sigma, \beta)$, the component-wise information metric does not coincide with the Fisher information metric of the ECD model, and this leads to a less efficient estimation. The fluctuations of the curves in Figure 7.4(c) are quite significant. This means the variance of the final estimate $\hat{\theta}$ is not negligible. Two additional experiments were carried out, in order to explain these fluctuations.

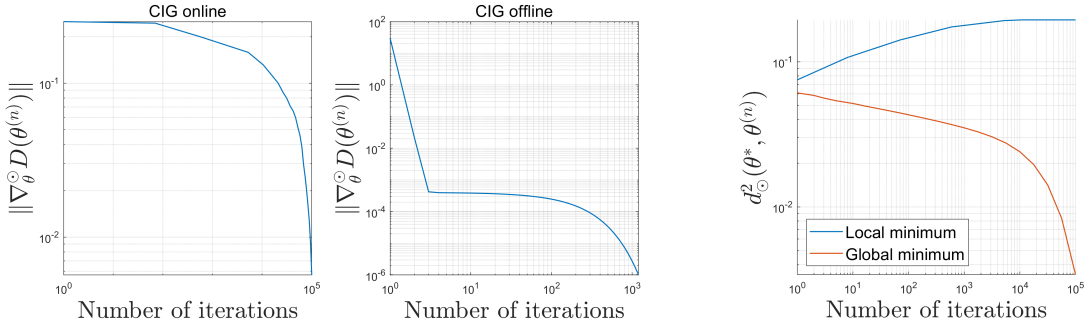
The first additional experiment shows that both the CIG offline and CIG online methods eventually converge to a stationary point. Figure 7.5(a) shows that the norm of the component-wise information gradient of the Kullback-Leibler divergence (3.1) always converge to 0, independently of the initial value $\theta^{(0)}$. Here, this component-wise information gradient is estimated empirically based on the Equation (5.2).

The second additional experiment shows that, for the CIG online method, even tough, the gradient of the Kullback-Leibler divergence converge to 0 (as seen in the first additional experiment), the iterates $\theta^{(n)}$ do not necessarily converge to the global minimum θ^* (the true parameter value). In Figure 7.5(b), two different initial values were used for this method. The blue curve has $\theta^{(0)}$ farther away from θ^*. This curve shows that the component-wise information distance $d_\odot(\theta^*, \theta^{(n)})$ converges to a non-zero constant, meaning that $\theta^{(n)}$ do not converge to θ^*. The red curve has $\theta^{(0)}$ closer to θ^*. In this case, the component-wise information distance $d_\odot(\theta^*, \theta^{(n)})$ converges to 0. Of course, this means $\theta^{(n)}$ converge to θ^*.

In conclusion, for $\theta = (\mu, \Sigma, \beta)$, the convergence to the global minimum θ^* can only be guaranteed locally. For the CIG offline methods, if the initial guess $\theta^{(0)}$ is chosen in a neighbourhood Θ^* of θ^*, the $\theta^{(n)}$ always converge to θ^*, as soon as Θ^* satisfies the assumptions of Proposition 6.

The difference between the CIG online and offline methods is that, due to its stochastic nature, CIG online may jump out of the neighbourhood Θ^* during the first few iterations. This leads to convergence to a local minimum, different from θ^*. This explains why the final averaged accuracy of CIG online is not as good as that of the CIG offline and FP methods, making the variance of the CIG online estimator notably larger.

As a possible remedy to this problem, the mini-batch CIG online was also tested, and compared with other methods, in the Figure 7.4(c). Two sizes of the mini-batch, 10 and 100 samples, were considered. However, the results show that increasing the mini-batch size has no significant effect on the accuracy of CIG online.



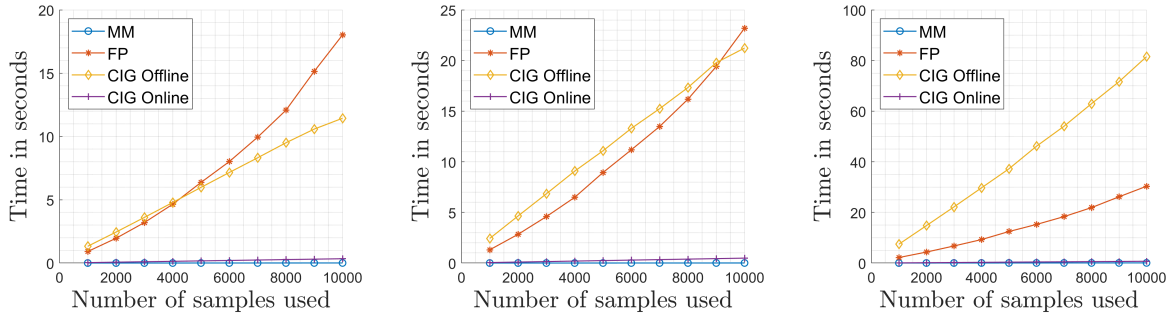
(a) The CIG always converges to 0

(b) The $\theta^{(n)}$ do not always converge to θ^*

Figure 7.5: Complete ECD: existence of local minima

As for computation time, CIG online method has a significant advantage. The computation time of the CIG online algorithm is similar to that of MM, and is significantly less than that of FP. Meanwhile, its accuracy is significantly better than that of MM. In most experiments, the accuracy of CIG online method is similar to, or even better than, FP. Although the computation time of CIG offline is greater than that of CIG online, it is comparable to that of FP, while, in most cases, CIG offline can achieve the best accuracy, among the four estimation methods considered.

Figure 7.6 refers to the same experiment as in Figure 7.4. In Figures 7.6(a), 7.6(b) and 7.6(c), the x-axis denotes the size of the dataset, and the y-axis denotes the computation time necessary to achieve convergence.



(a) The case $\theta = (\Sigma)$

(b) The case $\theta = (\mu, \Sigma)$

(c) The case $\theta = (\mu, \Sigma, \beta)$

Figure 7.6: ECD: Time consumption until convergence

7.2.2 Mixture of ECD

In this subsection, computer simulations are presented, to evaluate the performance of the CIG online method for MECD estimation, as considered in Chapter 6. These simulations will focus on two models, mixtures of MGGD and mixtures of Student T-distriutions.

A comparison of the CIG online method with the already existing EMFP (Expectation-Maximisation with Fixed-Point [59, 37]) and SG (Euclidean Stochastic Gradient [57]) methods is provided.

The dimension of the observed samples (datapoints) is supposed to be $m = 5$ and the number of mixture components $K = 5$. The mixture weights $w = r^2$ are drawn from a uniform distribution on the unit sphere (recall this representation of mixture weights, in terms of the unit sphere, from Subsection 3.4). The true location parameters $(\mu_k^*)_k$ are randomly chosen from a multivariate normal distribution. The true scatter matrices $(\Sigma_k^*)_k$ are defined as $\Sigma(i,j) = \rho^{|i-j|}$ for $i, j \in \{1, \cdots, m\}$, and ρ uniformly distributed in the interval $(0.2, 0.8)$. Then, the shape parameters (degree of freedom for Student T) $(\beta_k^*)_k$ are uniformly chosen in the interval $[0.2, 10]$.

For each scenario, we have done 100 Monte Carlo trials. The dataset for each trial has 3×10^5 observations. For all scenarios, algorithms are initialized by a k-mean with 0.1% of the complete dataset.

The first point we would like to illustrate is the rate of success. Under the same initialization conditions, we compared the rate of success of five methods (CIG-DS, CIG-OB, CIG-AS, EMFP, and Euclidean SG). Here, 'success' means that the log-likelihood increases with each update and finally converges.

In [57], expectation-maximisation using the fixed-point algorithm in [59] (EMFP) is proved to be the most robust method for mixtures of MGGD. Therefore, this method is included in our comparison. Also in [57], an online classic (that is, Euclidean) SG method was introduced, which we also included for comparison. All gradient methods (CIG and SG) used the same mini-batch size $b = 1000$.

In the following Table 7.1, the rates of success are presented. We can see that the rates of success of CIG methods are significantly better. This shows that the intrinsic geometric structures on the parameter space provide a more stable estimation process. Second, we confirm the convergence rates of the CIG-DS and CIG-AS methods, stated in

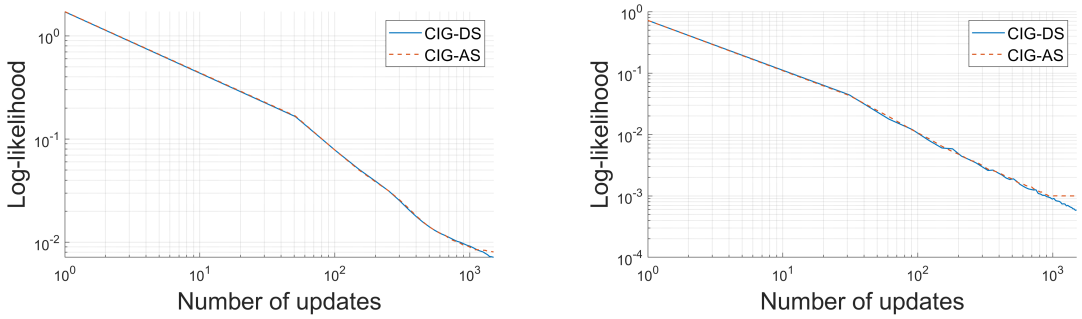
Table 7.1: MECD: percentage of successful runs

method	success rate
EM-FP	83%
SG	77%
CIG-DS	90%
CIG-OB	89%
CIG-AS	90%

Propositions 11 and 12. In Figure 7.7, two log-log plots are presented. In these plots, the x-axis denotes the number of updates and the y-axis denotes the empirically estimated Kullback-Leibler divergence (4.6). In both these plots, the slope of the curve approaches -1, which means the empirical divergence $\hat{D}(\boldsymbol{\theta})$ is $\mathcal{O}(n^{-1})$, as in (6.1).

As mentioned in Proposition 11, the coefficient a, used in computing step-sizes ($\eta^{(n)} = a/n$), needs to verify the condition $a > \frac{1}{2\lambda}$. Here, since it is quite difficult to compute λ, the coefficient a was selected manually, through trial and error. This procedure lead to $a = 1000$ for mixtures of MGGD, and $a = 10$ for mixtures of Student T.

Then, we compared the performance of the five methods (CIG-DS, CIG-OB, CIG-AS, EMFP, and Euclidean SG). In Figure 7.8, the x-axis represents the number of epochs (one epoch means all samples in the dataset have been traversed exactly once), and the y-axis represents the log-likelihood.

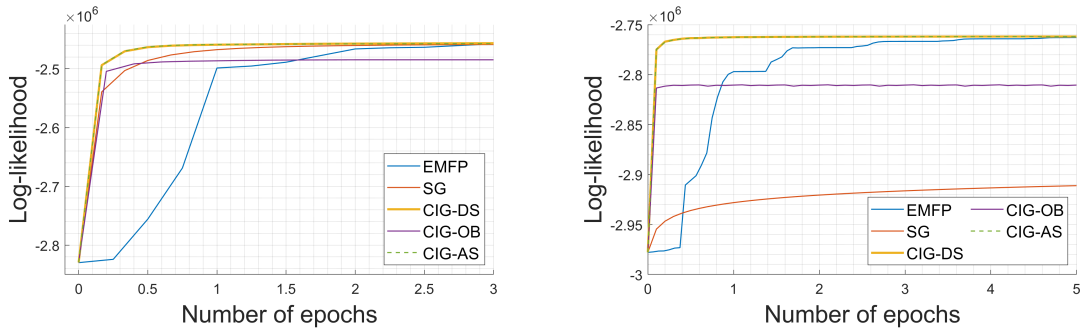


(a) mixture of MGGD (b) mixture of Student T

Figure 7.7: MECD: The convergence rate of CIG-DS and CIG-AS

For mixtures of MGGD, EMFP converged in 3 epochs. The classic SG method is faster, but it still takes almost 2 epochs. The CIG-DS and CIG-AS methods can achieve the same efficiency as EMFP in only 1 epoch. The CIG-OB is less efficient than the others, because the mini-batch size is constant $b = 1000$, leading to an insufficient number of good line-search step-sizes.

For mixtures of Student T, EMFP took nearly 5 epochs to obtain convergence. The classic SG converges slowly and is less efficient than the other methods. CIG-DS and CIG-AS have the best performance, since they achieve the same efficiency as EMFP in 1 epoch. Thanks to the line-search step-size, CIG-OB converges quickly. However, due to the fixed mini-batch size, its efficiency is not good enough.



(a) mixture of MGGD (b) mixture of Student T

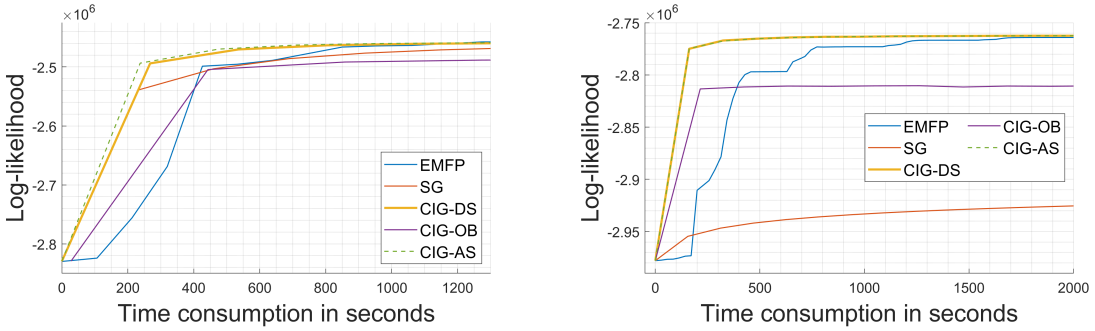
Figure 7.8: MECD: Number of epochs versus log-likelihood[1]

Finally, we observed the variation of the log-likelihood versus time consumption. In Figure 7.9, the x-axis represents the time consumption (in seconds), and the y-axis represents the variation of log-likelihood.

For mixtures of MGGD, EMFP converged in 1000 seconds. The classic (Euclidean) SG is similar to EMFP for time consumption. Then, CIG-DS and CIG-AS are the fastest methods, as they converged in 600 seconds, but CIG-OB did not show any advantages in terms of time consumption. We think that the reason is the sensitivity for β in the MGGD case, which requires more time for the best step-size to be selected.

[1]The number of epochs means the number of times that the complete dataset is traversed. For the EMFP method, which is a batch (offline) method, the number of epochs is equal to the number of complete parameter updates.

For mixtures of Student T, EMFP took 1750 seconds to converge to the maximum likelihood value, while the SG did not converge in this time interval. The CIG-DS and CIG-AS methods showed the fastest convergence, as they converged in 750 seconds. CIG-OB converged in less than 250 seconds, but it is still not rather inefficient. Based on the



(a) mixture of MGGD

(b) mixture of Student T

Figure 7.9: MECD: Time-consumption versus log-likelihood

above comparisons, we believe that CIG-DS and CIG-AS are the most effective methods for online estimation of MECD.

7.3 Applications to real data

7.3.1 Colour transformation with MGGD

Here, we consider a first application of the CIG method for ECD estimation, to colour transformation for image editing [43]. Precisely, this application will focus on the estimation of MGGD. Its goal is to replace the colour distribution of an input image by the colour distribution of a given target image. The main idea is to fit the input and the target distributions, with two different MGGD models. Then, the transformation between these two MGGD is implemented by a linear Monge-Kantorovich transformation for Σ, and a stochastic transformation for β [43, 63]. In each image, the pixels correspond to 3-dimensional vectors, in the case of RGB images, and to 5-dimensional vectors, when spatial gradient-field information is added to colour information.

Starting with the 3D RGB case, Figure 7.10 presents the transformed images and some of their details. The detail (a1) clearly shows that the cloud 'drawn' by MM appears too green. Similarly, FP also presents a green appearance, in detail (a2). On the contrary, the CIG offline and CIG online methods show pure white cloud colour in (a3) and (a4). Note also the difference in the amount of blue in the shadows on the grass. Too much blue is mixed with the shadows, in MM's output detail (b1). In details (b2),(b3),(b4), the results of FP and CIG methods lead to a more natural appearance.

From our point of view, the most interesting aspect of this application is in terms of computation time. The CIG online method takes about 10 seconds for two images (input and output). In contrast, FP and CIG offline methods each require more than two hours. In other words, CIG online has a decisive advantage, in terms of time consumption.

Then, gradient-field information was included, so the transformation came to involve 5-dimensional vectors, which consist in three colour components (of CIELAB) and two components of the image spatial gradient field ($\mathrm{d}x$ and $\mathrm{d}y$). For this application, the shape parameter of the MGGD model was supposed to fixed. Figure 7.11 presents the four different outputs. It can be observed that the output of the FP and CIG methods is significantly better than that of MM. In the transformed result of MM, the hue is darker and greener. FP and CIG results are better, since the frost on the grass is whiter and appears more natural, and the forest on the mountain in the image also appears darker. The two images in Figure 7.11 have more than 1.2×10^6 pixels (i.e. 1.2×10^6 samples). The FP and CIG offline methods need more than 4 hours to run, on the these two images. The CIG online method needs only 21 seconds.

We also considered an application to full HD images. In this case, as demonstrated in Figure 7.12, the advantages of the CIG online algorithm were significant. The result of MM failed to achieve the colour of the autumn leads in the target image, showing light green instead of yellow. Since the input image and the target image have more than 4×10^6 pixels (that is 4×10^6 samples), it was not feasible to run FP and CIG offline, with the entire dataset. Rather, the estimation was done on subsets of the complete dataset. These two subsets have 4×10^5 samples, that are randomly taken from the original images. In the autumn leafs obtained using FP and CIG offline, the yellow colour has obviously been smeared. CIG online is more natural, and the yellow colour is more uniform, and it is closer to the style of the target image.

Figure 7.10: 3D colour transformation

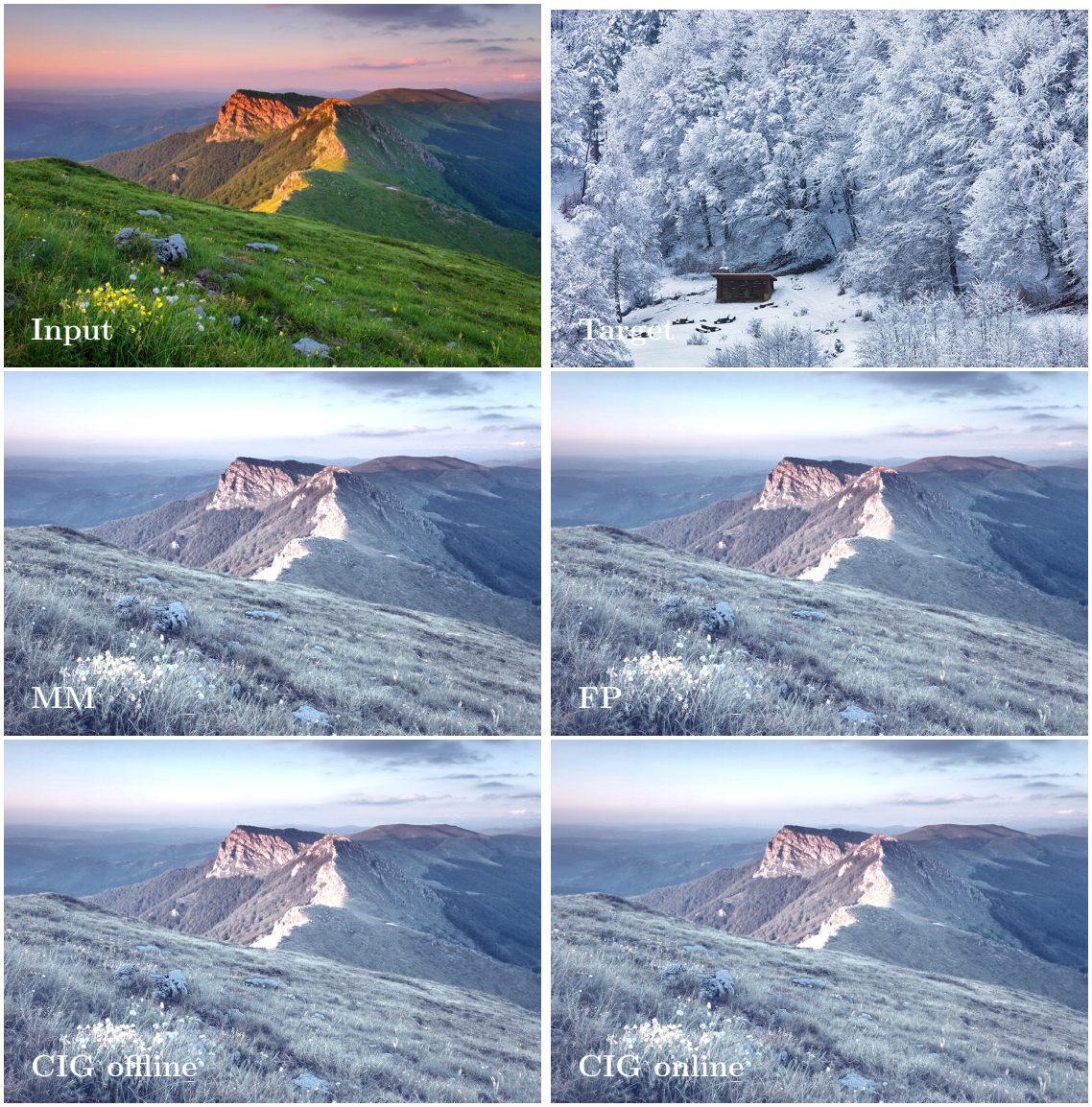


Figure 7.11: 5D colour transformation

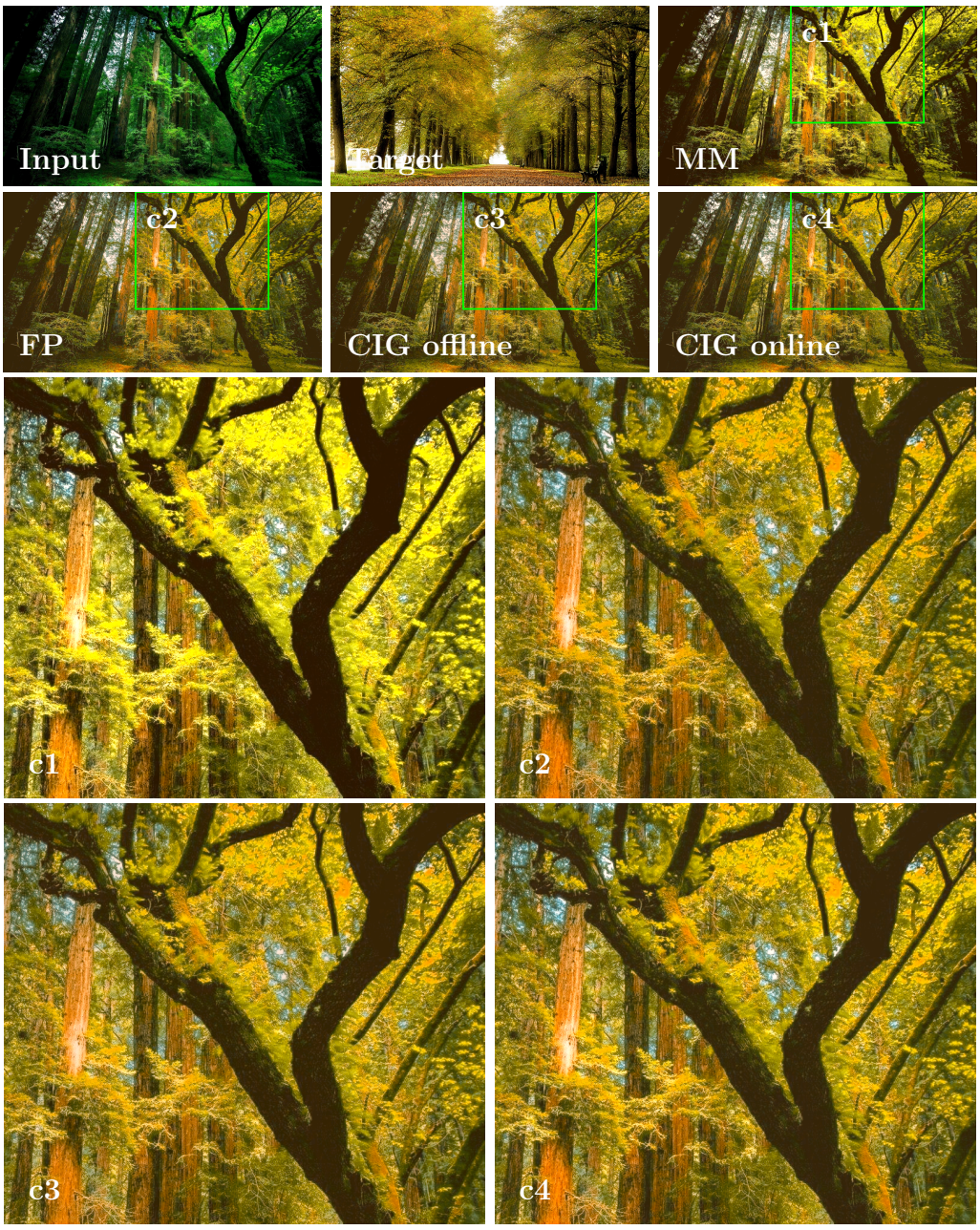


Figure 7.12: 5D colour transformation for full HD image

7.3.2 Texture segmentation with Mixture of MGGD

This section applies the CIG online method for MECD estimation, to the problem of texture segmentation. Specifically, we consider mixtures of MGGD, as they have been successfully used to model wavelet statistics of texture images [82, 49].

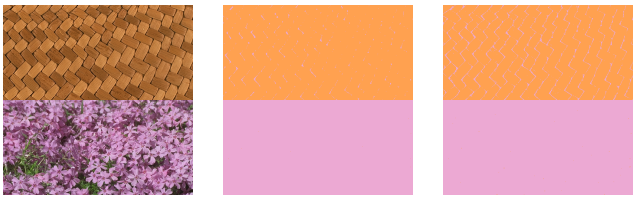
For this application, we randomly selected five pictures from the VisTex database [55], fabric, white flowers, pink flowers, leafs and water. Each texture image was considered as an RGB 3-dimensional image, and modelled by a mixture of MGGD, whose parameters $\boldsymbol{\theta} = (w, (\mu_k)_k, (\Sigma_k)_k, (\beta_k)_k)_k$ were estimated using five different methods, EMFP, SG, CIG-DS, CIG-OB and CIG-AS, already discussed in Subsection 7.2.2.

In order to evaluate the true performance of these estimation methods, we did not use special filters or feature extraction techniques. Rather, each given picture element was directly classified by its log-likelihood value. The sub-distribution (mixture component) with the largest log-likelihood value is the class to which the picture element is affected.

Each of the five methods was run for the same number of epochs (the number of epochs means the number of times that the complete dataset is traversed), and all were initialized using a k-means method.

For the first application, we considered a scenario with $K = 2$ textures. Figure 7.13 shows the visual segmentation results, and table 7.3.2 gives the accuracy of segmentation. In terms of accuracy, CIG-DS, CIG-AS and EMFP have the best performance (greater than 99%), followed by SG, while CIG-OB is the worst.

In terms of visual segmentation results, CIG-DS and AS are similar to the EMFP-method. The two parts (fabric and flower) are completely and clearly segmented. However, in Figure 7.13(c), the results of SG show traces in the fabric were misidentified as flowers, and in Figure 7.13(e), the green leafs in the flowers were misclassified as fabric. For this scenario, it is clear that CIG-DS and CIG-AS are the most effective methods.



(a) Texture 1 (b) EM-FP (c) SG

(d) CIG-DS (e) CIG-OB (f) CIG-AS

Figure 7.13: MECD: Segmentation with $K = 2$ textures

For the second application, the number of components is $K = 5$. The textures of leafs and water were added. The visual results are presented in Figure 7.14(b). For this more complicated cas, the five segmentations are evaluated by the confusion matrices, and the confusion matrices are summarized in Table 7.3.2 with accuracy and F1 score (weighted average).

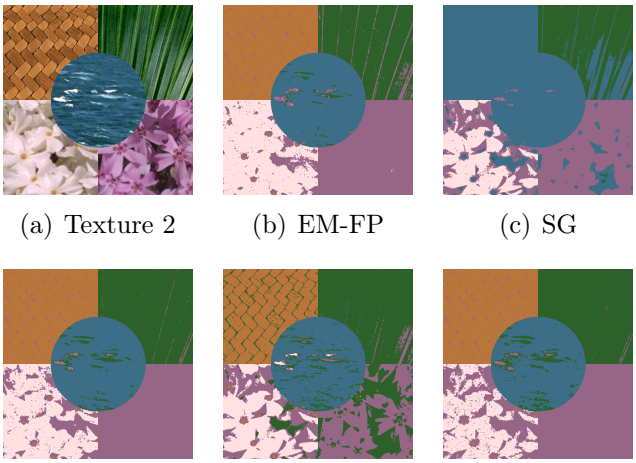
According to Table 7.3.2 and Figure 7.14(b), EMFP is always rather satisfactory. The fabric and the pink flowers are totally identified, while some traces in leafs, water and

method	correct estimates
EM-FP	99%
SG	97%
CIG-DS	99%
CIG-OB	73%
CIG-AS	99%

Table 7.2: MECD: Accuracy of segmentation (Texture 1)

white flowers are slightly misclassified. For SG method, in Figure 7.14(c), the fabric was completely misjudged as water, consequently, the accuracy was as low as 63.328% (see Table 7.3.2). For CIG methods, DS and AS also share the good performance of EMFP. In figures 7.14(d) and 7.14(f), fabric, leafs and pink flowers were clearly segmented. There are only some minor errors in water and white flowers. What is of note is that, with regard to the leafs texture, the performance of CIG-DS and CIG-AS is even better than EMFP.

Their accuracy is also quite good, with CIG-AS producing a similar accuracy to EMFP, i.e. 92%, and CIG-DS achieving an even better accuracy 96%. CIG-DS has almost achieved the same F1 score as EMFP with less computation time.



(a) Texture 2 (b) EM-FP (c) SG

(d) CIG-DS (e) CIG-OB (f) CIG-AS

Figure 7.14: MECD: Segmentation with $K = 5$ textures

Method	Accuracy	F1 score (weighted average)
EMFP	93%	0.93
SG	63%	0.58
CIG-DS	96%	0.92
CIG-OB	85%	0.78
CIG-AS	92%	0.88

Table 7.3: MECD: Accuracy of segmentation (Texture 2)

7.4 Conclusion

This final chapter compared the CIG methods, introduced in Chapters 5 and 6, to other, state-of-the-art methods, for ECD and MECD estimation, such as method of moments, fixed-point, expectation-maximization, and classical stochastic gradient [82, 59, 50, 57].

For ECD estimation, the CIG offline and CIG online methods were compared to the method of moments (MM) and to the fixed-point method (FP), as given in [82] and [59]. Figures 7.4 and 7.6 present the comparison, in terms of efficiency (roughly speaking, final estimation accuracy) and time consumption, respectively.

Figure 7.4 shows that the CIG offline method provides the best efficiency, being slightly better than the FP method. However, from Figure 7.6, it appears the CIG offline method is less attractive than the FP method, in terms of time consumption.

On the other hand, the same two figures (7.4 and 7.6) show the CIG online method is still comparable to the FP method, in terms of efficiency, but is highly attractive, in terms of time consumption, since it on par with the method of moments, and converges “instantly", even as the size of the dataset increases.

Overall, the CIG online method appears as the best choice for ECD estimation, among all methods under consideration.

For MECD estimation, CIG online methods were compared to two already existing methods, EMFP (Expectation-Maximisation with Fixed-Point [59, 37]) and SG (Euclidean Stochastic Gradient [57]).

First of all, Table 7.1 showed CIG online methods (CIG-DS and CIG-AS) had the highest rate of success among all methods considered. Under the same initialisation conditions, CIG online methods had the highest chance of converging to a local maximum of the log-likelihood function.

Figures 7.8 and 7.9 compared CIG online methods to EMFP and SG, in terms of their performance and time consumption. These figures showed the CIG-DS and CIG-AS methods were able to achieve the same performance as EMFP and SG, while being two to three times faster.

Overall, the CIG online methods also seemed the most attractive for MECD estimation, as they achieved comparable, or even better performance, to other methods, within a notably shorter time.

The applications to real data, discussed in Section 7.3, showed that CIG methods are able to tackle very large datasets (in the form of HD images, for example), and produce highly satisfactory results.

In conclusion, the numerical experiments and real-data applications in this chapter validate the methodology of CIG online methods, which may be regarded as very interesting new methods, to be considered by any users wishing to work with ECD or MECD.

Chapter 8

Conclusion and perspectives

In this thesis, an original geometric method was proposed, for online estimation of ECD and MECD models. This new method was named the CIG method (Component-wise Information Gradient method).

To begin, since the parameter spaces of ECD and MECD are Cartesian product spaces, a novel Riemannian metric was introduced on these spaces, and was named the CIM (Component-wise Information Metric). At each point of the parameter space, the CIM is a direct product of the Fisher Information Metrics, with respect to individual parameters (for example, location, scatter, and shape parameters).

This leads to the component-wise information gradient, which combines the information gradients with respect to these parameters. Based on the component-wise information gradient, the CIG method implements an alternating update scheme for the parameters, where each parameter is updated in its own turn.

The CIM is an easy-to-compute substitute, for the FIM (Fisher Information Metric). Similarly, the CIG method is a computationally-advantageous substitute for the Riemannian information gradient method, which would be very hard to implement, for complete ECD or MECD models.

For some particular cases of ECD models, the CIM and FIM coincide, so the CIG method shares many of the excellent properties of the Riemannian information gradient method. For instance, it achieves a mean-square convergence rate of order $\mathcal{O}(n^{-1})$, and it is asymptotically efficient, with an automatic choice of step-size.

For complete ECD or MECD models, the FIM is not tractable, and it is anyway different from the CIM. The CIG method shares some, but not all, of the properties of the Riemannian information gradient method. For example, it can still achieve a mean-square convergence rate of order $\mathcal{O}(n^{-1})$, but this requires a manual selection of step-size.

At the cost of some additional computations, this situation can be improved dramatically. The adaptive step-size version of the CIG method (proposed in Proposition 12), implements adaptive selection of step-size which provides a much faster convergence rate $\mathcal{O}(c^{-n})$ (where $c > 1$).

Numerical simulations and real applications have proved that the CIG method has significant advantages over existing methods, especially in terms of time consumption. In future work, the problem of order selection will be included into our investigation of MECD estimation, using information criteria (AIC or BIC), or a non-parametric Bayesian approach. In addition, new classes of algorithms (such as stochastic EM and its many variants) will be studied and adapted to the estimation of ECD and MECD. Finally, the idea of CIG may be generalized to online estimation of any distributions of position-scale type, such as Riemannien Gaussian distribution and von Mise-Fisher distributions.

Appendix A

Proofs of chapter 4

A.1 Proofs of the main results

A.1.1 Proof of Proposition 1

The proof is a generalisation of the original proof in [11], itself modeled on the proof for the Euclidean case in [68]. Throughout the following, let $\mathcal{X}_n$ be the σ-field generated by $x_1, \cdots, x_n$ [72]. Recall that $(x_n; n = 1, 2, \cdots)$ are i.i.d. with distribution P_{θ^*}. Therefore, by (4.1a), $\theta^{(n)}$ is $\mathcal{X}_n$-measurable and x_{n+1} is independent from $\mathcal{X}_n$. Thus, using elementary properties of conditional expectation [72],

$$\mathbb{E}\left[U(\theta^{(n)}, x_{n+1})\middle|\mathcal{X}_n\right] = -D(\theta^{(n)}) \tag{A.1a}$$

$$\mathbb{E}\left[\|U(\theta^{(n)}, x_{n+1})\|^2\middle|\mathcal{X}_n\right] = V(\theta^{(n)}) \tag{A.1b}$$

where (A.1a) follows from (4.1c), and (A.1b) from (u1). Let L be a Lipschitz constant for $\nabla D(\theta)$, and C be an upper bound on $V(\theta)$, for $\theta \in \Theta^*$. The following inequality is now proved, for any positive integer n,

$$\mathbb{E}\left[D(\theta^{(n+1)}) - D(\theta^{(n)})\middle|\mathcal{X}_n\right] \leqslant [\eta^{(n+1)}]^2\, LC - \eta^{(n+1)}\|\nabla D(\theta^{(n)})\|^2 \tag{A.2}$$

once this is done, Proposition 1 is obtained by applying the Robbins-Siegmund theorem [25].

Proof of (A.2) : let $c(t)$ be the geodesic connecting $\theta^{(n)}$ to $\theta^{(n+1)}$ with equation

$$c(t) = \mathrm{Exp}_{\theta^{(n)}}\left(t\eta^{(n+1)}U(\theta^{(n)}, x_{n+1})\right) \tag{A.3}$$

From the fundamental theorem of calculus,

$$\begin{aligned} D(\theta^{(n+1)}) - D(\theta^{(n)}) \;=& \eta^{(n+1)}\,\langle U(\theta^{(n)}, x_{n+1}), \nabla D(\theta^{(n)})\rangle \\ &+\; \eta^{(n+1)} \int_0^1 \left[\langle \dot{c}, \nabla D\rangle_{c(t)} - \langle \dot{c}, \nabla D\rangle_{c(0)}\right] \mathrm{d}t \end{aligned} \tag{A.4}$$

Since the online estimates $\theta^{(n)}$ are stable, $\theta^{(n)}$ and $\theta^{(n+1)}$ both lie in Θ^*. Since θ^* is convex, the whole geodesic $c(t)$ lies in Θ^*. Then, since $\nabla D(\theta)$ is Lipschitz on Θ^*, it follows from (A.4),

$$\begin{aligned} D(\theta^{(n+1)}) - D(\theta^{(n)}) \;\leqslant& \eta^{(n+1)}\,\langle U(\theta^{(n)}, x_{n+1}), \nabla D(\theta^{(n)})\rangle \\ &+\; [\eta^{(n+1)}]^2\, L\|U(\theta^{(n)}, x_{n+1})\|^2 \end{aligned} \tag{A.5}$$

Taking conditional expectations in this inequality, and using (A.1a) and (A.1b),

$$\mathbb{E}\left[D(\theta^{(n+1)}) - D(\theta^{(n)})\big|\mathcal{X}_n\right] \leqslant -\eta^{(n+1)}\|\nabla D(\theta^{(n)})\|^2 + [\eta^{(n+1)}]^2\, L\, V(\theta^{(n)}) \tag{A.6}$$

so (A.2) follows since (u1) guarantees $V(\theta^{(n)}) \leqslant C$. □

Conclusion: by the Robbins-Siegmund theorem, inequality (A.2) implies that, almost surely,

$$\lim D(\theta^{(n)}) = D_\infty < \infty \;\; \textit{and} \;\; \sum_{n=1}^{\infty} \eta^{(n+1)}\, \|\nabla D(\theta^{(n)})\|^2 < \infty \tag{A.7}$$

In particular, from the first condition in (4.1b), convergence of the sum in (A.7) implies

$$\lim \|\nabla D(\theta^{(n)})\| = 0 \qquad \text{almost surely} \tag{A.8}$$

Now, since the sequence of online estimates $\theta^{(n)}$ lies in the compact set Θ^*, it has at least one point of accumulation in this set, say θ_*. If $\theta^{(n_k)}$ is a subsequence of $\theta^{(n)}$, converging to θ_*,

$$\|\nabla D(\theta_*)\| = \lim \|\nabla D(\theta^{(n_k)})\| = \lim \|\nabla D(\theta^{(n)})\| = 0 \qquad \text{almost surely}$$

where the third equality follows from (A.8). This means that θ_* is a stationary point of $D(\theta)$ in Θ^*. Thus, (d1) implies $\theta_* = \theta^*$ is the unique point of accumulation of $\theta^{(n)}$. In other words, $\lim \theta^{(n)} = \theta^*$ almost surely.

A.1.2 Proof of Proposition 2

The proof is modeled on the proofs for the Euclidean case, given in [7, 58]. It relies on the following geometric Lemmas 1 and 2. Lemma 1 will be proved in Appendix A.2. On the other hand, Lemma 2 is the same as the trigonometric distance bound of [86]. For Lemma 1, recall that $\lambda > 0$ denotes the smallest eigenvalue of the matrix H defined in (4.8).

Lemma 1 *for any $\mu < \lambda$, there exists a neighborhood $\bar\Theta^*$ of θ^*, contained in Θ^*, with*

$$\langle \mathrm{Exp}_\theta^{-1}(\theta^*), \nabla D(\theta)\rangle \leqslant -\mu\, d^2(\theta,\theta^*) \qquad \textit{for } \theta \in \bar\Theta^* \tag{A.9a}$$

Lemma 2 *let $-\kappa^2$ be a lower bound on the sectional curvature of Θ in Θ^*, and $C_\kappa = R\kappa\coth(R\kappa)$ where R is the diameter of Θ^*. For $\tau, \theta \in \Theta^*$, where $\tau = \mathrm{Exp}_\theta(u)$,*

$$d^2(\tau,\theta^*) \leqslant d^2(\theta,\theta^*) - 2\,\langle \mathrm{Exp}_\theta^{-1}(\theta^*), U\rangle + C_\kappa \|U\|^2 \tag{A.9b}$$

Proof of equation (4.9): let $\eta^{(n)} = \frac{a}{n}$ with $2\lambda a > 2\mu a > 1$ for some $\mu < \lambda$, and let $\bar\Theta^*$ be the neighborhood corresponding to μ in Lemma 1. By Proposition 1, the $\theta^{(n)}$ converge to θ^* almost surely. Without loss of generality, it can be assumed that all the $\theta^{(n)}$ lie in $\bar\Theta^*$, almost surely. Then, (4.1a) and Lemma 2 imply, for any positive integer n,

$$\begin{aligned} d^2(\theta^{(n+1)},\theta^*) \leqslant\, & d^2(\theta^{(n)},\theta^*) - 2\eta^{(n+1)}\,\langle \mathrm{Exp}_{\theta^{(n)}}^{-1}(\theta^*), U(\theta^{(n)}, x_{n+1})\rangle \\ & + [\eta^{(n+1)}]^2\, C_\kappa \|U(\theta^{(n)}, x_{n+1})\|^2 \end{aligned} \tag{A.10a}$$

Indeed, this follows by replacing $\tau = \theta^{(n+1)}$ and $\theta = \theta^{(n)}$ in (A.9b). Taking conditional expectations in (A.10a), and using (A.1a) and (A.1b),

$$\mathbb{E}\left[d^2(\theta^{(n+1)},\theta^*)\big|\mathcal{X}_n\right] \leqslant d^2(\theta^{(n)},\theta^*) + 2\eta^{(n+1)}\,\langle \mathrm{Exp}_{\theta^{(n)}}^{-1}(\theta^*), \nabla D(\theta^{(n)})\rangle + [\eta^{(n+1)}]^2\, C_\kappa V(\theta^{(n)})$$

Then, by (u1) and (A.9a) of Lemma 1,

$$\mathbb{E}\left[d^2(\theta^{(n+1)},\theta^*)\middle|\mathcal{X}_n\right] \leqslant d^2(\theta^{(n)},\theta^*)(1-2\eta^{(n+1)}\mu) + [\eta^{(n+1)}]^2\, C_\kappa C \tag{A.10b}$$

where C is an upper bound on $V(\theta)$, for $\theta \in \Theta^*$. By further taking expectations

$$\mathbb{E}\, d^2(\theta^{(n+1)},\theta^*) \leqslant \mathbb{E}\, d^2(\theta^{(n)},\theta^*)(1-2\eta^{(n+1)}\mu) + [\eta^{(n+1)}]^2\, C_\kappa C \tag{A.10c}$$

Using (A.10c), the proof reduces to an elementary reasoning by recurrence. Indeed, replacing $\eta^{(n)} = \frac{a}{n}$ into (A.10c), it follows that

$$\mathbb{E}\, d^2(\theta^{(n+1)},\theta^*) \leqslant \mathbb{E}\, d^2(\theta^{(n)},\theta^*)\left(1-\frac{2\mu a}{n+1}\right) + \frac{a^2 C_\kappa C}{(n+1)^2} \tag{A.11a}$$

On the other hand, if $b(n) = \frac{b}{n}$ where $b > a^2 C_\kappa C\,(2\mu a - 1)^{-1}$, then

$$b(n+1) \geqslant b(n)\left(1-\frac{2\mu a}{n+1}\right) + \frac{a^2 C_\kappa C}{(n+1)^2} \tag{A.11b}$$

Let b be sufficiently large, so (A.11b) is verified and $\mathbb{E}\, d^2(\theta^{(n_o)},\theta^*) \leqslant b(n_o)$ for some n_o. Then, by recurrence, using (A.11a) and (A.11b), one also has that $\mathbb{E}\, d^2(\theta^{(n)},\theta^*) \leqslant b(n)$ for all $n \geqslant n_o$. In other words, (4.9) holds true.

A.1.3 Proof of Proposition 3

The proof is modeled on the proof for the Euclidean case in [58]. To begin, let W_n be the stochastic process given by

$$W_n \,=\, n^p\, d^2(\theta^{(n)},\theta^*) + n^{-q} \qquad \text{where } q \in (0, 1-p) \tag{A.12a}$$

The idea is to show that this process is a positive supermartingale, for sufficiently large n. By the supermartingale convergence theorem [72], it then follows that W_n converges to a finite limit, almost surely. In particular, this implies

$$\lim n^p\, d^2(\theta^{(n)},\theta^*) \,=\, \mathscr{L}_p < \infty \qquad \text{almost surely} \tag{A.12b}$$

Then, $\mathscr{L}_p$ must be equal to zero, since p is arbitrary in the interval $(0,1)$. Precisely, for any $\varepsilon \in (0, 1-p)$,

$$\mathscr{L}_p \,=\, \lim n^p\, d^2(\theta^{(n)},\theta^*) \,=\, \lim n^{-\varepsilon}\, n^{p+\varepsilon}\, d^2(\theta^{(n)},\theta^*) \,=\, \left(\lim n^{-\varepsilon}\right) \mathscr{L}_{p+\varepsilon} \,=\, 0$$

It remains to show that W_n is a supermartingale, for sufficiently large n. To do so, note that by (A.10b) from the proof of Proposition 2,

$$\mathbb{E}\left[W_{n+1} - W_n | \mathcal{X}_n\right] \leqslant d^2(\theta^{(n)},\theta^*)\,\frac{p-2\mu a}{(n+1)^{1-p}} \,+\, \frac{a^2 C_\kappa C}{(n+1)^{2-p}} \,-\, \frac{q}{(n+1)^{q+1}}$$

Here, the first term on the right-hand side is negative, since $2\mu a > 1 > p$. Moreover, the third term dominates the second one for sufficiently large n, since $q < 1-p$. Thus, for sufficiently large n, the right-hand side is negative, and W_n is a supermartingale.

A.1.4 Proof of Proposition 4

the proof relies on the following geometric Lemmas 3 and 4, which are used to linearise Algorithm (4.1a), in terms of the normal coordinates $(\theta^{(n)})^\alpha$. This idea of linearisation in terms of local coordinates also plays a central role in [77].

Lemma 3 *let $\theta^{(n)}, \theta^{(n+1)}$ be given by (4.1a) with $\eta^{(n)} = \frac{a}{n}$. Then, in a system of normal coordinates with origin at θ^*,*

$$\begin{aligned}(\theta^{(n+1)})^\alpha \;=\; & (\theta^{(n)})^\alpha + \eta^{(n+1)}\, U^\alpha_{n+1} + [\eta^{(n+1)}]^2\, \pi^\alpha_{n+1} \\ & \textit{with } \mathbb{E}\left|\pi^\alpha_{n+1}\right| = \mathcal{O}(n^{-1/2})\end{aligned} \tag{A.13a}$$

where U^α_{n+1} are the components of $U(\theta^{(n)}, x_{n+1})$.

Lemma 4 *let $v_n = \nabla D(\theta^{(n)})$. Then, in a system of normal coordinates with origin at θ^*,*

$$\begin{aligned} v^\alpha_n \;=& H_{\alpha,\beta}\,(\theta^{(n)})^\beta \;+\; \rho^\alpha_n \\ \rho^\alpha_n \;=& o\left(d(\theta^{(n)}, \theta^*)\right)\end{aligned} \tag{A.13b}$$

where v^α_n are the components of v_n and the $H_{\alpha,\beta}$ were defined in (4.8).

Linearisation of (4.1a): let $U(\theta^{(n)}, x_{n+1}) = -v_n + w_{n+1}$. Then, it follows from (A.13a) and (A.13b),

$$\begin{aligned}(\theta^{(n+1)})^\alpha \;=& (\theta^{(n)})^\alpha \;-\; \eta^{(n+1)}\, H_{\alpha,\beta}\,(\theta^{(n)})^\beta \;-\; \eta^{(n+1)}\,\rho^\alpha_n \;+\; \eta^{(n+1)}\, w^\alpha_{n+1} \\ & +\; [\eta^{(n+1)}]^2\, \pi^\alpha_{n+1}\end{aligned} \tag{A.14a}$$

Denote the re-scaled coordinates $n^{1/2}(\theta^{(n)})^\alpha$ by g^α_n, and recall $\eta^{(n)} = \frac{a}{n}$. Then, using the estimate $(n+1)^{1/2} = n^{1/2}(1 + (2n)^{-1} + \mathcal{O}(n^{-2}))$, it follows from (A.14a) that

$$g^\alpha_{n+1} \;=\; g^\alpha_n + \frac{A_{\alpha,\beta}}{n+1}\, g^\beta_n \;+\; \frac{a}{(n+1)^{1/2}}\left[B_{\alpha,\beta}\,(\theta^{(n)})^\beta \;-\; \rho^\alpha_n \;+\; w^\alpha_{n+1} \;+\; \frac{a\pi^\alpha_{n+1}}{n+1}\right] \tag{A.14b}$$

where $A_{\alpha,\beta} = \frac{1}{2}\delta_{\alpha,\beta} \;-\; a\, H_{\alpha,\beta}$ and $B_{\alpha,\beta} = \mathcal{O}(n^{-1})$. Equation (A.14b) is a first-order, inhomogeneous, linear difference equation, for the "vector" g_n of components g^α_n. □

Study of equation (A.14b): switching to vector-matrix notation, equation (A.14b) is of the general form

$$g_{n+1} \;=\; \left(I \;+\; \frac{A}{n+1}\right) g_n \;+\; \frac{a\,\xi_{n+1}}{(n+1)^{1/2}} \tag{A.15a}$$

where I denotes the identity matrix, A has matrix elements $A_{\alpha,\beta}$, and (ξ_n) is a sequence of inputs. The general solution of this equation is [45, 58]

$$g_n \;=\; A_{n,m}\, g_m \;+\; \sum_{k=m+1}^{n} A_{n,k}\, \frac{a\,\xi_k}{k^{1/2}} \qquad \text{for } n \geqslant m \tag{A.15b}$$

where the transition matrix $A_{n,k}$ is given by

$$A_{n,k} \;=\; \prod_{j=k+1}^{n}\left(I + \frac{A}{j}\right) \qquad A_{n,n} \;=\; I \tag{A.15c}$$

Since $2\lambda a > 1$, the matrix A is stable. This can be used to show that [45, 58]

$$q > \frac{1}{2} \text{ and } \mathbb{E}\left|\xi_n\right| = \mathcal{O}(n^{-q}) \implies \lim g_n = 0 \text{ in probability} \tag{A.15d}$$

where $|\xi_n|$ denotes the Euclidean vector norm. Then, it follows from (A.15d) that g_n converges to zero in probability, in each one of the three cases

$$\begin{aligned}
\xi^\alpha_{n+1} &= B_{\alpha,\beta}\,(\theta^{(n)})^\beta \\
\xi^\alpha_{n+1} &= \rho^\alpha_n \\
\xi^\alpha_{n+1} &= \frac{\pi^\alpha_{n+1}}{n+1}
\end{aligned}$$

Indeed, in the first two cases, the condition required in (A.15d) can be verified using (4.9), whereas in the third case, it follows immediately from the estimate of $\mathbb{E}|\pi^\alpha_{n+1}|$ in (A.13a). □

Conclusion: by linearity of (A.14b), it is enough to consider the case $\xi^\alpha_{n+1} = w^\alpha_{n+1}$ in (A.15a). Then, according to (A.15b), g_n has the same limit distribution as the sums

$$\tilde{g}_n = \sum_{k=1}^{n} A_{n,k}\,\frac{a w_k}{k^{1/2}} \tag{A.16}$$

By (A.1), (w_k) is a sequence of square-integrable martingale differences. Therefore, to conclude that the limit distribution of $\tilde{g}_n$ is a centred d-variate normal distribution, with covariance matrix Σ given by (4.11), it is enough to verify the conditions of the martingale central limit theorem [36],

$$\lim \max_{k \leqslant n} \left| A_{n,k}\,\frac{a w_k}{k^{1/2}} \right| = 0 \text{ in probability} \tag{A.17a}$$

$$\sup \mathbb{E}\left|\tilde{g}_n\right|^2 < \infty \tag{A.17b}$$

$$\lim \sum_{k=1}^{n} \frac{a^2}{k}\, A_{n,k} \Sigma_k A_{n,k} = \Sigma \text{ in probability} \tag{A.17c}$$

where Σ_k is the conditional covariance matrix

$$\Sigma_k = \mathbb{E}\left[w_k w_k^\dagger \middle| \mathcal{X}_{k-1} \right] \tag{A.18}$$

Conditions (A.17) are verified in Appendix A.3, which completes the proof.

A.1.5 Proof of Proposition 5

Denote $\partial_\alpha = \frac{\partial}{\partial \theta^\alpha}$ the coordinate vector fields of the normal coordinates $(\theta)^\alpha$. Since $\langle \cdot, \cdot \rangle$ coincides with the Fisher information metric of the model P, it follows from (4.8) and (A.32),

$$H_{\alpha,\beta} = \langle \partial_\alpha, \partial_\beta \rangle_{\theta^*} \tag{A.19a}$$

However, by the definition of normal coordinates [62], the ∂_α are orthonormal at θ^*. Therefore,

$$H_{\alpha,\beta} = \delta_{\alpha,\beta} \tag{A.19b}$$

Thus, the matrix H is equal to the identity matrix, and its smallest eigenvalue is $\lambda = 1$.

Proof of (i): this follows directly from Propositions 2 and 3. Indeed, since $\lambda = 1$, the conditions of these propositions are verified, as soon as $2a > 1$. Therefore, (4.9) and (4.10) hold true. □

Proof of (ii): this follows from Proposition 4. The conditions of this proposition are verified, as soon as $2a > 1$. Therefore, the distribution of the re-scaled coordinates $\big(n^{1/2}(\theta^{(n)})^\alpha\big)$ converges to a centred d-variate normal distribution, with covariance matrix Σ given by Lyapunov's equation (4.11). If $a = 1$, then (A.19b) implies $A_{\alpha,\beta} = -\frac{1}{2}\delta_{\alpha,\beta}$, so that Lyapunov's equation (4.11) reads $\Sigma = \Sigma^*$, as required. □

For the following proof of (iii), the reader may wish to recall that summation convention is used throughout the present work. That is [62], summation is implicitly understood over any repeated subscript or superscript from the Greek alphabet, taking the values $1\,, \ldots,\, d\,$.

Proof of (iii): let $\ell(\theta) = \log L(\theta)$ and assume $U(\theta, x)$ is given by (4.3). Then, by the definition of normal coordinates [62], the following expression holds

$$U^\alpha(\theta^*) \;=\; \frac{\partial \ell}{\partial \theta^\alpha}\bigg|_{\theta^* = 0} \tag{A.20a}$$

Replacing this into (4.7) gives

$$\Sigma^*_{\alpha,\beta} \;=\; \mathbb{E}_{\theta^*}\left[\frac{\partial \ell}{\partial \theta^\alpha}\frac{\partial \ell}{\partial \theta^\beta}\right]_{\theta^* = 0} \;=\; -\,\mathbb{E}_{\theta^*}\,\frac{\partial^2 \ell}{\partial \theta^\alpha \partial \theta^\beta}\bigg|_{\theta^\alpha = 0} \;=\; \frac{\partial^2 D}{\partial \theta^\alpha \partial \theta^\beta}\bigg|_{\theta^\alpha = 0} \tag{A.20b}$$

where the second equality is the so-called Fisher's identity (see [3], Page 28), and the third equality follows from (4.2) by differentiating under the expectation. Now, by (4.8) and (A.19b), Σ^* is the identity matrix.

To show that the online estimates $\theta^{(n)}$ are asymptotically efficient, let $(\tau^\alpha\,; \alpha = 1, \ldots, d)$ be any local coordinates with origin at θ^* and let $\tau^\alpha_n = \tau^\alpha(\theta^{(n)})$. From the second-order Taylor expansion of each coordinate function τ^α, it is straightforward to show that

$$n^{1/2}\tau^\alpha_n \;=\; \left(\frac{\partial \tau^\alpha}{\partial \theta^\gamma}\right)_{\theta^*} \big(n^{1/2}(\theta^{(n)})^\gamma\big) \;+\; \sigma^\alpha(\theta^{(n)})\,\big(n^{1/2} d^2(\theta^{(n)}, \theta^*)\big) \tag{A.21a}$$

where the subscript θ^* indicates the derivative is evaluated at θ^*, and where σ^α is a continuous function in the neighborhood of θ^*. By (4.10), the second term in (A.21a) converges to zero almost surely. Therefore, the limit distribution of the re-scaled coordinates $(n^{1/2}\tau^\alpha_n)$ is the same as that of the first term in (A.21a). By (ii), this is a centred d-variate normal distribution with covariance matrix Σ^τ given by

$$\Sigma^\tau_{\alpha,\beta} \;=\; \left(\frac{\partial \tau^\alpha}{\partial \theta^\gamma}\right)_{\theta^*} \Sigma^*_{\gamma,\kappa} \left(\frac{\partial \tau^\beta}{\partial \theta^\kappa}\right)_{\theta^*} \;=\; \left(\frac{\partial \tau^\alpha}{\partial \theta^\gamma}\right)_{\theta^*} \left(\frac{\partial \tau^\beta}{\partial \theta^\gamma}\right)_{\theta^*} \tag{A.21b}$$

where the second equality follows because $\Sigma^*_{\gamma,\kappa} = \delta_{\gamma,\kappa}$ since Σ^* is the identity matrix.

It remains to show that Σ^τ is the inverse of the Fisher information matrix I^τ as in (A.34). According to (A.32), this is given by

$$I^\tau_{\alpha,\beta} \;=\; \frac{\partial^2 D}{\partial \tau^\alpha \partial \tau^\beta}\bigg|_{\tau^\alpha = 0} \;=\; -\,\mathbb{E}_{\theta^*}\,\frac{\partial^2 \ell}{\partial \tau^\alpha \partial \tau^\beta}\bigg|_{\tau^\alpha = 0} \;=\; \mathbb{E}_{\theta^*}\left[\frac{\partial \ell}{\partial \tau^\alpha}\frac{\partial \ell}{\partial \tau^\beta}\right]_{\tau^\alpha = 0} \tag{A.21c}$$

where the second equality follows from (4.2), and the third equality from Fisher's identity (see [3], Page 28). Now, a direct application of the chain rule yields the following

$$I^{\tau}_{\alpha,\beta} \;=\; \mathbb{E}_{\theta^*}\left[\frac{\partial \ell}{\partial \tau^{\alpha}}\frac{\partial \ell}{\partial \tau^{\beta}}\right]_{\tau^{\alpha}=0} \;=\; \left(\frac{\partial \theta^{\gamma}}{\partial \tau^{\alpha}}\right)_{\theta^*} \mathbb{E}_{\theta^*}\left[\frac{\partial \ell}{\partial \theta^{\gamma}}\frac{\partial \ell}{\partial \theta^{\kappa}}\right]_{\theta^{\gamma}=0}\left(\frac{\partial \theta^{\kappa}}{\partial \tau^{\beta}}\right)_{\theta^*}$$

By the first equality in (A.20b), this is equal to

$$I^{\tau}_{\alpha,\beta} \;=\; \left(\frac{\partial \theta^{\gamma}}{\partial \tau^{\alpha}}\right)_{\theta^*} \Sigma^{*}_{\gamma,\kappa}\left(\frac{\partial \theta^{\kappa}}{\partial \tau^{\beta}}\right)_{\theta^*} \;=\; \left(\frac{\partial \theta^{\gamma}}{\partial \tau^{\alpha}}\right)_{\theta^*}\left(\frac{\partial \theta^{\gamma}}{\partial \tau^{\beta}}\right)_{\theta^*} \tag{A.21d}$$

because $\Sigma^{*}_{\gamma,\kappa} = \delta_{\gamma,\kappa}$ is the identity matrix. Comparing (A.21b) to (A.21d), it is clear that Σ^{τ} is the inverse of the Fisher information matrix I^{τ} as in (A.34). *Proof of (iv)*: (4.12a) and (4.12b) follow from (4.9) and (4.10), respectively, by using (A.33). Precisely, it is possible to write (A.33) in the form

$$D(\theta^{(n)}) \;=\; \frac{1}{2}\, d^2(\theta^{(n)},\theta^*) \;+\; \omega\big(\theta^{(n)}\big)\; d^2(\theta^{(n)},\theta^*) \tag{A.22a}$$

where ω is a continuous function in the neighborhood of θ^*, equal to zero at $\theta = \theta^*$. To obtain (4.12a), it is enough to take expectations in (A.22a) and note that ω is bounded above in the neighborhood of θ^*. Then, (4.12a) follows directly from (4.9).

To obtain (4.12b), it is enough to multiply (A.22a) by n^p where $p \in (0,1)$. This gives the following expression

$$n^p D(\theta^{(n)}) \;=\; \frac{1}{2}\, n^p d^2(\theta^{(n)},\theta^*)\; \left(1 + \omega\; \left(\theta^{(n)}\right)\right) \tag{A.22b}$$

From (4.10), $n^p\, d^2(\theta^{(n)},\theta^*)$ converges to zero almost surely. Moreover, by continuity of ω, it follows that $\omega\big(\theta^{(n)}\big)$ converges to $\omega(\theta^*) = 0$ almost surely. Therefore, by taking limits in (A.22b), it is readily seen that

$$\lim\, n^p D(\theta^{(n)}) \;=\; \frac{1}{2}\left(\lim\, n^p\, d^2(\theta^{(n)},\theta^*)\right)\left(1 + \lim\, \omega\big(\theta^{(n)}\big)\right) \;=\; 0 \tag{A.22c}$$

almost surely. However, this is equivalent to the statement that $D(\theta^{(n)}) = o(n^{-p})$ for $p \in (0,1)$, almost surely. Thus, (4.12b) is proved.

A.2 Proofs of geometric lemmas

A.2.1 Lemma 1

Let $c(t)$ be the geodesic connecting θ^* to some $\theta \in \Theta^*$, parameterised by arc length. In other words, $c(0) = \theta^*$ and $c(t_\theta) = \theta$ where $t_\theta = d(\theta,\theta^*)$. Denote Π_t the parallel transport along $c(t)$, from $T_{c(0)}\Theta$ to $T_{c(t)}\Theta$. Since the velocity $\dot{c}(t)$ is self-parallel [62],

$$\dot{c}(t_\theta) \;=\; \Pi_{t_\theta}(\dot{c}(0))$$

Multiplying this identity by $-t_\theta$, it follows that

$$\mathrm{Exp}^{-1}_{\theta}(\theta^*) \;=\; -\,\Pi_{t_\theta}\big(\mathrm{Exp}^{-1}_{\theta^*}(\theta)\big) \tag{A.23a}$$

Moreover, recall the first-order Taylor expansion of the gradient $\nabla D(\theta)$ [19, 62]

$$\nabla D(\theta) \;=\; \Pi_{t_\theta}\,\big(\nabla D(\theta^*) + t_\theta\,\nabla^2 D(\theta^*)\cdot \dot c(0) + t_\theta\,\phi(\theta)\big) \tag{A.23b}$$

where $\phi(\theta)$ is continuous and equal to zero at $\theta = \theta^*$. Here, $\nabla^2 D(\theta^*)$ is the Hessian of $D(\theta)$ at $\theta = \theta^*$, considered as a linear mapping of $T_{\theta^*}\Theta$ [19, 62]

$$\nabla^2 D(\theta^*)\cdot w \;=\; \nabla_w \nabla D(\theta^*) \qquad \text{for}\;\; w \in T_{\theta^*}\Theta$$

where ∇_w denotes the covariant derivative in the direction of w. By (d1), the first term on the right-hand side of (A.23b) is equal to zero, so that

$$\nabla D(\theta) \;=\; \Pi_{t_\theta}\big(\nabla^2 D(\theta^*)\cdot \mathrm{Exp}_{\theta^*}^{-1}(\theta) + t_\theta\,\phi(\theta)\big) \tag{A.23c}$$

Taking the scalar product of (A.23a) and (A.23c),

$$\langle \mathrm{Exp}_\theta^{-1}(\theta^*), \nabla D(\theta)\rangle \;=\; -\,\langle \mathrm{Exp}_{\theta^*}^{-1}(\theta), \nabla^2 D(\theta^*)\cdot \mathrm{Exp}_{\theta^*}^{-1}(\theta)\rangle \;-\; t_\theta\,\langle \mathrm{Exp}_{\theta^*}^{-1}(\theta), \phi(\theta)\rangle \tag{A.23d}$$

since parallel transport preserves scalar products. In terms of the normal coordinates θ^α, this reads [62]

$$\langle \mathrm{Exp}_\theta^{-1}(\theta^*), \nabla D(\theta)\rangle \;=\; -\,H_{\alpha,\beta}\,\theta^\alpha\theta^\beta \;-\; t_\theta^2\;\hat\theta^\alpha\phi^\alpha \tag{A.23e}$$

where $H = (H_{\alpha,\beta})$ was defined in (4.8), $\hat\theta^\alpha$ denotes the quotient θ^α/t_θ, and the ϕ^α denote the components of $\phi(\theta)$. Note that $t_\theta^2 = d^2(\theta,\theta^*) = \theta^\alpha\theta^\alpha$, so (A.23e) can be written

$$\langle \mathrm{Exp}_\theta^{-1}(\theta^*), \nabla D(\theta)\rangle \;=\; (\psi(\theta)\delta_{\alpha,\beta} \;-\; H_{\alpha,\beta})\;\theta^\alpha\theta^\beta \tag{A.23f}$$

where $\psi(\theta)$ is continuous and equal to zero at $\theta = \theta^*$. To conclude, let $\mu = \lambda - \varepsilon$ for some $\varepsilon > 0$, and $\bar\Theta^*$ a neighborhood of θ^*, contained in Θ^*, such that $\psi(\theta) \leqslant \varepsilon$ for $\theta \in \bar\Theta^*$. Then, since λ is the smallest eigenvalue of $H = (H_{\alpha,\beta})$,

$$\langle \mathrm{Exp}_\theta^{-1}(\theta^*), \nabla D(\theta)\rangle \;\leqslant\; (\varepsilon - \lambda)\;\theta^\alpha\theta^\alpha \;=\; -\mu\, d^2(\theta,\theta^*)$$

for $\theta \in \bar\Theta^*$. This is exactly (A.9a), so the lemma is proved. □

A.2.2 Lemma 3

To simplify notation, let $U_{n+1} = U(\theta^{(n)}, x_{n+1})$. Then, the geodesic $c(t)$, connecting $\theta^{(n)}$ to $\theta^{(n+1)}$, has equation

$$c(t) \;=\; \mathrm{Exp}_{\theta^{(n)}}\big(t\eta^{(n+1)}U_{n+1}\big)$$

Each one of the normal coordinates θ^α is a C^3 function $\theta^\alpha : \Theta^* \to \mathbb{R}$, with differential $d\theta^\alpha$ and Hessian [62]

$$\nabla^2\theta^\alpha \;=\; -\Gamma^\alpha_{\beta,\gamma}(\theta)\,\mathrm{d}\theta^\beta \,\otimes\, \mathrm{d}\theta^\gamma$$

where $\Gamma^\alpha_{\beta,\gamma}$ are the Christoffel symbols of the coordinates θ^α, and $\otimes$ denotes the tensor product. Then, the second-order Taylor expansion of the functions $\theta^\alpha \circ c$ reads

$$\begin{aligned}(\theta^\alpha\circ c)(1) =& (\theta^\alpha\circ c)(0) + \eta^{(n+1)}\,\mathrm{d}\theta^\alpha(U_{n+1}) \\ &- \frac{1}{2}[\eta^{(n+1)}]^2\,\Gamma^\alpha_{\beta,\gamma}(\theta^{(n)})\,\mathrm{d}\theta^\beta(U_{n+1})\,\mathrm{d}\theta^\gamma(U_{n+1}) + \,[\eta^{(n+1)}]^3 T^\alpha_{n+1}\end{aligned} \tag{A.24a}$$

where T^{α}_{n+1} satisfies

$$\left|T^{\alpha}_{n+1}\right| \leqslant K_1 \left\|U_{n+1}\right\|^3 \tag{A.24b}$$

for a constant K_1 which does not depend on n, as can be shown by direct calculation. Of course, $(\theta^{\alpha} \circ c)(1) = (\theta^{(n+1)})^{\alpha}$ and $(\theta^{\alpha} \circ c)(0) = (\theta^{(n)})^{\alpha}$. Moreover, $\mathrm{d}\theta^{\alpha}(U_{n+1}) = U^{\alpha}_{n+1}$ are the components of U_{n+1}. Replacing into (A.24a), this yields

$$(\theta^{(n+1)})^{\alpha} = (\theta^{(n)})^{\alpha} + \eta^{(n+1)}\, U^{\alpha}_{n+1} + [\eta^{(n+1)}]^2\, \pi^{\alpha}_{n+1} \tag{A.24c}$$

where π^{α}_{n+1} is given by

$$\pi^{\alpha}_{n+1} = [\eta^{(n+1)}]^2\, T^{\alpha}_{n+1} - \frac{1}{2}\, \Gamma^{\alpha}_{\beta,\gamma}(\theta^{(n)})\, U^{\beta}_{n+1}\, U^{\gamma}_{n+1} \tag{A.24d}$$

Comparing (A.24c) to (A.13a), it is clear the proof will be complete upon showing $\mathbb{E}\left|\pi^{\alpha}_{n+1}\right| = \mathcal{O}(n^{-1/2})$. To do so, note that each Christoffel symbol $\Gamma^{\alpha}_{\beta,\gamma}$ is a C^1 function on the compact set Θ^*, with $\Gamma^{\alpha}_{\beta,\gamma}(\theta^*) = 0$ by the definition of normal coordinates [62]. Therefore,

$$\left|\, \Gamma^{\alpha}_{\beta,\gamma}(\theta) \,\right| \leqslant K_2\, d(\theta, \theta^*) \tag{A.24e}$$

for a constant K_2 which does not depend on n. Replacing the inequalities (A.24b) and (A.24e) into (A.24d), and taking expectations, it follows that

$$\mathbb{E}\left|\pi^{\alpha}_{n+1}\right| \leqslant \eta^{(n+1)}\, K_1\, \mathbb{E}\left\|U_{n+1}\right\|^3 + d^2 \times K_2\, \mathbb{E}\left[\, d(\theta^{(n)}, \theta^*)\left\|U_{n+1}\right\|^2 \right] \tag{A.25a}$$

where d is the dimension of the parameter space Θ. However, using the fact that the x_n are i.i.d. with distribution P_{θ^*},

$$\mathbb{E}\left[\, \left\|U_{n+1}\right\|^3 \middle| \mathcal{X}_n \right] = \mathbb{E}_{\theta^*} \left\|U(\theta^{(n)}, x)\right\|^3 \leqslant R^{3/4}(\theta^{(n)}) \tag{A.25b}$$

by (u2) and Jensen's inequality [72]. On the other hand, by the Cauchy-Schwarz inequality,

$$\mathbb{E}\left[\, d(\theta^{(n)}, \theta^*)\left\|U_{n+1}\right\|^2 \right] \leqslant \left(\mathbb{E}\, d^2(\theta^{(n)}, \theta^*)\right)^{1/2} \left(\mathbb{E}\left\|U_{n+1}\right\|^4\right)^{1/2} \leqslant b\, n^{-1/2} \left(\mathbb{E}\left\|U_{n+1}\right\|^4\right)^{1/2}$$

for some $b > 0$ as follows from (4.9). Then, by the same reasoning that lead to (A.25b),

$$\mathbb{E}\left[\, d(\theta^{(n)}, \theta^*)\left\|U_{n+1}\right\|^2 \right] \leq b\, n^{-1/2} \left(\mathbb{E}\, R(\theta^{(n)})\right)^{1/2} \tag{A.25c}$$

By (u2), there exists a uniform upper bound M on $R(\theta)$ for $\theta \in \Theta^*$. Since $\theta^{(n)}$ lies in Θ^* for all n, it follows by replacing the inequalities (A.25b) and (A.25c) into (A.25a) that

$$\mathbb{E}\left|\pi^{\alpha}_{n+1}\right| \leqslant \eta^{(n+1)}\, K_1\, M^{3/4} + d^2 \times K_2\, b\, n^{-1/2} M^{1/2} \tag{A.25d}$$

Finally, by recalling that $\eta^{(n)} = \frac{a}{n}$, it is clear that the right-hand side of (A.25d) is $\mathcal{O}(n^{-1/2})$, so the proof is complete. □

A.2.3 Lemma 4

The lemma is an instance of the general statement : let $\theta \in \Theta^*$ and $v = \nabla D(\theta)$. Then, in a system of normal coordinates with origin at θ^*,

$$v^\alpha \;=\; H_{\alpha,\beta}\,\theta^\beta \;+\; o\,(d(\theta,\theta^*)) \tag{A.26a}$$

where v^α are the components of v. Indeed, (A.13b) follows from (A.26a) after replacing $\theta = \theta^{(n)}$, so that $v = v_n$, and setting

$$\rho_n^\alpha \;=\; v_n^\alpha \;-\; H_{\alpha,\beta}\,(\theta^{(n)})^\beta$$

To prove (A.26a), recall (A.23c) from the proof of Lemma 1, which can be written

$$v \;=\; \Pi_{t_\theta}\big(\nabla^2 D(\theta^*)\cdot \mathrm{Exp}_{\theta^*}^{-1}(\theta)\big) \;+\; d(\theta,\theta^*)\,\Pi_{t_\theta}(\phi(\theta)) \tag{A.26b}$$

Denote $\partial_\alpha = \frac{\partial}{\partial\theta^\alpha}$ the coordinate vector fields of the normal coordinates θ^α. Note that [19, 62]

$$\mathrm{Exp}_{\theta^*}^{-1}(\theta) \;=\; \theta^\beta \partial_\beta(\theta^*) \qquad\qquad \nabla^2 D(\theta^*)\cdot\partial_\beta(\theta^*) \;=\; H_{\alpha,\beta}\,\partial_\alpha(\theta^*)$$

Replacing in (A.26b), this gives

$$v \;=\; H_{\alpha,\beta}\,\theta^\beta\;\Pi_{t_\theta}(\partial_\alpha(\theta^*)) \;+\; d(\theta,\theta^*)\,\Pi_{t_\theta}(\phi(\theta)) \tag{A.26c}$$

From the first-order Taylor expansion of the vector fields ∂_α [19, 62]

$$\partial_\alpha(\theta) \;=\; \Pi_{t_\theta}\big(\partial_\alpha(\theta^*) \;+\; \nabla\partial_\alpha(\theta^*)\cdot\mathrm{Exp}_{\theta^*}^{-1}(\theta)\big) \;+\; d(\theta,\theta^*)\,\Pi_{t_\theta}(\chi^\alpha(\theta))$$

where $\chi^\alpha(\theta)$ is continuous and equal to zero at $\theta = \theta^*$. However, by the definition of normal coordinates [62], each covariant derivative $\nabla\partial_\alpha(\theta^*)$ is zero. In other words,

$$\partial_\alpha(\theta) \;=\; \Pi_{t_\theta}(\partial_\alpha(\theta^*)) \;+\; d(\theta,\theta^*)\,\Pi_{t_\theta}(\chi^\alpha(\theta)) \tag{A.26d}$$

Replacing (A.26d) into (A.26c), it follows

$$v \;=\; H_{\alpha\beta}\,\theta^\beta\,\partial_\alpha(\theta) \;+\; d(\theta,\theta^*)\,\Pi_{t_\theta}\big(\phi(\theta) - H_{\alpha\beta}\,\theta^\beta\chi^\alpha(\theta)\big) \tag{A.26e}$$

Now, to obtain (A.26a), it is enough to note the decomposition $v = v^\alpha\,\partial_\alpha(\theta)$ is unique, and $\phi(\theta) - H_{\alpha\beta}\,\theta^\beta\chi^\alpha(\theta)$ converges to zero as θ converges to θ^*. □

A.3 Conditions of the martingale CLT

For the verification of Conditions (A.17), the following inequality (A.27) will be useful. Let $\nu = a\lambda - \frac{1}{2}$, so $-\nu$ is the largest eigenvalue of the matrix A in (A.15a). There exists a constant C_A such that the transition matrices $A_{n,k}$ in (A.15c) satisfy [45, 58]

$$|A_{n,k}|_{\mathrm{Op}} \;\le C_A \left(\frac{k}{n}\right)^\nu \tag{A.27}$$

where $|A_{n,k}|_{\mathrm{Op}}$ denotes the Euclidean operator norm, equal to the largest singular value of the matrix $A_{n,k}$.

Condition (A.17a): to verify this condition, note that for arbitrary $\varepsilon > 0$,

$$\mathbb{P}\left(\max_{k\leq n}\left|A_{n,k}\,\frac{aw_k}{k^{1/2}}\right| > \varepsilon\right) \leq \sum_{k=1}^{n}\mathbb{P}\left(\left|A_{n,k}\,\frac{aw_k}{k^{1/2}}\right| > \varepsilon\right) \leq \sum_{k=1}^{n}\mathbb{P}\left(C_A\left(\frac{k}{n}\right)^{\nu}\left|\frac{aw_k}{k^{1/2}}\right| > \varepsilon\right) \tag{A.28a}$$

where the second inequality follows from (A.27). However, it follows from (u2) that there exists a uniform upper bound M_w on the fourth-order moments of $|w_k|$. Therefore, by Chebyshev's inequality [72]

$$\sum_{k=1}^{n}\mathbb{P}\left(C_A\left(\frac{k}{n}\right)^{\nu}\left|\frac{aw_k}{k^{1/2}}\right| > \varepsilon\right) \leq \left(\frac{aC_A}{\varepsilon}\right)^4\frac{M_w}{n^{4\nu}}\sum_{k=1}^{n}k^{4\nu-2} \tag{A.28b}$$

Since $\nu > 0$, the right-hand side of (A.28b) has limit equal to 0 as $n \to \infty$, by the Euler-Maclaurin formula [22]. Replacing this limit from (A.28b) into (A.28a) immediately yields Condition (A.17a). □

Condition (A.17b): to verify this condition, recall that (w_k) is a sequence of square-integrable martingale differences. Therefore, from (A.16)

$$\mathbb{E}\left|\tilde{g}_n\right|^2 = \sum_{k=1}^{n}\frac{a^2}{k}\,\mathbb{E}\,\mathrm{tr}\big(A_{n,k}^2\Sigma_k\big) \tag{A.29a}$$

where Σ_k is the conditional covariance matrix in (A.18). Applying (A.27) to each term under the sum in (A.29a), it follows that

$$\mathbb{E}\left|\tilde{g}_n\right|^2 \leqslant d^{\frac{1}{2}}\sum_{k=1}^{n}\frac{a^2}{k}\,\mathbb{E}\,|A_{n,k}|_{\mathrm{Op}}^2\left|\Sigma_k\right|_{\mathrm{F}} \leqslant \left(d^{\frac{1}{2}}\,a^2\,C_A^2\right)\frac{1}{n^{2\nu}}\sum_{k=1}^{n}k^{2\nu-1}\,\mathbb{E}\left|\Sigma_k\right|_{\mathrm{F}} \tag{A.29b}$$

where d is the dimension of the parameter space Θ, and $|\Sigma_k|_{\mathrm{F}}$ denotes the Frobenius matrix norm. However, it follows from (u1) that there exists a uniform upper bound S on $|\Sigma_k|_{\mathrm{F}}$. Therefore, by (A.29b)

$$\mathbb{E}\left|\tilde{g}_n\right|^2 \leqslant \left(d^{\frac{1}{2}}\,a^2\,C_A^2\right)\frac{S}{n^{2\nu}}\sum_{k=1}^{n}k^{2\nu-1} \tag{A.29c}$$

Since $\nu > 0$, the right-hand side of (A.29c) remains bounded as $n \to \infty$, by the Euler-Maclaurin formula [22]. This immediately yields Condition (A.17b). □

Condition (A.17c): to verify this condition, it is first admitted that the following limit is known to hold

$$\lim\,\mathbb{E}\,(\Sigma_k) = \Sigma^* \tag{A.30a}$$

where Σ^* was defined in (4.7). Then, let the sum in (A.17c) be written

$$\sum_{k=1}^{n}\frac{a^2}{k}\,A_{n,k}\,\Sigma_k\,A_{n,k} = \sum_{k=1}^{n}\frac{a^2}{k}\,A_{n,k}\,\Sigma^*A_{n,k} + \sum_{k=1}^{n}\frac{a^2}{k}\,A_{n,k}\left[\,\Sigma_k - \Sigma^*\right]A_{n,k} \tag{A.30b}$$

Due to the equivalence $A_{n,k} \sim \exp(\ln(n/k)A)$ (see [58], Page 125), the first term in (A.30b) is a Riemann sum for the integral [45, 58]

$$a^2\int_0^1 e^{-\ln(s)\,A}\,\Sigma^*\,e^{-\ln(s)\,A}\,d\ln(s) = a^2\int_0^{\infty}e^{-t\,A}\,\Sigma^*\,e^{-t\,A}\,dt$$

which is known to be the solution Σ of Lyapunov's equation (4.11). The second term in (A.30b) can be shown to converge to zero in probability, using inequality (A.27) and the limit (A.30a), by a similar argument to the ones in the verification of Conditions (A.17a) and (A.17b). Then, Condition (A.17c) follows immediately. □

Proof of (A.30a): recall that $w_k = U_k + v_{k-1}$ where $U_k = U(\theta^{(k-1)}, x_k)$ and $v_{k-1} = \nabla D(\theta^{(k-1)})$. Since (w_k) is a sequence of square-integrable martingale differences, it is possible to write, in the notation of (A.18),

$$\Sigma_k \,=\, \mathbb{E}\left[\, U_k U_k^\dagger \,\middle|\, \mathcal{X}_{k-1} \right] \,-\, v_{k-1} v_{k-1}^\dagger \tag{A.31a}$$

By (A.8), the second term in (A.31a) converges to zero almost surely, as $k \to \infty$. It also converges to zero in expectation, since $\nabla D(\theta)$ is uniformly bounded for θ in the compact set Θ^*. For the first term in (A.31a), since the x_k are i.i.d. with distribution P_{θ^*}, it follows that

$$\mathbb{E}\left[\, U_k U_k^\dagger \,\middle|\, \mathcal{X}_{k-1} \right] \,=\, \mathbb{E}_{\theta^*}\left[U(\theta^{(k-1)}, x) U^\dagger(\theta^{(k-1)}, x) \right] \tag{A.31b}$$

Since $U(\theta, x)$ is a continuous vector field on Θ for each $x \in X$, and $\theta^{(k-1)}$ converge to θ^* almost surely, it follows that $U(\theta^{(k-1)}, x)$ converge to $U(\theta^*, x)$ for each $x \in X$, almost surely. On the other hand, it follows from (u2) that the functions under the expectation $\mathbb{E}_{\theta^*}$ in (A.31b) have bounded second order moments, so they are uniformly integrable [72]. Therefore,

$$\lim\, \mathbb{E}_{\theta^*}\left[U(\theta^{(k-1)}, x) U^\dagger(\theta^{(k-1)}, x) \right] \,=\, \mathbb{E}_{\theta^*}\left[U(\theta^*, x) U^\dagger(\theta^*, x) \right] \,=\, \Sigma^* \tag{A.31c}$$

almost surely, by the definition (4.7) of Σ^*. It now follows from (A.31a), (A.31b), and (A.31c) that the following limit holds

$$\lim\, \Sigma_k \,=\, \Sigma^* \qquad \text{almost surely} \tag{A.31d}$$

To obtain (A.30a) it is enough to note, as already stated in the verification of Condition (A.17b), that the Σ_k are uniformly bounded in the Frobenius matrix norm. Thus, (A.31d) implies (A.30a), by the dominated convergence theorem. □

A.4 Background on the Fisher information metric

Let $D(\theta)$ be the Kullback-Leibler divergence (4.2), or any other so-called α-divergence [3]. Assume the Riemannian metric $\langle \cdot, \cdot \rangle$ of Θ coincides with the Fisher information metric of the model P. Then, for any local coordinates $(\tau^\alpha \,; \alpha = 1, \ldots, d\,)$, with origin at θ^*, the following relation holds, by definition of the Fisher information metric (see [3], Page 54),

$$\left.\frac{\partial^2 D}{\partial \tau^\alpha \partial \tau^\beta}\right|_{\tau^\alpha = 0} \,=\, \left\langle \frac{\partial}{\partial \tau^\alpha}, \frac{\partial}{\partial \tau^\beta} \right\rangle_{\theta^*} \tag{A.32}$$

where $\frac{\partial}{\partial \tau^\alpha}$ denote the coordinate vector fields of the local coordinates τ^α. It is also possible to express (A.32) in terms of the Riemannian distance $d(\cdot, \cdot)$, induced by the Fisher information metric $\langle \cdot, \cdot \rangle$. Precisely,

$$D(\theta) \,=\, \frac{1}{2}\, d^2(\theta, \theta^*) \,+\, o\left(d^2(\theta, \theta^*)\right) \tag{A.33}$$

This follows immediately from the second-order Taylor expansion of $D(\theta)$, since θ^* is a minimum of $D(\theta)$, by using (A.32). Formula (A.33) shows that the divergence $D(\theta)$ is equivalent to half the squared Riemannian distance $d^2(\theta, \theta^*)$, at $\theta = \theta^*$.

The scalar products appearing in (A.32) form the components of the Fisher information matrix I^τ of the coordinates τ^α,

$$I^\tau_{\alpha\beta} \;=\; \left.\frac{\partial^2 D}{\partial\tau^\alpha \partial\tau^\beta}\right|_{\tau^\alpha=0}$$

In any change of coordinates, these transform like the components of a $(0,2)$ covariant tensor [62]. That is, if $(\theta^{\,\alpha}; \alpha = 1, \ldots, d\,)$ are any local coordinates defined at θ^*,

$$I^\tau_{\alpha\beta} \;=\; \left(\frac{\partial\theta^{\,\gamma}}{\partial\tau^\alpha}\right)_{\theta^*} I^\theta_{\gamma\kappa} \left(\frac{\partial\theta^{\,\kappa}}{\partial\tau^\beta}\right)_{\theta^*}$$

where the subscript θ^* indicates the derivative is evaluated at θ^*, and where $I^\theta_{\gamma\kappa}$ are the components of the Fisher information matrix I^θ of the coordinates $\theta^{\,\alpha}$.

The recursive estimates $\theta^{(n)}$ are said to be asymptotically efficient, if they are asymptotically efficient in any local coordinates τ^α, with origin at θ^*. That is, according to the classical definition of asymptotic efficiency [44, 78], if the following weak limit of probability distributions is verified [72],

$$\mathcal{L}\left\{(n^{1/2}\tau^\alpha_n)\right\} \;\Longrightarrow\; N_d\,(0, \Sigma^\tau) \qquad\qquad \Sigma^\tau \;=\; (I^\tau)^{-1} \tag{A.34}$$

where $\mathcal{L}\{\ldots\}$ denotes the probability distribution of the quantity in braces, $\tau^\alpha_n = \tau^\alpha(\theta^{(n)})$ are the coordinates of the recursive estimates $\theta^{(n)}$, and $N_d\,(0, \Sigma^\tau)$ denotes a centred d-variate normal distribution with covariance matrix Σ^τ.

It is important to note that asymptotic efficiency of the recursive estimates $\theta^{(n)}$ is an intrinsic geometric property, which does not depend on the particular choice of local coordinates τ^α, with origin at θ^*. This can be seen from the transformation rule of the components of the Fisher information matrix, described above. In fact, since these transform like the components of a $(0,2)$ covariant tensor, the components of Σ^τ transform like those of a $(2,0)$ contravariant tensor, which is the correct transformation rule for the components of a covariance matrix.

Appendix B

Proof for chapter 5

B.1 Proof of Proposition 6

Let $(\theta^{(n)})_{n\geqslant 0}$ be a sequence generated by Algorithm 1. Recall the retraction Ret_θ is defined in (3.18). Consider the sequence of tangent vectors $\big(U(\theta^{(n)})\big)_{n\geqslant 0}$ where $U(\theta^{(n)})$ belongs to $T_{\theta^{(n)}}\Theta$, with

$$\begin{aligned} U(\theta^{(0)}) &= -\nabla^*_\mu \hat{D}(\mu^{(0)}, \Sigma^{(0)}, \beta^{(0)}) \\ U(\theta^{(1)}) &= -\nabla^*_\Sigma \hat{D}(\mu^{(1)}, \Sigma^{(0)}, \beta^{(0)}) \\ U(\theta^{(2)}) &= -\nabla^*_\beta \hat{D}(\mu^{(1)}, \Sigma^{(1)}, \beta^{(0)}) \end{aligned}$$

and so on for $U(\theta^{(n)})$ with $n \geqslant 0$.

It is easy to see that either there exist infinitely many vectors $U(\theta^{(n)})$ such that $U(\theta^{(n)}) \neq 0$, or $\theta^{(n)} = \theta^*$ for all n greater than some n_1. Indeed, if there are only finitely many $U(\theta^{(n)})$ such that $U(\theta^{(n)}) \neq 0$, then there exists n_0 such that $U(\theta^{(n)}) = 0$ for all n greater than n_0. This implies there exists $n_1 \geqslant n_0$ with

$$\begin{aligned} U(\theta^{(n_1)}) &= -\nabla^*_\mu \hat{D}(\theta^{(n_1)}) = 0 \\ U(\theta^{(n_1+1)}) &= -\nabla^*_\Sigma \hat{D}(\theta^{(n_1+1)}) = 0 \\ U(\theta^{(n_1+2)}) &= -\nabla^*_\beta \hat{D}(\theta^{(n_1+2)}) = 0 \end{aligned}$$

From Algorithm 1, this implies that $\theta^{(n_1)} = \theta^{(n_1+1)} = \theta^{(n_1+2)}$ and $\nabla^\odot_\theta \hat{D}(\theta^{(n_1)}) = 0$ (since all three components of $\nabla^\odot_\theta \hat{D}(\theta^{(n_1)})$ are zero). Since $\theta^{(n_1)} \in \Theta^*$, this implies $\theta^{(n_1)} = \theta^*$. Then (again by Algorithm 1) $\theta^{(n)} = \theta^*$ for all $n \geqslant n_1$.

Let $\big(\theta^{(n_i)}, U(\theta^{(n_i)})\big)_{i\geqslant 0}$ be the subsequence of $\big(\theta^{(n)}, U(\theta^{(n)})\big)_{n\geqslant 0}$ which consists of all the couples $\big(\theta^{(n)}, U(\theta^{(n)})\big)$ such that $U(\theta^{(n)}) \neq 0$ ($i \mapsto n_i$ is a function which counts the terms of this subsequence). Clearly, since $U(\theta^{(n)}) = 0$ means that $\theta^{(n)}$ will not be updated, it is enough to restrict attention to the subsequence $\big(\theta^{(n_i)}, U(\theta^{(n_i)})\big)_{i\geqslant 0}$. For simplicity, this will be denoted $\big(\theta^{(i)}, U(\theta^{(i)})\big)_{i\geqslant 0}$.

The subsequence $\big(\theta^{(i)}\big)_{i\geqslant 0}$ is given as in Algorithm 1 of [2], $\theta^{(i+1)} = \mathrm{Ret}_{\theta^{(i)}}\big(t_i\, U(\theta^{(i)})\big)$ with step-size t_i chosen according to Armijo-Goldstein rule (precisely, $t_i = t_{n_i}$ where t_n is given by $t_0 = \eta_\mu$, $t_1 = \eta_\Sigma$, $t_2 = \eta_\beta$, *etc.*). Moreover, $(\theta^{(i)})_{i\geqslant 0}$ remains within the compact neighborhood Θ^* of θ^*. According to Corollary 4.3.2 in [2], if $(\eta^{(i)})_{i\geqslant 0}$ is gradient-related then $\lim_{i\to\infty} \|\nabla^\odot_\theta \hat{D}(\theta^{(i)})\|_\odot = 0$.

Then, since θ^* is the only stationary point of the cost function (4.6) in Θ^*, it follows that $\lim_{i\to\infty} \theta^{(i)} = \theta^*$, as required. To show that the subsequence $\big(U(\theta^{(i)})\big)_{i\geqslant 0}$ is gradient-

related, note that

$$\left\langle U(\theta^{(0)}), \nabla_\theta^\odot \hat{D}(\theta^{(0)}) \right\rangle_\odot = -\left\| \nabla_\mu^* \hat{D}(\mu^{(0)}, \Sigma^{(0)}, \beta^{(0)}) \right\|_*^2$$
$$\left\langle U(\theta^{(1)}), \nabla_\theta^\odot \hat{D}(\theta^{(1)}) \right\rangle_\odot = -\left\| \nabla_\Sigma^* \hat{D}(\mu^{(1)}, \Sigma^{(0)}, \beta^{(0)}) \right\|_*^2$$
$$\left\langle U(\theta^{(2)}), \nabla_\theta^\odot \hat{D}(\theta^{(2)}) \right\rangle_\odot = -\left\| \nabla_\beta^* \hat{D}(\mu^{(1)}, \Sigma^{(1)}, \beta^{(0)}) \right\|_*^2$$

and so on, for $n \geqslant 3$. Therefore, $\left\langle U(\theta^{(n)}), \nabla_\theta^\odot \hat{D}(\theta^{(n)}) \right\rangle_\odot < 0$ whenever $U(\theta^{(n)}) \neq 0$. This means that $\left\langle U(\theta^{(i)}), \nabla_\theta^\odot \hat{D}(\theta^{(i)}) \right\rangle_\odot < 0$ for the whole subsequence $\left(U(\theta^{(i)})\right)_{i \geqslant 0}$, and this subsequence is indeed gradient related.

B.2 Proof of Proposition 7

The proof is a direct application of Remark 2, concerning Proposition 1, in [88]. According to this remark, if $U\left(\theta^{(n)}, \mathcal{X}_{mb}^{(n+1)}\right)$ denotes the direction of descent, and if

$$\mathbb{E}_n \left\langle U(\theta^{(n)}, \mathcal{X}_{mb}^{(n+1)}), \nabla_\theta^\odot D(\theta^{(n)}) \right\rangle_\odot < 0, \text{ almost surely, for } n > 0 \tag{B.1}$$

where $\mathbb{E}_i$ denotes conditional expectation with respect to $(\mathcal{X}_{mb}^{(0)}, \mathcal{X}_{mb}^{(1)}, \cdots, \mathcal{X}_{mb}^{(i)}, \cdots)$, then $\lim \theta^{(n)} = \theta^*$ almost surely. Here (compare to the proof of Proposition 6), the direction of descent is given by

$$U(\theta^{(0)}, \mathcal{X}_{mb}^{(1)}) = \nabla_\mu^* \ell_p(\mu^{(0)}, \Sigma^{(0)}, \beta^{(0)}; \mathcal{X}_{mb}^{(1)})$$
$$U(\theta^{(1)}, \mathcal{X}_{mb}^{(2)}) = \nabla_\Sigma^* \ell_p(\mu^{(1)}, \Sigma^{(0)}, \beta^{(0)}; \mathcal{X}_{mb}^{(2)})$$
$$U(\theta^{(2)}, \mathcal{X}_{mb}^{(3)}) = \nabla_\beta^* \ell_p(\mu^{(1)}, \Sigma^{(1)}, \beta^{(0)}; \mathcal{X}_{mb}^{(3)})$$

and so on. Therefore, the expectations in (B.1) can be found from

$$\mathbb{E}_0 \left\langle U(\theta^{(0)}, \mathcal{X}_{mb}^{(1)}), \nabla_\theta^\odot D(\theta^{(0)}) \right\rangle_\odot = -\|\nabla_\mu D(\theta^{(0)})\|_*^2 \tag{B.2a}$$

$$\mathbb{E}_1 \left\langle U(\theta^{(1)}, \mathcal{X}_{mb}^{(2)}), \nabla_\theta^\odot D(\theta^{(1)}) \right\rangle_\odot = -\|\nabla_\Sigma D(\theta^{(1)})\|_*^2 \tag{B.2b}$$

$$\mathbb{E}_2 \left\langle U(\theta^{(2)}, \mathcal{X}_{mb}^{(3)}), \nabla_\theta^\odot D(\theta^{(2)}) \right\rangle_\odot = -\|\nabla_\beta D(\theta^{(2)})\|_*^2 \tag{B.2c}$$

and so on, for $n \geqslant 3$. Thus, the expectation in (B.1) is always negative. Recall the assumption A1 in Proposition 7.

A1. θ^* is the unique stationary point of $D(\theta)$ in Θ^*. Moreover, the second derivatives $\nabla_\mu^{*^2} D(\theta^*)$, $\nabla_\Sigma^{*^2} D(\theta^*)$, $\nabla_\beta^{*^2} D(\theta^*)$ are all positive-definite.

We now show that Assumption A1 guarantees it is strictly negative, as required.

By assumption A1, and a direct application of the implicit function theorem (after taking Θ^* sufficiently small) [1], there exist three submanifolds of Θ, which pass through θ^*,

$$H_\mu = \{\theta \in \Theta^* : \nabla_\mu^* D(\theta) = 0\}$$
$$H_\Sigma = \{\theta \in \Theta^* : \nabla_\Sigma^* D(\theta) = 0\}$$
$$H_\beta = \{\theta \in \Theta^* : \nabla_\beta^* D(\theta) = 0\}$$

Moreover, H_μ is uniquely parameterised by (Σ, β), H_Σ is uniquely parameterised by (μ, β) and H_β is uniquely parameterised by (μ, Σ). In particular, this implies that H_μ, H_Σ and H_β are submanifolds of lower dimension, and therefore have zero Riemannian volume.

The expectations in (B.2) are therefore almost surely all strictly negative. Indeed, each θ_n with $n > 0$ has a probability density function with respect to Riemannian volume, and therefore almost surely does not belong to H_μ, H_Σ or H_β.

B.3 Proof of Propositions 8 and 9

As for Proposition 7, this is an application of Remark 2 in [88]. According to this remark, in order to obtain the mean-square rate and the asymptotic normality, it is enough to show the mean vector field $X(\theta) = \mathbb{E}_{\theta^*}[U(\theta, x)]$ has an attractive stationary point at $\theta = \theta^*$. Since $U(\theta, x) = \nabla^{\odot}_{\theta} \ell(\theta; x)$

$$\mathbb{E}_{\theta^*}[U(\theta, x)] = \begin{bmatrix} -\nabla^*_\mu D(\theta) \\ -\nabla^*_\Sigma D(\theta) \\ -\nabla^*_\beta D(\theta) \end{bmatrix} \tag{B.3}$$

The covariant derivative of this vector field at the point $\theta = \theta^*$ is equal to the Hessian $\mathcal{H}(\theta^*)$, which is positive-definite. Therefore, the results of Propositions 8 and 9 follow by Remark 2 in [88].

B.4 Proof of Proposition 10

For the case of $\theta = (\Sigma)$, the geodesic convexity of the cost function $D(\theta)$ (or of $\hat{D}(\theta)$) follows by proving $-\ell_p(\theta; x)$ is geodesically strictly convex in $\theta = (\Sigma)$ for any x.

Recall that, geodesic curves on $\mathcal{P}_m$ are of the form [61]

$$\begin{array}{rcl} \gamma : \mathbb{R} & \to & \mathcal{P}_m \\ t & \mapsto & A \exp(tr) A^\dagger \end{array} \tag{B.4}$$

where exp denotes the matrix exponential map, A is an invertible matrix, and r is a diagonal matrix, both of same size as Σ. Then, $-\ell_p(\theta)$ is geodesically convex if and only if the composition $(-\ell_p \circ \gamma)(t)$ is always a convex function with respect to t. Moreover, geodesic strict convexity is defined in exactly the same way. The composition $(-\ell_p \circ \gamma)(t)$ can be expressed

$$(-\ell_p \circ \gamma)(t) = \log\det(A) + \frac{1}{2}\mathrm{tr}(r)t + \log\left[(\mathfrak{f} \circ \varphi)(t)\right] \tag{B.5}$$

recall that, the funtion $\mathfrak{f}$ is defined in equation (5.7) and (5.8), and here

$$\varphi(t) = \sum_{i=1}^{m} u_i^2 \exp(-r_i t) \tag{B.6}$$

with $u = A^{-1}x$, its components u_i, and r_i are the diagonal elements of r. The function $\varphi : \mathbb{R} \to \mathbb{R}_+$ is strictly log-convex, because it is the Laplace transform of a positive measure [72]

$$\varphi(t) = \int_0^\infty \exp(-tx)\mu(\mathrm{d}x) \tag{B.7}$$

where $\mu = \sum_{i=1}^{m} u_i^2 \delta_{r_i}$, and δ_{r_i} is the Dirac measure concentrated at r_i.

Assume that the function $\mathfrak{f}$ verifies Condition (5.9). Then, since φ is strictly log-convex, $\mathfrak{f} \circ \varphi$ is strictly log-convex. Thus, the term $\log\left[(\mathfrak{f} \circ \varphi)(t)\right]$ of (B.5) is a strictly convex function of the real variable t. Since the term $\mathrm{tr}(r)\frac{t}{2}$ of (B.5) amounts to an affine function of t, it is now clear that $(-\ell_p \circ \gamma)(t)$ is a strictly convex function of the real variable t, for any geodesic curve $\gamma : \mathbb{R} \to \mathcal{P}_m$. Finally, since x was chosen arbitrarily, $-\ell_p(\theta; x)$ is geodesically strictly convex in $\theta = (\Sigma)$ for each x. Therefore, $D(\theta)$ and $\hat{D}(\theta)$ are both geodesically strictly convex.

B.5 Proof of Corollary 2 and 3

For the case of $\theta = (\Sigma)$, note that $\varphi : \mathbb{R} \to \mathbb{R}_+$ is strictly log-convex if and only if $\varphi(t) = \exp(\psi(t))$ where $\psi : \mathbb{R} \to \mathbb{R}$ is strictly convex.
1) plugging equation (5.8a) into equation (5.9),

$$\log(\mathfrak{f} \circ \varphi)(t) = \frac{1}{2} \exp\left(\beta\left(\psi(t)\right)\right) \tag{B.8}$$

Therefore, condition (5.9) is verified since $\beta > 0$.
2) plugging equation (5.8b) into equation (5.9),

$$\log\left(\mathfrak{f} \circ \varphi\right)(t) = \frac{\beta + m}{2} \log\left(1 + \frac{\exp(\psi(t))}{\beta}\right) \tag{B.9}$$

Therefore, condition (5.9) is verified since $\beta + m > 0$.

For the case of $\theta = (\mu, \Sigma)$, as mentioned above, the function $\tilde{\mathfrak{f}}$ is reformulated. Then, the same strategy is applied for this reformulated $\tilde{\mathfrak{f}}$.
1) For MGGD, recall the geodesic curve for reformulated matrix $\mathcal{S}(t)$,

$$\mathcal{S}(t) = B \exp(st) B^\dagger$$

where exp denotes the matrix exponential map, B is an invertible matrix, and s is a diagonal matrix, both of same size as $\mathcal{S}$.

$$\delta_y(t) = y^\dagger \mathcal{S}^{-1} y = \sum_{i=1}^{m+1} v_i^2 e^{-s_i t} \qquad \text{with } v = B^{-1} y \tag{B.10}$$

According to equation (B.10), we have $\delta_y > 1$. Therefore, $\exists w \in \mathbb{R}^{m+1}$ and $\exists q \in (0, +\infty)^{m+1}$ (e.g. $w = (u, 0)$ and $q = (r, 1)$) such that

$$\sum_{i=1}^{m+1} v_i^2 e^{-s_i t} = \sum_{i=1}^{m+1} w_i^2 e^{-q_i t} + 1 \tag{B.11}$$

Plugging $\sum_{i=1}^{p+1} w_i^2 e^{-q_i t} + 1$ into the reformulated $\tilde{f}$

$$\tilde{\mathfrak{f}} \circ \delta_y(t) = \exp\left\{\frac{1}{2}\left(\sum_{i=1}^{m+1} w_i^2 e^{-q_i t}\right)^\beta\right\} \tag{B.12}$$

This function is proved to be log-convex in equation (B.6). Therefore, condition (5.9) is verified since $\beta > 0$ for MGGD model.
2) For Student T, plugging (5.10) into (5.9),

$$\log\left(\tilde{\mathfrak{f}} \circ \varphi\right)(t) = \frac{\beta + m}{2}\left[1 - \frac{1}{\beta} + \frac{1}{\beta}\exp(\psi(t))\right] \tag{B.13}$$

Therefore, condition (5.9) is verified since $\beta > 0$.

Appendix C

Proofs for Chapter 6

C.1 Proof of proposition 11

This proof is based on Remark 2 of [88]. Denote $U\left(\boldsymbol{\theta}^{(n)}, x\right)$ the direction of descent. According to this remark, if

$$\mathbb{E}\left\langle U\left(\boldsymbol{\theta}^{(n)}, x\right), \nabla_{\boldsymbol{\theta}}^{\odot} D\left(\boldsymbol{\theta}^{(n)}\right)\right\rangle_{\boldsymbol{\theta}^{(n)}}^{\odot} < 0, \text{ for } t > 0 \tag{C.1}$$

almost surely, we have $\lim_{n\to\infty} \boldsymbol{\theta}^{(n)} = \boldsymbol{\theta}^*$. In this situation, for each iteration, the direction of descent could be considered as

$$U\left(\boldsymbol{\theta}^{(0)}, x\right) = -\begin{pmatrix} \nabla_r^* \ell_f\left(\boldsymbol{\theta}^{(0)}; x\right) \\ \left(\mathbf{0}_{\nabla_{\mu_k}^*}\right)_k \\ \left(\mathbf{0}_{\nabla_{\Sigma_k}^*}\right)_k \\ \left(\mathbf{0}_{\nabla_{\beta_k}^*}\right)_k \end{pmatrix} \tag{C.2a}$$

$$U\left(\boldsymbol{\theta}^{(1)}, x\right) = -\begin{pmatrix} \mathbf{0}_{\nabla_r^*} \\ \left(\nabla_{\mu_k}^* \ell_f\left(\boldsymbol{\theta}^{(1)}; x\right)\right)_k \\ \left(\mathbf{0}_{\nabla_{\Sigma_k}^*}\right)_k \\ \left(\mathbf{0}_{\nabla_{\beta_k}^*}\right)_k \end{pmatrix} \tag{C.2b}$$

$$U\left(\boldsymbol{\theta}^{(2)}, x\right) = -\begin{pmatrix} \mathbf{0}_{\nabla_r^*} \\ \left(\mathbf{0}_{\nabla_{\mu_k}^*}\right)_k \\ \left(\nabla_{\Sigma_k}^* \ell_f\left(\boldsymbol{\theta}^{(2)}; x\right)\right)_k \\ \left(\mathbf{0}_{\nabla_{\beta_k}^*}\right)_k \end{pmatrix} \tag{C.2c}$$

$$U\left(\boldsymbol{\theta}^{(3)}, x\right) = -\begin{pmatrix} \mathbf{0}_{\nabla_r} \\ \left(\mathbf{0}_{\nabla_{\mu_k}^*}\right)_k \\ \left(\mathbf{0}_{\nabla_{\Sigma_k}^*}\right)_k \\ \left(\nabla_{\beta_k}^* \ell_f\left(\boldsymbol{\theta}^{(3)}; x\right)\right)_k \end{pmatrix} \tag{C.2d}$$

where

$$\begin{aligned}
\boldsymbol{\theta}^{(0)} &= \left(r^{(0)}, \left(\mu_k^{(0)}\right)_k, \left(\Sigma_k^{(0)}\right)_k, \left(\beta_k^{(0)}\right)_k\right) \\
\boldsymbol{\theta}^{(1)} &= \left(r^{(1)}, \left(\mu_k^{(0)}\right)_k, \left(\Sigma_k^{(0)}\right)_k, \left(\beta_k^{(0)}\right)_k\right) \\
\boldsymbol{\theta}^{(2)} &= \left(r^{(1)}, \left(\mu_k^{(1)}\right)_k, \left(\Sigma_k^{(0)}\right)_k, \left(\beta_k^{(0)}\right)_k\right) \\
\boldsymbol{\theta}^{(3)} &= \left(r^{(1)}, \left(\mu_k^{(1)}\right)_k, \left(\Sigma_k^{(1)}\right)_k, \left(\beta_k^{(0)}\right)_k\right)
\end{aligned} \tag{C.3}$$

Recall that $D(\boldsymbol{\theta}) = -\mathbb{E}\left[\ell_f\left(\boldsymbol{\theta}; x\right)\right]$ and the form of $\nabla_{\boldsymbol{\theta}}^{\odot} D(\boldsymbol{\theta})$ is defined as

$$\nabla_{\boldsymbol{\theta}}^{\odot} D(\boldsymbol{\theta}) = \begin{pmatrix} \nabla_r^* D(\boldsymbol{\theta}) \\ \left(\nabla_{\mu_k}^* D(\boldsymbol{\theta})\right)_k \\ \left(\nabla_{\Sigma_k}^* D(\boldsymbol{\theta})\right)_k \\ \left(\nabla_{\beta_k}^* D(\boldsymbol{\theta})\right)_k \end{pmatrix} \tag{C.4}$$

Therefore, for step 0 to 3, the expectation in (C.1) are

$$\begin{aligned}
\mathbb{E}\left\langle U\left(\boldsymbol{\theta}^{(0)}, x\right), \nabla_{\boldsymbol{\theta}} D\left(\boldsymbol{\theta}^{(0)}\right)\right\rangle_{\boldsymbol{\theta}}^{\odot} &= -\left(\left\|\nabla_r D\left(\boldsymbol{\theta}^{(0)}\right)\right\|_r^*\right)^2 \\
\mathbb{E}\left\langle U\left(\boldsymbol{\theta}^{(1)}, x\right), \nabla_{\boldsymbol{\theta}} D\left(\boldsymbol{\theta}^{(1)}\right)\right\rangle_{\boldsymbol{\theta}}^{\odot} &= -\sum_{k=1}^{K}\left(\left\|\nabla_{\mu_k} D\left(\boldsymbol{\theta}^{(0)}\right)\right\|_{\mu_k}^*\right)^2 \\
\mathbb{E}\left\langle U\left(\boldsymbol{\theta}^{(2)}, x\right), \nabla_{\boldsymbol{\theta}} D\left(\boldsymbol{\theta}^{(2)}\right)\right\rangle_{\boldsymbol{\theta}}^{\odot} &= -\sum_{k=1}^{K}\left(\left\|\nabla_{\Sigma_k} D\left(\boldsymbol{\theta}^{(0)}\right)\right\|_{\Sigma_k}^*\right)^2 \\
\mathbb{E}\left\langle U\left(\boldsymbol{\theta}^{(3)}, x\right), \nabla_{\boldsymbol{\theta}} D\left(\boldsymbol{\theta}^{(3)}\right)\right\rangle_{\boldsymbol{\theta}}^{\odot} &= -\sum_{k=1}^{K}\left(\left\|\nabla_r D\left(\boldsymbol{\theta}^{(0)}\right)\right\|_{\beta_k}^*\right)^2
\end{aligned} \tag{C.5}$$

And so on, for $t > 3$, all these verify condition (C.1). According to remark 2 in [88], in order to obtain the asymptotically linear convergence rate, it is enough to show the mean vector field $X(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\theta}^*}\left[U\left(\boldsymbol{\theta}, x\right)\right]$ has an attractive stationary point at $\boldsymbol{\theta} = \boldsymbol{\theta}^*$. Since $U\left(\boldsymbol{\theta}, x\right) = -\nabla_{\boldsymbol{\theta}} \ell_f(\boldsymbol{\theta}; x)$

$$\mathbb{E}_{\boldsymbol{\theta}^*}\left[U\left(\boldsymbol{\theta}, x\right)\right] = \begin{pmatrix} \nabla_r^* D(\boldsymbol{\theta}) \\ \left(\nabla_{\mu_k}^* D(\boldsymbol{\theta})\right)_k \\ \left(\nabla_{\Sigma_k}^* D(\boldsymbol{\theta})\right)_k \\ \left(\nabla_{\beta_k}^* D(\boldsymbol{\theta})\right)_k \end{pmatrix} \tag{C.6}$$

Under the same assumptions of B.2, the covariant derivative of this vector field at the point $\boldsymbol{\theta} = \boldsymbol{\theta}^*$ is equal to the Hessian $\mathcal{H}(\boldsymbol{\theta}^*)$, which is positive-definite. Therefore the linear convergence rate holds.

C.2 Proof of proposition 12

Note Θ^* is the neighborhood of $\boldsymbol{\theta}^*$ which satisfies all assumptions in proposition 12. The objective function $D(\boldsymbol{\theta})$ is geodesically strongly convex and L-lipschitz smooth in Θ^*. Recall the geodesically L-lipschitz

$$\begin{aligned}
D(\boldsymbol{\theta}^{(n+1)}) \leqslant & D(\boldsymbol{\theta}^{(n)}) - \eta^{(n)}\left\langle \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)}), U(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})\right\rangle_{\boldsymbol{\theta}^{(n)}} \\
& + \frac{L}{2} d^2\left(\boldsymbol{\theta}^{(n)}, \boldsymbol{\theta}^{(n+1)}\right)
\end{aligned} \tag{C.7}$$

Where $\langle\cdot,\cdot\rangle$ denotes the Riemannian metric in the general sense, therefore $\nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)})$ denotes the general Riemannian gradient, and $d(\cdot,\cdot)$ denotes the general Riemannian distance. The vector $U(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})$ denotes the direction of descent, and $\eta^{(n)}$ is the step-size.

The set $\mathcal{X}_{mb}^{(n)}$ is supposed to be a random mini-batch that is uniformly selected from the complete dataset. Therefore

$$\mathbb{E}_{\mathcal{X}_{mb}}\left[U(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})\right] = U\left(\boldsymbol{\theta}^{(n)}, \mathcal{X}\right) \tag{C.8}$$

here $\mathcal{X}$ denotes the complete dataset. Take expectation with respect to $\mathcal{X}_{mb}$

$$\begin{aligned}\mathbb{E}_{\mathcal{X}_{mb}}\left[D(\boldsymbol{\theta}^{(n+1)})\right] \leqslant & D(\boldsymbol{\theta}^{(n)}) - \eta^{(n)}\left\langle \nabla_{\boldsymbol{\theta}}(\boldsymbol{\theta}^{(n)}), U(\boldsymbol{\theta}^{(n)}, \mathcal{X})\right\rangle_{\boldsymbol{\theta}^{(n)}} \\ & + \frac{L}{2}\mathbb{E}_{\mathcal{X}_{mb}}\left[d^2\left(\boldsymbol{\theta}^{(n)}, \boldsymbol{\theta}^{(n+1)}\right)\right]\end{aligned} \tag{C.9}$$

Expand the last term

$$\begin{aligned}d^2(\boldsymbol{\theta}^{(n)}, \boldsymbol{\theta}^{(n+1)}) = & d^2(r^{(n)}, r^{(n+1)}) + \sum_{k=1}^{K}\Big[d^2(\mu_k^{(n)}, \mu_k^{(n+1)}) + d^2(\Sigma_k^{(n)}, \Sigma_k^{(n+1)}) \\ & + d^2(\beta_k^{(n)}, \beta_k^{(n+1)})\Big]\end{aligned} \tag{C.10}$$

Recall the square distance in each sub-space

$$d^2(r^{(n)}, r^{(n+1)}) = \left\|\mathrm{Exp}_{r^{(n)}}^{-1}\left(r^{(n+1)}\right)\right\|^2 = [\eta^{(n)}]^2\left\|U_r(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})\right\|^2 \tag{C.11a}$$

$$d^2(\mu_k^{(n)}, \mu_k^{(n+1)}) = \left\|\mathrm{Exp}_{\mu_k^{(n)}}^{-1}(\mu_k^{(n+1)})\right\|^2 = [\eta^{(n)}]^2\left\|U_{\mu_k}(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})\right\|^2 \tag{C.11b}$$

$$d^2(\Sigma_k^{(n)}, \Sigma_k^{(n+1)}) = \left\|\mathrm{Exp}_{\Sigma_k^{(n)}}^{-1}(\Sigma_k^{(n+1)})\right\|^2 = [\eta^{(n)}]^2\left\|U_{\Sigma_k}\left(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}\right)\right\|^2 \tag{C.11c}$$

$$d^2(\beta_k^{(n)}, \beta_k^{(n+1)}) = \left\|\mathrm{Exp}_{\beta_k^{(n)}}^{-1}\left(\beta_k^{(n+1)}\right)\right\|^2 = [\eta^{(n)}]^2\left\|U_{\beta_k}(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})\right\|^2 \tag{C.11d}$$

therefore

$$\begin{aligned}\mathbb{E}_{\mathcal{X}_{mb}}\left[D\left(\boldsymbol{\theta}^{(n+1)}\right)\right] \leqslant & D(\boldsymbol{\theta}^{(n)}) - \eta^{(n)}\left\langle \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}), U(\boldsymbol{\theta}^{(n)}, \mathcal{X})\right\rangle \\ & + \frac{L[\eta^{(n)}]^2}{2}\mathbb{E}_{\mathcal{X}_{mb}}\left[\left\|U(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})\right\|^2\right]\end{aligned} \tag{C.12}$$

For $r \in \mathcal{S}^{K-1}$, the classic Euclidean gradient coincides with the Riemannian information gradient (3.29).

$$\left\|U_r(\boldsymbol{\theta}^{(n)}, \mathrm{X}_{mb}^{(n)})\right\|^2 = \left\|\nabla_r^* D(\boldsymbol{\theta}^{(n)}, \mathrm{X}_{mb}^{(n)})\right\|^2 = \left\|\nabla_r D(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})\right\|^2 \tag{C.13}$$

For μ_k in $\mathbb{R}^p$,

$$\begin{aligned}\left\|U_{\mu_k}(\boldsymbol{\theta}^{(n)}, \mathrm{X}_{mb}^{(n)})\right\|^2 = & \left\|I_{\mu_k}^{-1}\Sigma_k^{(n)}\nabla_{\mu_k} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})\right\|^2 \\ \leqslant & \frac{[\lambda_{max,k}^{(n)}]^2}{I_{\mu_k}^2}\left\|\nabla_{\mu_k} D\left(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}\right)\right\|^2\end{aligned} \tag{C.14}$$

where $\lambda_{max,k}^{(n)}$ is the largest eigenvalue of $\Sigma_k^{(n)}$. Then for Σ_k, its information gradient is

$$\begin{aligned} U_{\Sigma_k}(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}) =& J_{\Sigma_k,1} \left[\nabla_{\Sigma_k}^{\uparrow} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})\right]^{\perp} + J_{\Sigma_k,2} \left[\nabla_{\Sigma_k}^{\uparrow} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})\right]^{\parallel} \\ =& J_{\Sigma_k,1} \nabla_{\Sigma_k}^{\uparrow} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}) \\ &+ \frac{J_{\Sigma_k,2} - J_{\Sigma_k,1}}{m} \operatorname{tr}\left\{[\Sigma_k^{(n)}]^{-1} \nabla_{\Sigma_k}^{\uparrow} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})\right\} \Sigma_k^{(n)} \end{aligned} \tag{C.15}$$

Using the affine invariant metric, the square norm of $U_{\Sigma_k}(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)})$ is

$$\left\| U_{\Sigma_k}(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}) \right\|_{\uparrow}^2 = \operatorname{tr}\left\{ \left[(\Sigma_k^{(n)})^{-1} U_{\Sigma_k}(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}) \right]^2 \right\} \tag{C.16}$$

After some necessary simplifications, the following relation ship could be summarized

$$\left\| U_{\Sigma_k}\left(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}\right) \right\|_{\uparrow}^2 \leqslant J_{\Sigma_k,1}^2 \left\| \nabla_{\Sigma_k}^{\uparrow} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}) \right\|_{\uparrow}^2 \tag{C.17}$$

Finally, for β_k, it is easy to obtain

$$\left\| U_{\beta_k}\left(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}\right) \right\|^2 = \left| I_{\beta_k}^{-1} \nabla_{\beta_k} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}) \right|^2 = I_{\beta_k}^{-2} \left\| \nabla_{\beta_k} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}) \right\|^2 \tag{C.18}$$

We have then

$$\begin{aligned} \mathbb{E}_{\mathcal{X}_{mb}}\left[D\left(\boldsymbol{\theta}^{(n+1)}\right)\right] \leqslant& D\left(\boldsymbol{\theta}^{(n)}\right) - \eta^{(n)} \left\langle \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}), U(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\rangle \\ &+ \frac{L\left[\eta^{(n)}\right]^2 \tau_{max}^{(n)}}{2} \mathbb{E}_{\mathcal{X}_{mb}}\left[\left\| \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}) \right\|^2 \right] \end{aligned} \tag{C.19}$$

Where $\tau_{max}^{(n)} = \max\left\{1, \left(\frac{\lambda_{max,k}^{(n)}}{I_{\mu_k}}\right)_k^2, \left(J_{\Sigma_k,1}^2\right)_k, \left(I_{\beta_k}^{-2}\right)_k\right\}$. Recall the definition of ρ-strong growth condition

$$\mathbb{E}_{\mathcal{X}_{mb}} \left\| \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}, \mathcal{X}_{mb}) \right\|^2 \leqslant \rho \left\| \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}, \mathcal{X}) \right\|^2 \tag{C.20}$$

Then, using Lemma 1 of [54], we could obtain

$$\mathbb{E}_{\mathcal{X}_{mb}}\left[\left\| \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}_{mb}^{(n)}) \right\|^2 \right] \leqslant \rho' \left\| \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\|^2 \tag{C.21}$$

where $\rho' = \frac{(n-b)(\rho-1)}{(n-1)b} + 1$, and b is the size of mini-batch, n is the size of dataset. The expected decrease becomes

$$\begin{aligned} \mathbb{E}_{\mathcal{X}_{mb}}\left[D(\boldsymbol{\theta}^{(n+1)})\right] \leqslant& D(\boldsymbol{\theta}^{(n)}) - \eta^{(n+1)} \left\langle \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}), U(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\rangle \\ &+ \frac{L\left[\eta^{(n+1)}\right]^2 \tau_{max}^{(n)} \rho'}{2} \left\| \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\|^2 \end{aligned} \tag{C.22}$$

Then, for item $\left\langle \nabla_{\boldsymbol{\theta}} D(\theta^{(n)}, \mathcal{X}), U(\theta^{(n)}, \mathcal{X}) \right\rangle$, starting with r

$$\left\langle \nabla_r D(\boldsymbol{\theta}^{(n)}, \mathcal{X}), U_r(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\rangle = \left\| \nabla_r D(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\|^2 \tag{C.23}$$

For μ_k

$$\left\langle \nabla_{\mu_k} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}), U_{\mu_k}(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\rangle \geqslant \frac{\lambda_{min,k}}{I_{\mu_k}} \left\| \nabla_{\mu_k} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\|^2 \tag{C.24}$$

where $\lambda_{min,k}$ is the smallest eigenvalue of $\Sigma_k^{(n)}$. For scatter matrix Σ_k

$$\left\langle \nabla_{\Sigma_k} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}), U_{\Sigma_k}(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\rangle \geqslant J_{\Sigma_k,2} \left\| \nabla_{\Sigma_k} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\|_{\uparrow}^2 \tag{C.25}$$

Finally, the shape parameter β_k

$$\left\langle \nabla_{\beta_k} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}), U_{\beta_k}(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\rangle = I_{\beta_k}^{-1} \left\| \nabla_{\beta_k} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\|^2 \tag{C.26}$$

Therefore

$$\begin{aligned} \mathbb{E}_{\mathcal{X}_{mb}} \left[D(\boldsymbol{\theta}^{(n+1)}) \right] \leqslant & D(\boldsymbol{\theta}^{(n)}) - \eta^{(n+1)} \tau_{min}^{(n)} \left\| \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\|^2 \\ & + \frac{L \left[\eta^{(n+1)} \right]^2 \tau_{max}^{(n)} \rho'}{2} \left\| \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\|^2 \end{aligned} \tag{C.27}$$

where $\tau_{min}^{(n)} = \min \left\{ 1, \left(\frac{\lambda_{min,k}}{I_{\mu_k}} \right)_k, \left(J_{\Sigma_k,2} \right)_k, \left(I_{\beta_k}^{-1} \right)_k \right\}$. Since $\eta^{(n+1)} = \frac{\tau_{min}^{(n)}}{\rho' L \tau_{max}^{(n)}}$, we have

$$\mathbb{E}_{\mathcal{X}_{mb}} \left[D(\boldsymbol{\theta}^{(n+1)}) \right] \leqslant D(\boldsymbol{\theta}^{(n)}) - \frac{\left[\tau_{min}^{(n)} \right]^2}{2 \rho' L \tau_{max}^{(n)}} \left\| \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\|^2 \tag{C.28}$$

Using the Polyak-Lojasiewicz inequality in Riemannian context (which is proved in section C.3) $\left\| \nabla_{\boldsymbol{\theta}} D(\boldsymbol{\theta}^{(n)}, \mathcal{X}) \right\| \geqslant 2\alpha \left(D(\boldsymbol{\theta}^{(n)}) - D(\boldsymbol{\theta}^*) \right)$, we have

$$\mathbb{E}_{\mathcal{X}_{mb}} \left[D(\boldsymbol{\theta}^{(n+1)}) \right] \leqslant D(\boldsymbol{\theta}^{(n)}) - \frac{\left[\tau_{min}^{(n)} \right]^2 \alpha}{\rho' L \tau_{max}^{(n)}} \left(D(\boldsymbol{\theta}^{(n)}) - D(\boldsymbol{\theta}^*) \right) \tag{C.29}$$

substracting $D(\boldsymbol{\theta}^*)$ from both sides

$$\mathbb{E}_{\mathcal{X}_{mb}} \left[D(\boldsymbol{\theta}^{(n+1)}) \right] - D(\boldsymbol{\theta}^*) \leqslant \left(1 - \frac{\left[\tau_{min}^{(n)} \right]^2 \alpha}{\rho' L \tau_{max}^{(n)}} \right) \left[D(\boldsymbol{\theta}^{(n)}) - D(\boldsymbol{\theta}^*) \right] \tag{C.30}$$

Range these N times of iteration

$$\mathbb{E}_{\mathcal{X}_{mb}} \left[D(\boldsymbol{\theta}^{(N)}) \right] - D(\boldsymbol{\theta}^*) \leqslant \left\{ \prod_{n=0}^{N} \left(1 - \frac{\alpha \left[\tau_{min}^{(n)} \right]^2}{\rho' L \tau_{max}^{(n)}} \right) \right\} \left[D(\boldsymbol{\theta}^{(0)}) - D(\boldsymbol{\theta}^*) \right] \tag{C.31}$$

C.3 Proof of Polyak-Lojasiewicz inequality in Riemannian context

Consider a continuous and differentiable function defined on a Riemannian manifold $f : \mathcal{M} \to \mathbb{R}$. Note $x^* = \arg\inf_{x \in \mathcal{M}} f(x)$ its minimum, there exists a neighborhood $\mathcal{M}^* \subset \mathcal{M}$ of x^* such that the function f is strongly geodesically convex in $\mathcal{M}^*$. For any $x \in \mathcal{M}^*$, $\exists \alpha > 0$ such that

$$f(x^*) - f(x) \geqslant \left\langle \mathrm{Exp}_x^{-1}(x^*), \nabla_x f(x) \right\rangle_x + \frac{\alpha}{2} \mathrm{d}^2(x, x^*) \tag{C.32}$$

Take the negative on both sides of this inequality to get

$$\begin{aligned} f(x) - f(x^*) \leqslant & - \left\langle \mathrm{Exp}_x^{-1}(x^*), \nabla_x f(x) \right\rangle_x - \frac{\alpha}{2} \left\| \mathrm{Exp}_x^{-1}(x^*) \right\|_x^2 \\ = & - \frac{\alpha}{2} \left\| \mathrm{Exp}_x^{-1}(x^*) + \frac{1}{\alpha} \nabla_x f(x) \right\|_x^2 + \frac{1}{2\alpha} \left\| \nabla_x f(x) \right\|_x^2 \\ \leqslant & \frac{1}{2\alpha} \left\| \nabla_x f(x) \right\|_x^2 \end{aligned} \tag{C.33}$$

where there is the result.

Bibliography

[1] R. Abraham, J.E. Marsden, and T. Ratiu. Manifolds, tensor analysis, and applications. Springer, 1988.

[2] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. Optimization algorithms on matrix manifolds. Princeton University Press, 2009.

[3] S. Amari and H. Nagaoka. Methods of Information Geometry. American Mathematical Society, 2000.

[4] Shun-ichi Amari. Information geometry and its applications, volume 194. Springer, 2016.

[5] Jeffrey L Andrews and Paul D McNicholas. Model-based clustering, classification, and discriminant analysis via mixtures of multivariate t-distributions. Statistics and Computing, 22(5):1021–1029, 2012.

[6] Yakoub Bazi, Lorenzo Bruzzone, and Farid Melgani. Image thresholding based on the EM algorithm and the generalized gaussian distribution. Pattern Recognition, 40(2):619–634, 2007.

[7] Albert Benveniste, Michel Métivier, and Pierre Priouret. Adaptive algorithms and stochastic approximations, volume 22. Springer Science & Business Media, 2012.

[8] Maia Berkane, Kevin Oden, and Peter M Bentler. Geodesic estimation in elliptical distributions. Journal of Multivariate Analysis, 63(1):35–46, 1997.

[9] Olivier Besson and Yuri I Abramovich. On the Fisher information matrix for multivariate elliptically contoured distributions. IEEE Signal Processing Letters, 20(11):1130–1133, 2013.

[10] L. Bombrun, G. Vasile, M. Gay, and F. Totir. Hierarchical segmentation of polarimetric SAR images using heterogeneous clutter models. IEEE Transactions on Geoscience and Remote Sensing, 49(2):726–737, 2011.

[11] Silvere Bonnabel. Stochastic gradient descent on Riemannian manifolds. IEEE Transactions on Automatic Control, 58(9):2217–2229, 2013.

[12] Larbi Boubchir and Jalal M Fadili. Multivariate statistical modeling of images with the curvelet transform. In Proceedings of the Eighth International Symposium on Signal Processing and Its Applications, volume 2, pages 747–750. IEEE, 2005.

[13] Florent Bouchard, Ammar Mian, Jialun Zhou, Salem Said, Guillaume Ginolhac, and Yannick Berthoumieu. Riemannian geometry for compound Gaussian distributions: Application to recursive change detection. Signal Processing, 176:107716, 2020.

[14] Zois Boukouvalas, Salem Said, Lionel Bombrun, Yannick Berthoumieu, and Tülay Adalı. A new Riemannian averaged fixed-point algorithm for MGGD parameter estimation. IEEE Signal Processing Letters, 22(12):2314–2318, 2015.

[15] Michel Broniatowski. Minimum divergence estimators, maximum likelihood and exponential families. Statistics & Probability Letters, 93:27–33, 2014.

[16] Michel Broniatowski and Amor Keziou. Parametric estimation and tests through divergences and the duality technique. Journal of Multivariate Analysis, 100(1):16–36, 2009.

[17] Stamatis Cambanis, Steel Huang, and Gordon Simons. On the theory of elliptically contoured distributions. Journal of Multivariate Analysis, 11(3):368–385, 1981.

[18] George Casella and Roger L Berger. Statistical inference. Cengage Learning, 2021.

[19] Isaac Chavel. Riemannian geometry: a modern introduction, volume 98. Cambridge university press, 2006.

[20] Q. Chen, H. Yang, L. Li, and X. Liu. A novel statistical texture feature for SAR building damage assessment in different polarization modes. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 13:154–165, 2020.

[21] Dongwook Cho, Tien D Bui, and Guangyi Chen. Image denoising based on wavelet shrinkage using neighbor and level dependency. International journal of wavelets, multiresolution and information processing, 7(03):299–311, 2009.

[22] R. Courant and F. John. Introduction to Calculus and Analysis, volume 1. Interscience Publishers, 1965.

[23] Arthur P Dempster, Nan M Laird, and Donald 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.

[24] Mukund N Desai and Rami S Mangoubi. Robust Gaussian and non-Gaussian matched subspace detection. IEEE Transactions on Signal Processing, 51(12):3115–3127, 2003.

[25] Marie Duflo. Random iterative models, volume 34. Springer Science & Business Media, 2013.

[26] Tarek Elguebaly and Nizar Bouguila. Bayesian learning of generalized Gaussian mixture models on biomedical images. In IAPR Workshop on Artificial Neural Networks in Pattern Recognition, pages 207–218. Springer, 2010.

[27] K. Fang and Y.T. Zhang. Generalized multivariate analysis. Science Press, 1990.

[28] Kai Wang Fang. Symmetric multivariate and related distributions. CRC Press, 2018.

[29] J. I. Fernández-Michelli, M. Hurtado, J. A. Areta, and C. H. Muravchik. Unsupervised polarimetric SAR image classification using $\mathcal{G}_p^0$ mixture model. IEEE Geoscience and Remote Sensing Letters, 14(5):754–758, 2017.

[30] Mario A. T. Figueiredo and Anil K. Jain. Unsupervised learning of finite mixture models. IEEE Transactions on pattern analysis and machine intelligence, 24(3):381–396, 2002.

[31] Gabriel Frahm, Markus Junker, and Alexander Szimayer. Elliptical copulas: applicability and limitations. Statistics & Probability Letters, 63(3):275–286, 2003.

[32] Joana Frontera-Pons, Mélanie Mahot, Jean Philippe Ovarlez, Frédéric Pascal, Sze Kim Pang, and Jocelyn Chanussot. A class of robust estimates for detection in hyperspectral images using elliptical distributions background. In 2012 IEEE International Geoscience and Remote Sensing Symposium, pages 4166–4169. IEEE, 2012.

[33] Jeffrey C Fuhrer, George R Moore, and Scott D Schuh. Estimating the linear-quadratic inventory model maximum likelihood versus generalized method of moments. Journal of Monetary Economics, 35(1):115–157, 1995.

[34] E Gómez, MA Gomez-Viilegas, and JM Marin. A multivariate generalization of the power exponential family of distributions. Communications in Statistics-Theory and Methods, 27(3):589–600, 1998.

[35] Praful Gupta, Anush Krishna Moorthy, Rajiv Soundararajan, and Alan Conrad Bovik. Generalized Gaussian scale mixtures: A model for wavelet coefficients of natural images. Signal Processing: Image Communication, 66:87 – 94, 2018.

[36] Peter Hall and Christopher C Heyde. Martingale limit theory and its application. Academic press, 2014.

[37] Marzieh Hasannasab, Johannes Hertrich, Friederike Laus, and Gabriele Steidl. Alternatives to the EM algorithm for ML estimation of location, scatter matrix, and degree of freedom of the Student t distribution. Numerical Algorithms, pages 1–42, 2020.

[38] Johannes Hertrich and Gabriele Steidl. Inertial stochastic PALM and its application for learning student-t mixture models. arXiv preprint arXiv:2005.02204, 2020.

[39] Christopher C Heyde. Quasi-likelihood and its application: a general approach to optimal parameter estimation. Springer Science & Business Media, 2008.

[40] Nicholas J Higham. Functions of matrices: theory and computation. SIAM, 2008.

[41] Hajo Holzmann, Axel Munk, and Tilmann Gneiting. Identifiability of finite mixtures of elliptical distributions. Scandinavian journal of statistics, 33(4):753–763, 2006.

[42] Reshad Hosseini and Suvrit Sra. An alternative to EM for gaussian mixture models: batch and stochastic Riemannian optimization. Mathematical Programming, 181(1):187–223, 2020.

[43] Hristina Hristova, Olivier Le Meur, Remi Cozot, and Kadi Bouatouch. Transformation of the multivariate generalized Gaussian distribution for image editing. IEEE transactions on visualization and computer graphics, 24(10):2813–2826, 2017.

[44] I. A. Ibragimov and R. Z. Khasminskii. Statistical estimation–asymptotic theory. Springer-Verlag, 1981.

[45] Thomas Kailath. Linear systems, volume 156. Prentice-Hall Englewood Cliffs, NJ, 1980.

[46] Douglas Kelker. Distribution theory of spherical distributions and a location-scale parameter generalization. Sankhyā: The Indian Journal of Statistics, Series A, pages 419–430, 1970.

[47] Samuel Kotz and Saralees Nadarajah. Multivariate t-distributions and their applications. Cambridge University Press, 2004.

[48] Harold Kushner and G George Yin. Stochastic approximation and recursive algorithms and applications, volume 35. Springer Science & Business Media, 2003.

[49] Roland Kwitt, Peter Meerwald, Andreas Uhl, and Geert Verdoolaege. Testing a multivariate model for wavelet coefficients. In 2011 18th IEEE International Conference on Image Processing, pages 1277–1280. IEEE, 2011.

[50] Friederike Laus and Gabriele Steidl. Multivariate myriad filters based on parameter estimation of student-t distributions. SIAM Journal on Imaging Sciences, 12(4):1864–1904, 2019.

[51] Shengxi Li, Zeyang Yu, and Danilo Mandic. A universal framework for learning the elliptical mixture model. IEEE Transactions on Neural Networks and Learning Systems, 2020.

[52] Tsung-I Lin, Paul D McNicholas, and Hsiu J Ho. Capturing patterns via parsimonious t mixture models. Statistics & Probability Letters, 88:80–87, 2014.

[53] Gaétan Marceau-Caron and Yann Ollivier. Practical Riemannian neural networks. arXiv preprint arXiv:1602.08007, 2016.

[54] Si Yi Meng, Sharan Vaswani, Issam Hadj Laradji, Mark Schmidt, and Simon Lacoste-Julien. Fast and furious convergence: Stochastic second order methods under interpolation. In International Conference on Artificial Intelligence and Statistics, pages 1375–1386. PMLR, 2020.

[55] MIT. Vision texture database. `http://vismod.media.mit.edu/pub/VisTex`.

[56] Julien Munier. Steepest descent method on a Riemannian manifold: the convex case. Balkan Journal of Geometry & Its Applications, 12(2), 2007.

[57] Fatma Najar, Sami Bourouis, Rula Al-Azawi, and Ali Al-Badi. Online recognition via a finite mixture of multivariate generalized Gaussian distributions. In Mixture Models and Applications, pages 81–106. Springer, 2020.

[58] M. B. Nevelson and R. Z. Hasminskii. Stochastic approximation and recursive estimation. American Mathematical Society,, 1973.

[59] Frédéric Pascal, Lionel Bombrun, Jean-Yves Tourneret, and Yannick Berthoumieu. Parameter estimation for multivariate generalized Gaussian distributions. IEEE Transactions on Signal Processing, 61(23):5960–5971, 2013.

[60] David Peel and Geoffrey J McLachlan. Robust mixture modelling using the t distribution. Statistics and computing, 10(4):339–348, 2000.

[61] Xavier Pennec, Pierre Fillard, and Nicholas Ayache. A Riemannian framework for tensor computing. International Journal of computer vision, 66(1):41–66, 2006.

[62] Peter Petersen, S Axler, and KA Ribet. Riemannian geometry, volume 171. Springer, 2006.

[63] François Pitié and Anil Kokaram. The linear monge-kantorovitch linear colour mapping for example-based colour transfer. In 4th European conference on visual media production, pages 1–9. IET, 2007.

[64] C Radhakrishna Rao. Information and the accuracy attainable in the estimation of statistical parameters. Reson. J. Sci. Educ, 20:78–90, 1945.

[65] Meisam Razaviyayn, Mingyi Hong, and Zhi-Quan Luo. A unified convergence analysis of block successive minimization methods for nonsmooth optimization. SIAM Journal on Optimization, 23(2):1126–1153, 2013.

[66] Violeta Roizman, Gordana Draskovic, and Frédéric Pascal. A new clustering algorithm for PolSAR images segmentation. In 2019 IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), pages 684–688. IEEE, 2019.

[67] Violeta Roizman, Matthieu Jonckheere, and Frédéric Pascal. A flexible EM-like clustering algorithm for noisy data. arXiv preprint arXiv:1907.01660, 2019.

[68] David Saad. On-line learning in neural networks. Number 17. Cambridge University Press, 2009.

[69] Eusebio Gómez Sánchez-Manzano, Miguel Angel Gomez-Villegas, and Juan-Miguel Marín-Diazaraque. A matrix variate generalization of the power exponential family of distributions. Communications in Statistics-Theory and Methods, 31(12):2167–2182, 2002.

[70] Jacob Scharcanski. A wavelet-based approach for analyzing industrial stochastic textures with applications. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 37(1):10–22, 2006.

[71] Jonathan R Senning. Computing and estimating the rate of convergence, 2007.

[72] Albert N. Shiryaev. Probability. Springer-Verlag New York, 1996.

[73] Lene Theil Skovgaard. A riemannian geometry of the multivariate normal model. Scandinavian journal of statistics, pages 211–223, 1984.

[74] Suvrit Sra and Reshad Hosseini. Geometric optimisation on positive definite matrices for elliptically contoured distributions. In Advances in Neural Information Processing Systems, pages 2562–2570, 2013.

[75] Suvrit Sra and Reshad Hosseini. Conic geometric optimization on the manifold of positive definite matrices. SIAM Journal on Optimization, 25(1):713–739, 2015.

[76] Shan Tan and Licheng Jiao. Multivariate statistical models for image denoising in the wavelet domain. International Journal of Computer Vision, 75(2):209–230, 2007.

[77] Nilesh Tripuraneni, Nicolas Flammarion, Francis Bach, and Michael I Jordan. Averaging stochastic gradient descent on Riemannian manifolds. In Conference On Learning Theory, pages 650–687. PMLR, 2018.

[78] Aad W Van der Vaart. Asymptotic statistics, volume 3. Cambridge university press, 2000.

[79] Sharan Vaswani, Francis Bach, and Mark Schmidt. Fast and faster convergence of SGD for over-parameterized models and an accelerated perceptron. In The 22nd International Conference on Artificial Intelligence and Statistics, pages 1195–1204. PMLR, 2019.

[80] Sharan Vaswani, Aaron Mishkin, Issam Laradji, Mark Schmidt, Gauthier Gidel, and Simon Lacoste-Julien. Painless stochastic gradient: Interpolation, line-search, and convergence rates. In Advances in Neural Information Processing Systems, pages 3732–3745, 2019.

[81] Geert Verdoolaege, Steve De Backer, and Paul Scheunders. Multiscale colour texture retrieval using the geodesic distance between multivariate generalized gaussian models. In 2008 15th IEEE International Conference on Image Processing, pages 169–172. IEEE, 2008.

[82] Geert Verdoolaege and Paul Scheunders. Geodesics on the manifold of multivariate generalized Gaussian distributions with an application to multicomponent texture discrimination. International Journal of Computer Vision, 95(3):265, 2011.

[83] Geert Verdoolaege and Paul Scheunders. On the geometry of multivariate generalized Gaussian models. Journal of mathematical imaging and vision, 43(3):180–193, 2012.

[84] Bin Wang, Huanyu Zhang, Ziping Zhao, and Ying Sun. Globally convergent algorithms for learning multivariate generalized Ggaussian distributions.

[85] Ami Wiesel. Geodesic convexity and covariance estimation. IEEE transactions on signal processing, 60(12):6182–6189, 2012.

[86] Hongyi Zhang and Suvrit Sra. First-order methods for geodesically convex optimization. In Conference on Learning Theory, pages 1617–1638. PMLR, 2016.

[87] Teng Zhang, Ami Wiesel, and Maria Sabrina Greco. Multivariate generalized Gaussian distribution: Convexity and graphical models. IEEE Transactions on Signal Processing, 61(16):4141–4148, 2013.

[88] Jialun Zhou and Salem Said. Fast, asymptotically efficient, recursive estimation in a Riemannian manifold. Entropy, 21(10):1021, 2019.

[89] Jialun Zhou, Salem Said, and Yannick Berthoumieu. Online estimation of mixtures of ECD: Component-wise information gradient method.

[90] Jialun Zhou, Salem Said, and Yannick Berthoumieu. Riemannian information gradient methods for the parameter estimation of ECD: Some applications in image processing. arXiv preprint arXiv:2011.02806, 2020.