Joint analysis of eye movements and EEGs using coupled hidden Markov

Brice Olivier

Communauté
UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTÉ UNIVERSITÉ GRENOBLE ALPES

Spécialité : Mathématiques Appliquées

Arrêté ministériel : 25 mai 2016

Présentée par

Brice OLIVIER

Thèse dirigée par **Jean-Baptiste DURAND**, MCF, ENSIMAG
et codirigée par **Anne GUERIN-DUGUE**, Professeur , UGA

préparée au sein du **Laboratoire Laboratoire Jean Kuntzmann** dans **l'École Doctorale Mathématiques, Sciences et technologies de l'information, Informatique**

Analyse conjointe de traces oculométriques et d'EEG à l'aide de modèles de Markov cachés couplés

Joint analysis of eye movements and EEGs using coupled hidden Markov

Thèse soutenue publiquement le **26 juin 2019**,
devant le jury composé de :

Monsieur JEAN-BAPTISTE DURAND
MAITRE DE CONFERENCES, GRENOBLE INP, Directeur de thèse
Madame JARODZKA HALSZKA
PROFESSEUR ASSOCIE, UNIVERSITE OUVERTE A HEERLEN - PAYS-BAS, Rapporteur
Monsieur ALFONS JUAN
PROFESSEUR, UNIV. POLITECHNIQUE DE VALENCE - ESPAGNE, Rapporteur
Monsieur JEAN-MICHEL BOUCHEIX
PROFESSEUR, UNIVERSITE DE BOURGOGNE, Examinateur
Madame SOPHIE ACHARD
DIRECTRICE DE RECHERCHE, CNRS DELEGATION ALPES, Président
Madame ANNE GUERIN-DUGUE
PROFESSEUR, UNIVERSITE GRENOBLE ALPES, Co-directeur de thèse

A Poornima.

Acknowledgements

Je tiens à remercier en premier lieu mes directeurs de thèse, Jean-Baptiste Durand et Anne Guérin-Dugué, pour m'avoir donné l'opportunité de collaborer avec eux sur ce sujet si complexe à la jointure entre statistiques et sciences cognitives. Vous avez toujours su trouver le bon compromis entre conseils éclairés, pour me recentrer quand je divaguais du sujet, et autonomie, pour me laisser m'épanouir. Grâce à cela, j'ai beaucoup appris de vous tout en prenant du plaisir à travailler. Merci aussi à Marianne Clausel pour m'avoir aiguillé durant la première année de ma thèse et pour avoir été si dynamique et force de proposition.

Je remercie également mes rapporteurs, Halszka Jarodzka et Alfons Juan ainsi que les autres membres du jury, Jean-Michel Boucheix et Sophie Achard, pour s'être intéressés et d'avoir pris le temps de lire mon travail mais aussi pour les retours et discussions plus que pertinentes, qui m'ont apporté une nouvelle vision sur mon propre travail. Sophie, je suis particulièrement reconnaissant pour ton expertise et les discussions fructueuses que nous avons pu avoir et qui m'ont permis de compléter mes travaux.

Ils ne le savent sûrement pas, mais j'éprouve également une profonde gratitude et admiration pour mes enseignants de Master, plus particulièrement Jairo Cugliari et Julien Ah-Pine, pour m'avoir transmis leur démence pour la recherche et pour avoir été si patients et pédagogues pendant mais aussi en dehors des cours.

Merci aux membres du laboratoire Jean Kuntzmann pour ces très précieux derniers mois de ma thèse que nous avons passés ensemble. Chloé, Fairouz, Modibo, Giulio, Philomène, Anna, Guillaume, Julien, Vincent et les autres, vos sourires, votre bonne humeur et votre positivité vont me manquer.

Merci à la grande famille Mistis à Inria, ceux avec qui j'ai passé la plupart de mon temps ces quatre dernières années. J'ai eu la chance de voir s'en aller les anciens, Pablo, Aina, Nourou, Jaime, Gildas, Emeline, Priscillia, Marta, Hongliang, Thomas, Pascal, Jean-Michel, Alessandro, de me développer avec ceux de la génération dorée, Thibaud, Clément, Alexis, de voir grandir les petits, Julyan, Fatima, Meryem, Karina, Michal, Aleksandra, Artem, Antoine, Pascal, Fei, Veronica, Fabien, Mariia, Benoît, Alexandre, Yaroslav, Théo, Nicolas, Mariem, Yassine, Caroline, Daria, Virgilio, Fatoumata, Valentin, mais aussi de côtoyer ceux qui sont aussi anciens que les murs du bâtiment Florence, Stéphane, Jean-Baptiste.

A toutes les belles rencontres que j'ai pu faire à Grenoble, en Inde, ou ailleurs et qui, pour certaines, ont changées ma vie, merci. Je pense notamment à Riddhima, Davinder, Anju, Elisabeth, Cécile, Arielle, Valeria, Keerthi, Meryem, Thomas, Jordi et Pierre. Enfin, frer, je ne serai jamais assez reconnaissant et admiratif pour tout ce qu'on a traversé et à quel point on a évolué ensemble. Ta présence envahissante au quotidien va me manquer.

A tous mes vieux amis qui sont comme ma famille, Vincent, Thomas, Fabien, Corentin, Amélie, et leur compagnon de vie respectif, Stéphanie, Betty, Domitille, Anthony, merci du fond du coeur ! Nous avons grandi et partagé tellement de choses ensemble... Sans vous je n'en serai jamais arrivé là.

Je ne trouverai jamais les mots pour remercier mes parents pour tout ce qu'ils ont fait pour moi. Ils m'ont toujours laissé m'éveiller en me laissant faire mes propres choix, mes propres erreurs, en me secourant quand j'en avais besoin et en me fournissant ce merveilleux cocon familial.

Enfin, ma Poornima, je te dois tout. Merci du fond du coeur.

Abstract

This PhD thesis consists in jointly analyzing eye-tracking signals and multi-channel electroencephalograms (EEGs) acquired concomitantly on participants doing an information collection reading task in order to take a binary decision - is the text related to some topic or not ? Textual information search is not a homogeneous process in time - neither on a cognitive point of view, nor in terms of eye-movement. On the contrary, this process involves several steps or phases, such as normal reading, scanning, careful reading - in terms of oculometry - and creation and rejection of hypotheses, confirmation and decision - in cognitive terms.

In a first contribution, we discuss an analysis method based on hidden semi-Markov chains on the eye-tracking signals in order to highlight four interpretable phases in terms of information acquisition strategy: normal reading, fast reading, careful reading, and decision making.

In a second contribution, we link these phases with characteristic changes of both EEGs signals and textual information. By using a wavelet representation of EEGs, this analysis reveals variance and correlation changes of the inter-channels coefficients, according to the phases and the bandwidth. And by using word embedding methods, we link the evolution of semantic similarity to the topic throughout the text with strategy changes.

In a third contribution, we present a new model where EEGs are directly integrated as output variables in order to reduce the state uncertainty. This novel approach also takes into consideration the asynchronous and heterogeneous aspects of the data.

Contents

List of Figures

List of Tables

List of Algorithms

Nomenclature

Other notations

$B = (G, \theta)$	Bayesian Network with graph G and parameters θ
$G = (V, E)$	Graph with vertices V and edges E
$\mathbb{1}\{\}$	Indicator function
$pa(.)$	Parent function of a node in the graph

Random variables and probability distributions

$\mathbb{E}[X]$	Expectation of random variable X
$\mathscr{L}_D(\theta)$	Log-likelihood of the parameter θ w.r.t. dataset D
$\mathcal{L}_D(\theta)$	Likelihood of the parameter θ w.r.t. dataset D
p_θ	Discrete probability distribution parameterized by θ
p_X	Probability distribution of discrete random variable X
f_θ	Continuous probability distribution parameterized by θ
F_X	Cumulative distribution function of random variable X
f_X	Probability density function of continuous variable X
P_θ	Probability distribution parameterized by θ
P_X	Probability distribution of X
X	Random variable X
x	Realization of random variable X

$\mathbf{X}$	Set of random variables $(X_1,...,X_N)$
D	Dataset $\{x_n\}_{n\in[\![1,N]\!]}$
$\mathbf{x}$	Set $(x_1,...,x_N)$ of realizations of random variables $(X_1,...,X_N)$
$\mathbf{X}$, $X_{1:T}$, $\{X_t\}_{t=1^T}$	Stochastic process $(X_1,...,X_T)$ indexed by t
$\mathbb{V}[X]$	Variance of random variable X

Sets

$\times$	Cartesian product symbol
$\setminus$	Set difference symbol
$\in$	’in’ symbol
$\mathbb{N}$	Set of non-negative integers
$[\![1,n]\!]$	Set of integers from 1 to n
$\mathbb{R}$	Set of real numbers
$\mathscr{X}$	Set of values on random variable X
$\mathbb{Z}$	Set of integers

Acronyms / Abbreviations

AHHSMM	Asynchronous Heterogeneous Hidden Semi-Markov Model
BN	Bayesian Network
CPD	Conditional Probability Distribution
DBN	Dynamic Bayesian Network
EDHMM	Explicit Duration Hidden Markov Model
EEG	Electroencephalogram
EM	Expectation Maximization
ESS	Expected Sufficient Statistic

HMM	Hidden Markov Model
HR	Highly related text to the topic
HSMM	Hidden Semi-Markov Model
i.i.d.	Independent and Identically Distributed
IS	Information Search
JPD	Joint Probability Distribution
KI	Knowledge Injection
MR	Moderately related text to the topic
NR	Normal Reading
SB	Sequence Breaking
SC	Slow Confirmation
SR	Speed Reading
UR	Unrelated text to the topic
s.a.	such as
s.t.	such that
w.r.t.	with respect to

Introduction

PhD context

This PhD takes place within the French national research agency funded project PERSYVAL-Lab, project-team OculoNimbus.

In vision, the human interacts with its environment by performing dynamic exploration of visual regions of interest through eye movements. Understanding mechanisms responsible for this efficient information sampling opens multipurpose perspectives for innovation in human-computer interaction, either by imitating human visual exploration in robots or by creating truly user-friendly experiences for humans.

The goal of the OculoNimbus project is to provide statistical models that notably: segment spatiotemporal into cognitive strategies, analyze dependencies with respect to individual differences.

Topic

Eye movements hold information. Invented in the late 40's by Hartridge and Thompson, the eye-tracker has opened a breach for researchers to analyze the way we, humans, read and process information. As a matter of fact, empirical studies have shown that eye movement itself holds information about the reading process. For example, longer eye fixations have been observed on misspelled, less common words or incongruent words regarding the topic (Rayner, 1998). However, reading studies mainly focused on the microprocesses of reading. Experimentally-driven models have been proposed to simulate human reading behavior by modelling its microprocesses (Reichle et al., 2012; Engbert et al., 2005). At macroscopic scales and based on empirical studies, Carver (1990) identified that readers leverage distinct processes to better accomplish their goals. He characterized these processes as reading strategies and discovered five of them, which could be clustered according to the reading rate. Finally, Simola et al. (2008) proposed a data-driven method to highlight variations of eye movement patterns within a same information search task. In the same context, we raise the following question: how to rigorously and robustly segment a sequence of eye movements into interpretable phases in terms of cognitive phases in information acquisition and processing?

Hidden (semi)-Markov models (H(S)MMs). This class of statistical models (Rabiner, 1989; Yu, 2010) belongs to the class of Dynamics Bayesian Networks (DBNs, Murphy and Russell, 2002; Koller et al., 2009). DBNs are probabilistic graphical models that compactly represent the joint distribution of a set of random variables. Their graphical structure provides knowledge concerning the dependencies and independencies of the random variables in order to identify how random variables influence each other. Moreover, they enable density estimation, thanks to their parameters. Probabilistic requests are sped up using inference to estimate the value of variables given information concerning other variables as evidence (Nagarajan et al., 2013). HSMMs may also be characterized as latent-variable models, which means that not all their random variables are observed. More particularly, an HSMM is composed of a double stochastic process. The former is observed while the latter is a latent semi-Markov chain, which preconditions the first process, and is used to uncover the changes of (semi-Markovian) dynamics in the observations. This makes HSMMs perfectly suited to uncover and segment latent reading strategies that drive observed features of eye movements over a sequence. However, one of the main cons of latent models is that their parameters need to be estimated with an iterative procedure called Expectation-Maximization (EM, McLachlan and Krishnan, 2007), which finds a local maximum of the likelihood of the parameters that sometimes might not be good enough. A common strategy with latent structure models is to perform random restart of EM (Biernacki et al., 2003). Such procedure for HSMMs does not exist yet and is a current requirement.

Co-recording electroencephalograms. The eye-mind link assumption suggests that the location of an observer's gaze partially reflects what is being processed in his mind at that time (Reichle and Reingold, 2013). Eye movements therefore constitute natural markers for time-locking the ongoing neural activity with respect to eye-movement events, such as fixations. The co-registration of eye movements and EEGs is generally analyzed under a framework called eye fixation-related potential (EFRP) (Dimigen et al., 2011) and aims at detecting delayed electrical changes produced by the nervous system in response of an external stimulus such as a cognitive activity. This analysis is conducted on the time domain by time-locking signals and averaging EEGs within a window to bring out a specific pattern (Luck, 2014). However, little is known about reading in more complex settings such as free text exploration (Frey et al., 2018). Another approach is to study repetitive patterns of neural activity called neural oscillations or brain waves on the frequency domain (Neuper and Klimesch, 2006). Few studies only addressed both concomitant acquisitions of eye movements and EEGs on the frequency domain (Seidkhani et al., 2017) and none of them analyzed phases (reading strategies) within a sequence.

Coupled models. Coupled H(S)MMs (Zhong and Ghosh, 2001; Natarajan and Nevatia, 2007) have been proposed to model interactions between multiple signals with different latent dynamics. However, there is currently no model that may handle heterogeneous data types such as eye movements and EEGs. Moreover, word semantical access is performed with a latency with respect to eye-movement activity and is known to involve different types of EEG patterns according to the cognitive processes involved (Frey et al., 2018). Therefore, a data-driven and automatic procedure to synchronize and segment eye movements and EEGs sequences into interpretable reading strategies is an unaddressed challenge.

Outline of the PhD

The organization of this thesis will be articulated around four chapters bridging probabilistic notions along with eye-movement and EEG analysis concepts.

In Chapter 1, we introduce the subject from a probabilistic perspective. We recall different probabilistic concepts before reviewing models that belong to the class of dynamic Bayesian networks to handle temporal signals. We put the light on Hidden semi-Markov Models, how they can be interrogated to perform inference, and how their parameters can be learned from data.

In Chapter 2, we first introduce the eye-movement context, past studies and the experiment on which the analysis was conducted to better justify what we aim at: the preprocessing of eye-movement features and their segmentation into interpretable cognitive processes (reading strategies) with HSMMs. The chapter contains an interlude on what we pointed out to be a key point in this study: the search of the highest maximum likelihood through adequate EM initialization when learning parameters.

Then in Chapter 3, we propose the use the model learned in Chapter 2 to segment the data of our experiment and perform an a posteriori analysis on model covariates with respect to reading strategies. More eye-movement features (internal covariates) are treated but also textual information (external covariates), corresponding to texts that users read during the experiment, and individual effects. At last, we analyze EEGs on the time-frequency domain with respect to eye-movement segmentation to highlight characteristic patterns on some given bands.

Finally in Chapter 4, we describe a new model coupling both eye-movement and EEG data that we call asynchronous heterogeneous hidden semi-Markov model. We also provide a wide range of possibilities to model the delayed interaction of these two signals. Finally, we raise and discuss many practical issues of this new model that leaves the door open to further improvements.

Chapter 1

Introduction to statistics of stochastic processes

Contents

1 Probabilistic framework

1.1 Preliminaries on random variables

Let us define $(\Omega, \mathcal{F}, P)$ a probability triple with:

- Ω being the sample space, i.e. the set of possible outcomes, towards another measurable state space E,
- $\mathcal{F}$, the set of all possible events,
- P, a function mapping events to probabilities s.t. $P : \mathcal{F} \to [0,1]$ with $P(\Omega) = 1$.

A **random variable** X is a measurable function which maps the set of all events to a measurable space:

$$X : \Omega \to E.$$

In order to make notations clearer, we shall denote E as $\mathscr{X}$. Answering the question "How likely does X take a certain set of values?" or equivalently "How likely is $X \in A$, where $A \in \mathcal{F}$?" is the same as measuring the event $\{\omega : X(\omega) \in A\}$, also written as $P(\{\omega \in \Omega | X(\omega) \in A\})$ or much more simply: $P(X \in A)$.

A random variable can take different forms according to the nature of $\mathscr{X}$. It is said to be:

- **continuous** if $\mathscr{X}$ is (infinite) uncountable. In general, $\mathscr{X} \subset \mathbb{R}$.
- **discrete** if $\mathscr{X}$ is countable (finite or infinite). For example $\mathscr{X} \subset \mathbb{N}$.

Additionally, we focus on the two most important forms of a discrete variables: **ordinal** for which the order of every value of Ω matters while it does not for a **nominal** variable.

In a more general setup, it should also be mentioned that $\mathscr{X}$ can take different dimensions. It is:

- **multivariate** when $\mathscr{X} \subset \mathbb{R}^n$, with $n \geq 2$,
- **univariate** when $n = 1$.

Further, we denote $\mathbf{X} = (X_1, ..., X_N)$ a set of random variables associated with its realizations $\mathbf{x} = (x_1, ..., x_N)$. If any random variable X_n or its realization x_n is multivariate, we note m as its m-th dimension and so we write $X_n^{(m)}$ and $x_n^{(m)}$ respectively. Hence the notation $x_n = \{x_n^{(m)}\}_{m \in [\![1,M]\!]}$ does not pay importance about the dimension of the variable. Moreover, we note D, a dataset containing realizations of one or more random variables s.t.

$$D = \{x_n\}_{n \in [\![1,N]\!]}.$$

1.2 Probability distributions

So far we have noted that P is used to map events to probabilities. We call **probability distribution** the set of functions which maps every event of Ω to a probability.

In this subsection, we focus on describing probability distributions which is used subsequently.

A **discrete probability distribution** of a discrete random variable X is entirely defined by its probability mass function (PMF) P_X where:

$$P_X(X = x) = p_X(x), \forall x \in \mathscr{X},$$

for which an interval can be easily computed by summing over all the elements of a given interval:

$$P_X(x_{inf} < X < x_{sup}) = \sum_{x_i \in [\![x_{inf}, x_{sup}]\!]} p_X(x_i).$$

Focusing on parametric distributions, the parameter of such a distribution is noted θ and acts as a container of the events to probabilities P_X.

Multinomial distribution. If X is a discrete random variable then P_X can follow a multinomial distribution, it is noted $X \sim \mathscr{M}(\theta)$ with parameters $\theta = \{\theta_x | x \in \mathscr{X}\}$. In other terms, there is a one-to-one mapping between θ and $\mathscr{X}$. Therefore it has the following properties:

$$\theta_x \in [0,1], \forall x \in \mathscr{X},$$

$$\sum_{x \in \mathscr{X}} \theta_x = 1,$$

and we write $P_\theta(X = x) = p_\theta(x) = \theta_x$, the distribution of X parameterized by θ.

Geometric distribution. If X is an discrete random variable then P_X can follow a Geometric distribution, noted $X \sim G(\theta)$, defined by a single parameter $\theta \in [0,1]$ and a specific PMF:

$$P_\theta(X = x) = (1-\theta)^{x-1}\theta, \forall x \in [\![1,\infty]\!],$$

$$\sum_{x\in[\![1,\infty]\!]} p_\theta(x) = 1.$$

Note here that the geometric probability distribution is encoded by one single parameter using a specific PMF whereas the multinomial distribution had as many parameters, as the random variable had factors.

Negative Binomial distribution. If X is an discrete random variable then P_X can follow a negative binomial distribution $X \sim \mathcal{NB}(\theta)$, defined by two parameters $\theta = \{\theta_1, \theta_2\}$, with $\theta_1 > 0$, $\theta_2 \in [0,1]$ and the following PMF:

$$P_\theta(X = x) = \binom{\theta_1 + \theta_2 - 1}{\theta_2}(1-\theta_2)^{x-1}\theta_2^{\theta_1}, \forall x \in [\![1,\infty]\!]. \tag{1.1}$$

We remark that, while a geometric distribution counts the number of success till the first failure (or vice-versa), a negative binomial distribution counts the number of success till θ_1 failures. Hence if $\theta_1 = 1$, both are equivalent.

Poisson distribution. If X in an discrete random variable then P_X can follow a Poisson distribution $X \sim \mathcal{P}(\theta)$, defined by one single parameter $\theta > 0$ and the ensuing PMF:

$$P_\theta(X = x) = \frac{\theta^x e^{-\theta}}{x!}, \forall x \in \mathbb{N}.$$

The **continuous probability distribution** of a continuous random variable X denoted P_X is entirely defined by its **cumulative distribution function** (CDF) $F_X : \mathcal{X} \to [0,1]$, with $\mathcal{X} = \mathbb{R}$ and where:

$$F_X(x) = P_X(X < x), \forall x \in \mathbb{R}.$$

When it exists, it may also be defined by its **probability density function** (PDF) $f_X : \mathbb{R} \to \mathbb{R}^+$ s.t. :

$$F_X(x) = \int_{-\infty}^{x} f_X(t)dt, \forall x \in \mathbb{R}.$$

Multivariate normal distribution. If X is a continuous random vector following a multivariate normal distribution of dimension M, we write $X \sim \mathcal{N}(\theta)$ with $\theta = \{\mu, \Sigma\}$,

$\mu \in \mathbb{R}^M$, $\Sigma \in \mathbb{R}^{M\times M}$. We define its PDF $f_\theta : \mathbb{R} \to \mathbb{R}^+$, $\forall x \in \mathbb{R}^M$:

$$f_\theta(x) = \frac{1}{(2\pi)^{\frac{M}{2}}|\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)},$$

with $|\Sigma|$ being the determinant of Σ and ^{T} being the matrix transpose operator. In other words, μ is a $1 \times M$ column vector representing the mean, and Σ is a $M \times M$ matrix representing the covariance.

1.3 Maximum Likelihood

In this subsection, we introduce the concept of likelihood, show how it is used in parameter estimation and propose estimators for a couple of discrete distributions based on histograms which will be used subsequently in this thesis.

1.3.1 Definition

Let a set of N independent and identically distributed (i.i.d.) random variables $\mathbf{X} = (X_1,...,X_N)$ with their respective realizations $\mathbf{x}$. We note the **likelihood function** $\mathcal{L}$ of θ given $\mathbf{X}$: $\mathcal{L} : \theta \to [0,1]$ as:

$$\mathcal{L}_\mathbf{X}(\theta) = p_\theta(\mathbf{x}) = p_\theta(x_1,...,x_N) = \prod_{x_i \in \mathbf{x}} p_\theta(x_i),$$

where the last step, $p_\theta(x_1,...,x_N) = \prod_{x_i \in \mathbf{x}} p_\theta(x_i)$, is possible because $X_1,...,X_N$ are i.i.d.. Along with the likelihood, we denote $\hat{\theta}$ the **maximum likelihood estimator** (MLE) of θ given $\mathbf{X}$ s.t.:

$$\hat{\theta} = \arg\max_{\theta\in\Theta} \mathcal{L}_\mathbf{X}(\theta),$$

which is often more conveniently achieved by maximizing the log-likelihood, if defined, that we further write $\mathscr{L}$.

1.3.2 Examples with Multinomial and Gaussian distributions

MLE for Multinomial distribution. If $\mathbf{X} \sim \mathscr{M}(\theta) = \{\theta_m | m \in \mathscr{X}\}$, likelihood of θ is as follows:

$$\mathcal{L}_\mathbf{X}(\theta) = \prod_{m\in\mathscr{X}} \theta_m^{\sum_{x\in\mathbf{x}} \mathbb{1}\{x_i=m\}},$$

where $\mathbb{1}$ is the indicator function which is equal to 1 if the condition is satisfied and 0 if not. The maximization problem can then be written as follows:

$$\begin{aligned} \max \qquad & \mathscr{L}(\theta) = \sum_{m \in \mathscr{X}} \sum_{x \in \mathbf{x}} \mathbb{1}\{x = m\} \log \theta_m \\ \text{subject to} \quad & \sum_{m \in \mathscr{X}} \theta_m = 1, \forall \theta_m \geq 0. \end{aligned}$$

which can be equivalently written using its Lagrangian:

$$\max \mathscr{L}(\theta) - \lambda(1 - \sum_{m \in \mathscr{X}} \theta_m),$$

where λ is called the Lagrangian multiplier. Since the optimization problem is convex, the maximum is reached by finding the values for which the partial derivatives of θ_m and λ are equal to 0, leading to the MLE for θ_m:

$$\theta_m = \frac{\sum_{x \in \mathbf{x}} \mathbb{1}\{x = m\}}{\sum_{n \in \mathscr{X}} \sum_{x \in \mathbf{x}} \mathbb{1}\{x = n\}} \tag{1.2}$$

which is easily interpreted as the empirical frequencies.

MLE for multivariate Normal distribution. Let $\mathbf{X} \sim \mathscr{N}(\mu, \Sigma)$, where $\mathbf{X}$ is a set of i.i.d. random multidimensional variables, and $\mathbf{x}$ the associated realizations, log-likehood is given by:

$$\begin{aligned} \mathscr{L}_{\mathbf{X}}(\mu, \Sigma) &= \prod_{x_n \in \mathbf{x}} \log p_{\mu, \Sigma}(x_n) \\ &= -\left[\frac{NM}{2} \log(2\pi) \frac{N}{2} \log(|\Sigma|) + \frac{1}{2} \sum_{n=1}^{N} (x_n - \mu)^T \Sigma^{-1} (x_n - \mu) \right], \end{aligned}$$

where ^{T} is the matrix transpose operator, $^{-1}$ the inverse matrix operator and M the dimension of the random variables. In this case, the log-likelihood is not concave with respect to the pair of parameters (μ, Σ). They are concave w.r.t. to μ for Σ fixed but the contrary is not true. The MLE of μ is therefore given by:

$$\hat{\mu} = \frac{1}{N} \sum_{n=1}^{N} x_n,$$

i.e. the empirical mean written $\bar{x}$, while the MLE of Σ is:

$$\bar{\Sigma} = \frac{1}{N}\sum_{n=1}^{N}(x_n - \bar{x})^T(x_n - \bar{x}).$$

If M is large, the MLE of Σ can quickly lead to the estimation of a very large number of parameter ($M \times M$) That is often not wanted and we usually prefer a parsimonious model with less parameters. We refer to Fan et al. (2016) for a review on sparse matrix estimation.

1.3.3 Computing ML from frequency tables

Sometimes, realizations of random variables are not directly accessible but observed through frequency tables. For example, a dataset might be pre-processed and might contain frequencies only or one might be constrained by modeling issues to replace a multinomial distribution by any parametric distribution. The latter case will be of interest for us. Thus, we subsequently describe how to derive log-likelihood for key distributions, i.e. Geometric and Negative Binomial, given frequency tables. The following results are based on the theoretical results provided by Johnson et al. (2005) but with a more practical perspective.

When it comes to fitting discrete distribution, one of the main issue is related to the fact that some discrete distributions have no parameter dedicated to localization. Their parameter describe the localization, variance and skewness at the same time. A common practice is to introduce a shift parameter represented by a scalar and use an ad-hoc loop procedure to find the best shift parameter which maximizes the likelihood.

Geometric distribution. Considering $\mathbf{x} = (x_1, ..., x_N)$ realizations from i.i.d. $\mathbf{X} = (X_1, ..., X_N) \sim G(\theta)$, the MLE of θ is as follows:

$$\hat{\theta} = \frac{1}{\hat{X}_N},$$

with $\hat{X}_N$ being the sample mean.

We now remind the expectation of a discrete random variable $\mathbb{E} : X \to \mathbb{R}$ s.t.

$$\mathbb{E}(X) = \sum_{x \in \mathscr{X}} x p_X(x). \tag{1.3}$$

It is also well known result that the expectation of a Geometric distribution is:

$$\mathbb{E}(X) = \frac{1}{\theta}. \tag{1.4}$$

The goal here is to give a parametric MLE for θ noted $\theta^{(p)}$ given from its non parametric (multinomial) MLE $\hat{\theta}^{(np)} = \{\hat{\theta}_x^{(np)} | x \in \mathscr{X}\}$, which has previously been estimated. The case of the Geometric distribution is quite straightforward. By substituting (1.4) into (1.3), the estimate of θ is given by:

$$\hat{\theta}^{(p)} = \frac{1}{\sum_{x \in \mathscr{X}} x p_{\theta^{(np)}}(x)}.$$

From a practical aspect, $\mathscr{X}$ is usually upper bounded and P_X is known and estimated by the MLE for categorical variable as showed in equation (1.2).

Remark. Computing parameters for each distribution given frequency tables is usually as straightforward as for the Geometric distribution and generalizes as long as the MLE of the parameters has a closed-form.

Negative Binomial distribution. Finding good estimates of the parameters of the negative binomial distribution is a bit more challenging since the MLE of the parameters has no closed-form. The following papers discuss several aspect to find a good MLE Fisher (1941); Wise (1946); Bliss and Fisher (1953); Ross et al. (1980); Ross and Preece (1985); Clark and Perry (1989) such as iterative procedures, initial parameters, bad behaviors of the MLE when the sample variance is much lesser than the sample mean.

With $\mathbf{x} = (x_1, \ldots, x_N)$ realizations from i.i.d. $\mathbf{X} = (X_1, \ldots, X_N) \sim \mathscr{N}\mathscr{B}(\theta), \theta = \{\theta_1, \theta_2\}$, the PMF P_X is described by equation (1.1) and the MLE is given by the following system of equations:

$$\begin{cases} \log(1 + \frac{\hat{X}}{\hat{\theta}_1^{(np)}}) = \sum_{i=1}^{\infty} \left(\frac{1}{\hat{\theta}_1 + i - 1} \sum_{j=i}^{\infty} p_\theta^{(p)}(x_i) \right), \\ \hat{\theta}_2^{(p)} = \frac{\hat{\theta}_1^{(p)}}{\hat{\theta}_1^{(p)} + \hat{X}}. \end{cases}$$

The first equation is is solved using an iterative procedure and gives a value of θ_1 while θ_2 is calculated in the second equation by injecting θ_1. To retrain the search space, a good practice is to set the initial values using the moment estimators which are computed in closed-form:

$$\overline{\theta_1} = \frac{\overline{X}}{S^2},$$

and

$$\overline{\theta_2} = \frac{\overline{X}^2}{(S^2 - \overline{X}^2)},$$

where $\overline{X}$ is the sample mean of X and S^2 its variance.

1.4 Joint probability distributions

Let (X,Y) a tuple of two random variables which are not necessarily independent and identically distributed and that therefore interact with each other. The study of these interactions is encapsulated in their **joint probability distribution** (JPD) denoted P, defined $\forall x \in \mathscr{X}$ and $y \in \mathscr{Y}$ by $P(X = x, Y = y)$. The JPD is a full description of the interactions between of **X** because it also encapsulates:

- the **marginal probability distributions** (MPD) P_X, P_Y of X and Y respectively. They can be computed using the **sum rule** defined as:

  $$P(X = x) = \sum_{y \in \mathscr{Y}} P(X = x, Y = y).$$

- The **conditional probability distributions** (CPD) $P_{X|Y=y}, P_{Y|X=x}$ of X given $Y = y$ and Y given $X = x$ respectively. They are computed using the **product rule** defined as:

  $$P(X = x|Y = y) = \frac{P(X = x, Y = y)}{P(Y = y)},$$

  hence

  $$P(X = x, Y = y) = P(X = x|Y = y)P(Y = y),$$

From the product rule together with the symmetry property of the JPD s.t. $P(X = x, Y = y) = P(Y = y, X = x)$, we obtain the following property:

$$P(Y = y|X = x) = \frac{P(X = x|Y = y)P(Y = y)}{P(X = x)}$$

which is called the **Bayes theorem**. The denominator can be rewritten using the distributions found in the numerator:

$$P(Y=y|X=x)=\frac{P(X=x|Y=y)P(Y=y)}{\sum_{y\in\mathscr{Y}}P(X=x|Y=y)P(Y=y)},$$

allowing us to compute the conditional of $Y|X$ while having only access to the information of the conditional $X=x|Y=y$ and of the marginal $Y=y$. It should also be noticed that the denominator acts as a normalizing constant so that the probabilities sum to one.

Furthermore, for a set of N categorical variables $\mathbf{X}=(X,...,X_N)$, we also introduce the **chain rule** $(x,...,x_N)\in(\mathscr{X}_1,...,\mathscr{X}_N)$:

$$\begin{aligned}P(X=x,...,X_N=x_N)=\ &P(X=x|Y=y,...,X_{N-1}=x_{n-1})\\ &P(Y=y|X_3=x_3,...,X_{N-1}=x_{n-1})\\ &...\\ &P(X_{N-1}=x_{n-1})\end{aligned}$$

which is another direct consequence of the product rule by propagating it by induction.

Finally, we recall the **independence** of two random variables X and Y, denoted $X\perp Y$ if $\forall x\in\mathscr{X}$ and $y\in\mathscr{Y}$:

$$P(X=x,Y=y)=P(X=x)P(Y=y)$$

which can be seen as a special case of the product rule when $P(X=x|Y=y)=P(X=x)$, meaning that knowing Y adds no knowledge to X.

To prevent notations to get too heavy, when talking about the distribution of $p(X=x,Y=y),\forall x\in\mathscr{X},\forall y\in\mathscr{Y}$, we often abbreviate this notation by $P(X,Y)$ whereas $P(X,Y)$ indicates the joint distribution of (X,Y). This scheme also applies to marginal and conditional distributions.

1.5 Discrete Markov Chain

Stochastic process. Let us define a probability space $(\Omega,\mathcal{F},P)$ where the random variables X_t are indexed by time t and take values into a same measurable space $\mathscr{X}$, a stochastic process is a time evolving process s.t. $\{X(t,\omega)|t\in\mathscr{T},\omega\in\Omega\}$. When considering $\mathscr{T}$ finite and so $\mathscr{T}=[\![1,T]\!]$, we write as $\{X_t\}_{t\in T}$, or sometimes simply $\mathbf{X}$.

Stationarity. A stochastic process $\{X_t\}_{t\in[\![1,T]\!]}$ w.r.t to a probability measure P_X is said to be **weakly stationary** iff:

- $E(X_t) = \mu$, $\forall t \in T$ with μ finite, the mean, i.e. it's expectation is constant regarding time,
- $Cov(X_t, X_{t+\tau}) = S_\tau^2$, $\forall t, \tau \in T$ with S_τ^2 the variance i.e. its covariance only depends on a lag τ but not t.

Furthermore, it is said to be **strongly stationary** iff:

$$P_X(X_t, ..., X_{t'}) = P_X(X_{t+\tau}, ..., X_{t'+\tau}), \forall \tau, t \in T, \forall t' > t,$$

which can be seen as the fact that the joint distribution of the stochastic process does not change with time, it is said to be identically distributed.

Discrete Markov Chain. Considering a stochastic process $\{X_t\}_{t\in[\![1,T]\!]}$, with $X_t \in \mathscr{X}$ with $\mathscr{X}$ finite, a discrete Markov chain assumes that the joint probability distribution of the stochastic process is written as follows:

$$\begin{aligned} P(\mathbf{X}) &= P(X_1)P(X_2|X_1)P(X_3|X_2)...P(X_T|X_{T-1}) \\ &= P(X_1)\prod_{t=2}^{T} P(X_t|X_{t-1}). \end{aligned} \tag{1.5}$$

A Markov chain describes the idea that the best way to predict the future, X_{t+1}, is encapsulated in the current information X_t. This assumption is called the Markov property.

Parameters. The equation of the joint likelihood of a Markov Chain (1.5) highlights a first set of parameters describing $P(X_1)$ called **initial probabilities** s.t. $\forall k \in \mathscr{X}$,

$$\pi_k \equiv P(X_1 = k), \tag{1.6}$$

π is then a vector of size $1 \times K$, with K being the cardinal of $\mathscr{X}$.

Now, assuming that the transition function $P(X_t|X_{t-1})$ is **time-invariant**, or homogeneous, this leads to parameterizing the right term of (1.5), $\forall t \in [\![2,T]\!], P(X_t|X_{t-1})$ by a single set of parameters called the **transition matrix** s.t. $\forall k,k' \in \mathscr{X}$,

$$A_{k'k} \equiv P(X_t = k | X_{t-1} = k'), \tag{1.7}$$

with $\forall k \in \mathscr{X}, \sum_{k' \in \mathscr{X}} A_{k'k} = 1$, describing that each row sums to one. A is then a matrix of size $K \times K$. The terminology of stochastic matrix can also be found in the literature because of the last property.

Parameter estimation. From equation (1.5), we rewrite the likelihood as follows:

$$\begin{aligned} \mathscr{L}_{X_1,\dots,X_T}(\pi,A) &= P_{\pi,A}(X_1 = x_1,\dots,X_T = x_T) \\ &= \prod_{k=1}^{K} \pi_k^{\mathbb{1}(x_1=k)} \prod_{t=2}^{T} \prod_{k=1}^{K} \prod_{k'=1}^{K} A_{k'k}^{\mathbb{1}(x_t=k,x_{t-1}=k')}. \end{aligned} \tag{1.8}$$

We remark that maximizing the likelihood w.r.t. π leads to a count function, while maximizing the likelihood w.r.t. A leads a conditional count function. From there, we identify that maximizing π is equivalent as maximizing the likelihood of multinomial distributions as we performed in equation (1.2) using the Lagrangian. Hence the MLE of π is given by:

$$\hat{\pi}_k = \frac{\mathbb{1}(x_1 = k)}{\sum_{k'=1}^{K} \mathbb{1}(x_1 = k')}, \tag{1.9}$$

and the MLE of A is simply solved using the Lagrangian with K separate optimization problems:

$$\hat{A}_{k'k} = \frac{\sum_{t=2}^{T} \mathbb{1}(x_t = k, x_{t-1} = k')}{\sum_{t=2}^{T} \sum_{k''=1}^{K} \mathbb{1}(x_t = k'', x_{t-1} = k')}. \tag{1.10}$$

Left-to-Right. If the transition matrix A is lower triangular, i.e. it is filled with zeros bellow the diagonal then the transition matrix is said to be **left-to-right**. Such modeling hypothesis are often used in speech recognition and require that multiple sequences are available for parameter estimation (Juang and Rabiner, 1985; Rabiner, 1989; Varga and Moore, 1990; Eddy, 1998).

Chapman-Kolmogorov equations. An application to the Chapman-Kolmogorov equations to a discrete Markov chain states that:

$$P(X_{t+t'+t''} = k | X_t = k') = \sum_{k''=1}^{K} P(X_{t+t'+t''} = k | X_{t+t''} = k'') P(X_{t+t''} = k'' | X_t = k'), \tag{1.11}$$

which can be rewritten in the following matrix form $P(X_{t+t'} = k|X_t = k') = (A^{t'})_{k'k}$. Thus, it means that simulating t' time steps of a discrete Markov Chain can be done by powering up the transition matrix by t', leading to matrices with paths of length t' from one state to another.

Properties. Given a transition matrix A, a state $k \in [\![1,K]\!]$ may have different properties such as:

- **absorbing** iff $A_{kk} = 1$, i.e. a state which can not be exited once entered,
- **recurrent** iff $\sum_{n=0}^{+\infty}(P^n)_{kk} = +\infty$, i.e. if once entered in state k, there is a probability of 1 to return in an infinite (or finite, unbounded) amount of time,
- **transient** iff $\sum_{n=0}^{+\infty}(P^n)_{kk} < +\infty$, i.e. if once entered in state k, there is a probability lesser than 1 of over returning in a infinite amount of time.

A Markov chain is said to be **irreducible** if $\forall k,k' \in [\![1,K]\!], \exists n \in \mathbb{N}$ s.t. $(A^n)_{k'k} > 0$. In other words, if all states are communicating two by two which happens when all states are recurrent.

The **period** of a state k is defined as $c(k) = gcd\{n|(A^n)_{kk} > 0\}$, with gcd calculating the greatest common divisor, interpreted as the period at which returning to a state k is possible. A state k is called **periodic** with period $c(k)$ if $c(k) > 1$, else it is called **aperiodic**. A state that is recurrent and aperiodic is said to be **ergodic**. If all states are ergodic, then the Markov chain is ergodic.

A homogeneous Markov chain is **stationary** iif $P(X_t = k) = P(X_1 = k) = \pi_k$ and its stationary distribution is noted π^*. Since $\forall k \in \mathscr{X}, P(X_t = k) = P(X_1 = k)\sum_{k' \in \mathscr{X}} P(X_t = k'|X_1 = k)$, then the distribution is stationary iff the initial distribution satisfies $P(X_t = k) = P(X_t = k)\sum_{k' \in \mathscr{X}} P(X_t = k|X_{t-1} = k')$, or $\pi = \pi A$. A stationary distribution π^* may not be unique, though if the Markov chain is irreducible and ergodic, then it has a unique stationary distribution which is equal to its limiting distribution defined by $\pi^*_k = \lim_{n \to +\infty}(A^n)_{k'k}$.

Sojourn distribution. If a state k has a zero on its transition matrix diagonal, i.e. $A_{kk} = 0$, it is obvious to see that the sojourn time (also called dwell time) in the state, once entered, is of constant duration 1. However, when $A_{kk} \neq 0$, the duration in the state is random with, at every time, a probability A_{kk} of staying in k and a probability $1 - A_{kk}$ of exiting k.

Lemma 1. *Let* $\{X_t\}_{t\in T}, X_t \in \mathscr{X}$*, a discrete Markov chain of transition matrix A, the state* $X_t = k$ *residual time* R_t *at time t s.t.* $R_t = \min\{t' > t | X_{t'} \neq X_t\}$ *has a Geometric distribution of parameter* $1 - A_{kk}$*.*

Proof.

$$\begin{aligned}
P(R_t = u) &= P(X_{t+u+1} \neq k, X_{t+u} = k, ..., X_{t+1} = k | X_t = k, X_{t-1} \neq k) \\
&= P(X_{t+u+1} \neq k, X_{t+u} = k) P(X_{t+u} = k | X_{t+u-1} = k) ... P(X_{t+1} = k | X_t = k) \\
&= \sum_{l \neq k} P(X_{t+u+1} = l | X_{t+u} = k) \prod_{v=1}^{u} P(X_{t+v} = k | P(X_{t-1+v} = k) \\
&= (1 - A_{kk}) A_{kk}^{u} \\
&= G(1 - A_{kk})(u)
\end{aligned}$$

□

Markov order. Sometimes applying the Markov property on the distribution of X_t given its most recent predecessor X_{t-1} may be too constraintful because of its short memory. A Markov chain of *m*-th order relaxes the Markov property allowing longer time dependency. Hence the conditional probability distribution is, $\forall t' < t$, written:

$$P(X_t | X_{t-1}, ..., X_1) = P(X_t | X_{t-1} ..., X_{t-t'})$$

where the right term becomes the transition matrix A of size $K^{t'+1}$. The Markov order has received a lot of attention in the literature where model selection procedures have tried to find automatically the best Markovian order based on information theory criterion (Katz, 1981; Rabiner, 1989; Finesso, 1992; van Handel, 2011), hypothesis testing, or Bayesian nonparametric approaches (Mochihashi and Sumita, 2008). Several methods for this purpose are summarized by Cappé et al. (2006).

2 Dynamic Bayesian Networks

In this section, we briefly give some reminders of statistical modeling, inference and learning using Dynamic Bayesian Networks (DBNs) (Dean and Kanazawa, 1989) which are central to probabilistic signal processing. More particularly, our interest is focused on DBN with latent, or unobserved, random variables which allow us to recover some indirectly observed structure though the observed data. Besides, we present the most well known instance of DBN called Hidden Markov Model (HMM) along with some of its extensions overcoming its limitations. An introduction to DBN is presented in Ghahramani (2001) while a more exhaustive review of DBN can be found in Murphy and Russell (2002).

2.1 Representation

A DBN is an instance of Bayesian Network (BN). A DBN has the particularity of modeling a dynamic system or a stochastic system, connoting that it encodes a time evolving structure of the data whereas the structure of the BN is constant over time.

A BN is a specific instance of a Graphical Model. The model is represented by a graph and to each vertex corresponds a random variable. BNs have the particularity of having directed edges, representing conditional probability distributions and independence relationships. Hereunder, we briefly review DBN representations through graphs.

More formally, we are interested in a tuple of random variables $\mathbf{X} = (X_1, \ldots, X_N)$, their realizations $\mathbf{x} = (x_1, \ldots, x_N) \in (\mathscr{X}_1, \ldots, \mathscr{X}_N)$ and their joint distribution $P(X_1 = x_1, \ldots, X_N = x_N)$. A Bayesian Network is defined as a tuple $\mathscr{B} = (G, \theta)$ where:

- G is a directed acyclic graph, a tuple (V, E):
  - **vertices:** V is a finite set of vertices where each node n is associated to a random variable X_n,
  - distinct directed **edges** with $E \subset V^2$. Each oriented edge (or arc) is an ordered tuple $(x, y) \in E$.
- $\theta = (\theta_1, \ldots, \theta_N)$, a set of parameters where each parameter θ_n is associated to a random variable X_n. θ_n encodes the conditional probability distribution $P(X_n | \mathbf{X}_{pa(n)})$ where $pa(n)$ relates to the parent function which indicates the parents of n in the graph G, i.e. $pa(n) = \{\forall m \in V | (m, n) \in E\}$.

A BN encodes the distribution factorization of $\mathbf{X}$: $\forall \mathbf{x} \in (\mathscr{X}_1, \ldots, \mathscr{X}_N)$ as follows:

$$p_\theta(\mathbf{X}) = \prod_{n=1}^{N} p_{\theta_n}(x_n | \mathbf{x}_{pa(n)}) = \prod_{n=1}^{N} \theta_n,$$

with the main hypothesis that a random variable is independent of its non(descendents in the graph given its parents. This property, introduced in Verma and Pearl (1990), is called **d-separation** or conditional independence denoted $X_1 \perp X_2 | X_3$ for X_1 is independent of X_2 conditionally to X_3.

Furthermore, we define $\mathbf{X} = (X_1, ..., X_T)$, where $X_t \in \mathbf{X}$ now represent a time slice at time t and is a composite random variable of M different compounds s.t. $X_t = (X_{t,1}, ..., X_{t,M})$. As a consequence, a vertex from a graph is noted as a tuple with an additional value indicating its time slice s.t. $\forall t \in [\![1,T]\!], m \in [\![1,M]\!], (t,m) \in V$. We define a **Dynamic Bayesian Network** to be a pair of BNs $(\mathcal{B}_1, \mathcal{B}_2)$ where $\mathcal{B}_2$ is a two-slice temporal BN defined as:

$$P(X_t | X_{t-1}) = \prod_{m=1}^{M} P(X_{t,m} | \mathbf{X}_{pa(t,m)})$$

where $\mathbf{X}_{\pi(t,m)}$ contains the ancestors of $X_{t,m}$ on the same time slice t as well as those on the previous time slice $t-1$ i.e. $pa(t,m) \in \{\forall (n,t) \in V | ((n,t-1),(m,t)) \bigcup ((n,t),(m,t)) \in E\}$. $\mathcal{B}_1$ is a BN corresponding to the prior distribution at time $t = 1$:

$$P(X_1) = \prod_{m=1}^{M} P(X_{1,m} | \mathbf{X}_{\pi(1,m)})$$

which has a specific representation because it has no temporal ancestor but still has the same ancestors on the time slice 1 as other time slices i.e. $\pi(1,m) \in \{\forall (n,1) \in V | ((n,1),(m,1)) \in E\}$.

The joint distribution of a DBN is denoted:

$$P(\mathbf{X}) = \prod_{t=1}^{T} \prod_{m=1}^{M} P(X_{t,m} | \mathbf{X}_{pa(t,m)}) = \prod_{t=1}^{T} \prod_{m=1}^{M} \theta_{t,m} | \theta_{\pi(t,m)}. \tag{1.12}$$

In order to maintain tractability from the point of view of the number of parameters as well as for inference, which is treated in the next section, we state several assumptions, extracted from Nagarajan et al. (2013), that Dynamic Bayesian Networks should verify:

Assumption 1. *The stochastic process* $\mathbf{X}$ *is first order Markovian.*

Assumption 2. *The process is homogeneous over time.*

Remark. Dynamic Bayesian Networks represent a global framework for modeling stochastic processes and come with their own global tools for inference and learning.

However, in this thesis, we take a particular look on specific DBN instances, considering that the graph is known. Representing them in a similar framework is a way to provide links and comparisons easily between different models as well as common tools.

2.2 Inference

The Dynamic Bayesian Network structure relates to links between the random variables in the data. Arcs are used to describe how well the parent random variable explain its child but should not be interpreted as causality Pearl (2009). For a given graphical structure, there are different tasks to be performed known as **parameter inference** and **state inference** as in Cappé et al. (2009). Most commonly, the former task is often also called "learning" while the latter is found as "inference", see Murphy and Russell (2002), Ghahramani (2001), Nagarajan et al. (2013), Rabiner (1989).

In this section, we focus on inference as in state inference which aims at going beyond the probability distributions encoded by the model itself by answering some specific queries about the data like the state of a set of variables while the state of another set of variables is provided as an **evidence**.

More formally, we wish to investigate the effect of a piece of evidence $\mathbf{E}$ on the distribution of a set of variable $\mathbf{X}$ given the network structure $\mathcal{B} = (G, \theta)$, that is the conditional probability distribution $P(\mathbf{X}|\mathbf{E}, \mathcal{B})$.

Most of the time, we are provided a **hard evidence** which is a direct instance of a non empty set of random variables:

$$\mathbf{E} = \{X_{i_1} = e_1, ..., X_{i_k} = e_k\}$$

with $i_1, ..., i_k = [\![1, n]\!]$ and, $(X_{i_1}, ..., X_{i_k}) \in (\mathcal{X}_{i_1}, ..., \mathcal{X}_{i_k})$ respectively. However, another common issue is the **soft evidence**, when the probability distributions of a set of random variables are being provided rather than instantiations, i.e.

$$\mathbf{E} = \{X_{i_1} \sim \theta_{X_{i_1}}, ..., X_{i_k} \sim \theta_{X_{i_k}}\}.$$

Such types of evidence are mainly used to perform hypothesis testing while hard evidences are used to compute conditional probability distribution or their maximum and is called maximum a posteriori (MAP).

Conditional probability distribution.

$$P(\mathbf{X}_Q|\mathbf{E},\mathcal{B}),$$

with $Q \subset [\![1,n]\!]$ and $\mathbf{X}_Q$, a queried subset of $\mathbf{X}$

Maximum a posteriori.

$$\mathbf{x}_Q^* = \arg\max_{\mathbf{x}_Q} P(\mathbf{X}_Q = \mathbf{x}_Q|\mathbf{E},\mathcal{B})$$

Subsequently, we focus on inference given hard evidences. Moreover, in the context of Dynamic Bayesian Networks, that is in presence of temporal aspect, there are a couple of queries of interest concerning the distribution of $X_{t,i}$, the random variable associated with the node i at time t conditionally to other nodes at time $1,...,T$.

- **Filtering** consists in querying the network "online" about the current state given all the past states and the current states, that is when $t = T$. It is called so because it does not only use data at t but also the previous one to filter the noise.
- **Smoothing** queries the network "offline" about the state of some time t when $t < T$ meaning that it also uses information from the future to complete its computation.
- **Prediction** is a query about the future $t > T$ for which no evidence has yet been observed.

These distributions can be computed using different inference fashions. An overview of these approaches can be found in Murphy and Russell (2002), here we briefly remind inference categories while providing a non exhaustive list.

Exact inference. Since summing (resp. integrating) over all the possible variables in the network would result in a non-polynomial complexity (Cooper, 1990), exact inference relies on a cascade application of the Bayes Theorem along with the conditional independence property in order to provide exact values of the conditional distributions. Algorithms falling into this category are: the message passing algorithm (Kim and Pearl, 1983) which has specific instances as forward-backward algorithm or frontier algorithm (Zweig, 1996), junction trees (Dechter and Pearl, 1988; Smyth et al., 1997), variable elimination (Zhang and Poole, 1994; Dechter, 1999).

Approximate inference. Exact inference can quickly get intractable when there are too many random variables in the network. Approximate inference helps reducing the computation time, at the cost of an additional error on the approximation on a probability distributions. Approximate inference falls into two categories:

- **deterministic**, when there is no variability in the result of the inference. Such techniques usually consists in approximating the joint distribution by the product of their marginal. We can notably quote the Boyen-Koller algorithm Boyen and Koller (1998), the factored frontier algorithm Murphy and Weiss (2001) and their generalization: the loopy belief propagation Murphy et al. (1999). These algorithms are generic to dynamic Bayesian networks while for specific DBN instances such as Mixed Memory Markov Model Ghahramani and Hinton (2000) or factorial HMM Ghahramani and Jordan (1996) variational inference Jordan et al. (1999) is performed.

- **stochastic**, when the target distribution is computed using random processes based on Monte Carlo simulations from the joint probability distribution to approximate the conditional distribution given the query. There are two categories of stochastic algorithms: online, which regroup particle filtering algorithms Doucet et al. (2000) and offline such as likelihood weighting Fung and Chang (1990) Shachter and Peot (1990) and Monte Carlo Markov Chains Gilks et al. (1995).

In this thesis, we work with network structures for which the exact inference is tractable. Therefore, we focus on Forward-Backward types algorithms.

2.3 Learning with complete data

We consider a DBN $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ with $\mathcal{B}_1 = (G_1, \theta_1)$ and $\mathcal{B}_2 = (G_2, \theta_2)$, associated with random variables $\mathbf{X} = (\mathbf{X}_1, ..., \mathbf{X}_T)$ where each time slice $\mathbf{X}_t$ is itself a set of random variables $\mathbf{X}_t = (X_{t,1}, ..., X_{t,M})$ for which the structure (G_1, G_2) is known but the set of parameters θ_1, θ_2 is unknown. We also suppose we are in possession of a complete dataset D which contains N i.i.d. observations of $\mathbf{X}$.

Parameter learning or parameter estimation on complete data using maximum likelihood is straightforward. For this purpose, using the joint probability distribution of a DBN given by equation (1.12) along with the assumption (2), we write the log-likelihood of a DBN:

$$\begin{aligned}
\mathscr{L}_D(\theta_1,\theta_2) &= \log P_{\theta_1}(\mathbf{X}_1)\sum_{t=2}^{T} + \log P_{\theta_2}(\mathbf{X}_t|\mathbf{X}_{pa(t,m)}) \\
&= \sum_{m=1}^{M} \log P_{\theta_1}(X_{1,m}|\mathbf{X}_{\pi(1,m)}) + \sum_{t=2}^{T} \log P_{\theta_2}(X_{t,m}|\mathbf{X}_{\pi(t,m)}) \\
&= \sum_{n=1}^{N}\Big[\sum_{m=1}^{M} \log p_{\theta_{1,n}}(x^{(n)}_{1,m}|\mathbf{x}^{(n)}_{\pi(1,m)}) + \sum_{t=2}^{T} \log p_{\theta_{2,n}}(x^{(n)}_{t,m}|\mathbf{x}^{(n)}_{\pi(t,m)})\Big] \\
&= \sum_{m=1}^{M}\Big[\sum_{n=1}^{N} \log p_{\theta_{1,n}}(x^{(n)}_{1,m}|\mathbf{x}^{(n)}_{\pi(1,m)}) + \sum_{t=2}^{T} \log p_{\theta_{2,n}}(x^{(n)}_{t,m}|\mathbf{x}^{(n)}_{\pi(t,m)})\Big]
\end{aligned} \tag{1.13}$$

where $\theta_{1,n}$ is the set of parameter defining the separately distribution of $X_{1,m}|\mathbf{X}_{\pi(1,m)}$ (resp. $\theta_{2,n}$) which can be seen as a **global decomposition** of the local log-likelihood of each node given its parents Spiegelhalter and Lauritzen (1990) Koller et al. (2009) Ghahramani (2001). Indeed, in this form, each term can be locally maximized.

In the categorical case, both the DBN prior distribution $\log p_{\theta_1}(x^{(n)}_{1,m})$ and the conditional distribution $\log p_{\theta_2}(x^{(n)}_{t,m})$ can be estimated in a similar fashion combining both MLE of the prior distribution of the discrete Markov Chain, equation (1.9), and of its conditional distribution, equation (1.10). Basically for categorical data, each parameter is simply a normalized table containing counts of each occurrence given each occurrence of its parents in the data set.

2.4 Learning with incomplete data: the EM algorithm

In the presence of latent variable, the choice of model parameters denoted as θ is much more difficult. Hereunder, we discuss this procedure through the EM algorithm.

We denote the set of observed variables $\mathbf{X}$ existing in the set $\mathscr{X}$, and latent variables $\mathbf{S}$ with the corresponding value set $\mathscr{S}$.

When facing latent variables, one intuition could consists in computing $\mathscr{L}(\theta) = P(\mathbf{X};\theta) = \sum_{\mathbf{S}\in\mathscr{S}} P(\mathbf{X},\mathbf{S};\theta)$. Note that if $\mathbf{S}$ is continuous, sums are replaced by integrals. This task is usually difficult since in requires integrating/summing over $\mathscr{S}$. Hence, a procedure for maximizing the likelihood is the Expectation-Maximization (EM) algorithm, introduced by Dempster et al. (1977), reviewed in McLachlan and Krishnan (2007), which ensure to find a local maximum of the likelihood.

EM relies on the decomposition of the log-likelihood ((1.14)).

Lemma 2.

$$\log P(\mathbf{X};\theta) = f(q,\theta) + KL(q||p) \tag{1.14}$$

where,

$$f(q,\theta) = \sum_{\mathbf{S}\in\mathscr{Z}} q(\mathbf{S}) \log \frac{P(\mathbf{X},\mathbf{S};\theta)}{q(\mathbf{S})}$$

and,

$$KL(q||p) = -\sum_{\mathbf{S}\in\mathscr{S}} q(\mathbf{S}) \log \frac{P(\mathbf{S}|\mathbf{X};\theta)}{q(\mathbf{S})}$$

with $f(q,\theta)$ a functional of the probability distribution $q(\mathbf{S})$, $p = P(\mathbf{S}|\mathbf{X};\theta)$, and $KL(q||p)$ stands for the Kullback-Leibler divergence which satisfies $KL(q||p) \geq 0$, and is equal to zero when $q(\mathbf{S}) = P(\mathbf{S}|\mathbf{X};\theta)$.

Proof.

$$\begin{aligned}
f(q,\theta) + KL(q||p) &= \sum_{\mathbf{S}\in\mathscr{S}} q(\mathbf{S}) \log \frac{P(\mathbf{X},\mathbf{S};\theta)}{q(\mathbf{S})} - \sum_{\mathbf{S}\in\mathscr{S}} q(\mathbf{S}) \log \frac{P(\mathbf{S}|\mathbf{X};\theta)}{q(\mathbf{S})} \\
&= \sum_{\mathbf{S}\in\mathbf{S}} q(\mathbf{S}) \log P(\mathbf{X},\mathbf{S};\theta) - q(\mathbf{S}) \log q(\mathbf{S}) - q(\mathbf{S}) \log P(\mathbf{S}|\mathbf{X};\theta) + q(\mathbf{S}) \log q(\mathbf{S}) \\
&= \sum_{\mathbf{S}\in\mathscr{S}} q(\mathbf{S}) \log \frac{P(\mathbf{X},\mathbf{S};\theta)}{P(\mathbf{S}|\mathbf{X};\theta)} \\
&= \sum_{\mathbf{S}\in\mathscr{S}} q(\mathbf{S}) \log P(\mathbf{X};\theta) \\
&= \log P(\mathbf{X};\theta)
\end{aligned}$$

□

From (1.14) and since $KL(q||p) \geq 0$, it follows that $\mathscr{L}(\theta) \geq f(q,\theta)$. Hence that $f(q,\theta)$ is a lower bound of the log-likelihood, which is a pillar of the EM algorithm. Indeed, one can see that for an initial value of the parameters θ^{old}, optimizing the lower bound of the log-likelihood results in canceling the KL divergence, that is, setting $q(\mathbf{S}) = P(\mathbf{S}|\mathbf{X};\theta^{old})$. Maximization of the log-likelihood is achieved w.r.t. $q(\mathbf{S})$ while holding θ^{old} fixed, which corresponds to the E-step, and guaranties not to decrease the log-likelihood. Then, the M-step computes the new parameters θ^{new} by maximizing $f(q,\theta)$ w.r.t. θ while holding $q(\mathbf{S})$ fixed this time, i.e. $q(\mathbf{S}) = P(\mathbf{S}|\mathbf{X};\theta^{old})$, causing the lower bound to increase. we have:

$$\begin{aligned} f(q,\theta) &= \sum_{Z\in\mathscr{Z}} P(Z|X;\theta^{old}) \log \frac{P(X,Z;\theta)}{P(Z|X;\theta^{old})} \\ &= \sum_{Z\in\mathscr{Z}} P(Z|X;\theta^{old}) \log P(X,Z;\theta) - \sum_{Z\in\mathscr{Z}} P(Z|X;\theta^{old}) \log P(Z|X;\theta^{old}) \\ &= \mathscr{Q}(\theta,\theta^{old}) + const, \end{aligned} \tag{1.15}$$

where the right term is constant since it does not depend on θ, and the left term is the expected value of the complete-data likelihood with respect to the conditional distribution, i.e. $\mathbb{E}[\log P(\mathbf{X},\mathbf{S};\theta)|\mathbf{X},\theta^{old}] = \mathbb{E}_{\mathbf{S}|\mathbf{X};\theta^{old}}[\log P(\mathbf{X},\mathbf{S};\theta)]$. In other words, M-step resides in finding $\theta^{new} = \arg\max_\theta \mathscr{Q}(\theta,\theta^{old})$, and since $q(\mathbf{S})$ is known, it is usually as trivial as if there was no latent variable, i.e. as setting parameter values which cancel the partial derivatives of the log-likelihood. Afterwards, since in the M-step $q(\mathbf{S}) \neq P(\mathbf{S}|\mathbf{X};\theta)$, the KL divergence is now non-null and we can go back to the E-step and iterate over and over until convergence of the log-likelihood. Algorithm 1 corresponds to EM.

Algorithm 1: Expectation-Maximization algorithm

```
Expectation-Maximization ($\theta^{new}$, $\varepsilon$);
Input: $\theta^{new}$, a set of initial parameters. $\varepsilon$ a convergence tolerance.
repeat
    $\theta^{old} \leftarrow \theta^{new}$
    Compute $P(\mathbf{S}|\mathbf{X};\theta^{old})$ // E-step
    $\theta^{new} \leftarrow \arg\max_\theta \mathscr{Q}(\theta,\theta^{old})$ // M-step
until $|\mathscr{L}(\theta^{old}) - \mathscr{L}(\theta^{new})| < \varepsilon$;
Output: $\theta^{new}$, Parameters locally maximizing the log-likelihood
```

2.5 Special case: Hidden Markov Models

Introduced in 1966 by Baum and Petrie (1966), revisited many times notably by Rabiner (1989) who came up with a user friendly tutorial, Smyth et al. (1997) who suggested to link HMMs along with DBNs, Ephraim and Merhav (2002) who proposed a more theorical study of Hidden Markov Processes and Cappé et al. (2006) who reviewed all the HMM inference state of the art in a book. the Hidden Markov Model (HMM) is probably the most well known instance of Dynamic Bayesian Network for its computational efficiency and its performance in several signal processing domains such as time series prediction (Fraser, 2008), automatic speech recognition (Jurafsky and Martin, 2009) or part of speech tagging (Kupiec, 1992). The HMM is discrete-time finite-state homogeneous Markov chain observed through a discrete-time memoryless stationary channel. In other words, it is a double stochastic process. The former is a hidden discrete finite state Markov chain and is only observed through the latter which is emitted at every time step through the first one.

Representation. Let $\mathbf{S} = (S_1, \dots, S_T)$ denote the **state** latent process, i.e. the discrete Markov chain, with $\forall t \in [\![1,T]\!]$, and $S_t \in \mathscr{S}\ \forall S_t$. $\mathscr{S} = [\![1,K]\!]$ is cardinal K. We recall the parameters of a MC s.a. π a vector of size $1 \times K$, the initial distribution, and the transition matrix A of size $K \times K$.

Let $\mathbf{O} = (O_1, \dots, O_T)$ the **observed** process s.t. $\forall t \in [\![1,T]\!], O_t \in \mathscr{O}$. At each time step t, an observation O_t is emitted conditionally to S_t leading to model the set of K conditional distributions $P(O_t|S_t = k) \equiv b_k(O_t)$. If O_t is discrete, we note $\mathscr{O} = \{v_1, \dots, v_G\}$ and the CPD takes the form of a $K \times G$ matrix, denoted B, s.t. $b_k(j) \equiv P(O_t = j|S_t = k)$. If O_t is continuous $\forall k, O_t|S_t = k$ is a probability density function s.t. $b_k(o_t) \equiv f(o_t|S_t = k)$.

Combining all the CPDs, the JPD of a HMM writes as follows:

$$P(\mathbf{O},\mathbf{S}) = P_\pi(S_1) \prod_{t=1}^{T} P_B(O_t|S_t) \prod_{t=2}^{T} P_A(S_t|S_{t-1}). \tag{1.16}$$

We provide the graphical representation corresponding to the equation of the JPD (1.16) in figure 1.1. A node that is filled corresponds to an observed random variable while a non filled one corresponds to a hidden random variable.

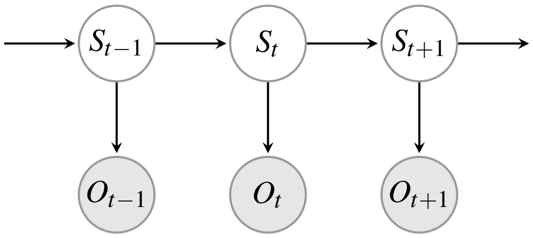


Figure 1.1: Graphical model corresponding to a 1st order HMM

Learning. Since S_t is hidden and discrete, the likelihood of the observed data can be obtained by summing over all possible of $\mathbf{S}$ at each time S_t:

$$P_\theta(\mathbf{O}) = \sum_{k_1 \in \mathscr{S}} \dots \sum_{k_T \in \mathscr{S}} P_\theta(\mathbf{O}, S_1 = k_1, \dots, S_T = k_T)$$

However, this results in K^T operations Rabiner (1989) since it has to test all the combinations of hidden states. Hence we use the EM algorithm based on a representation of the JPD to maximize a lower bound of the log-likelihood as stated in equation (1.14).

The JPD, equation (1.16), can be rewritten in terms of the parameters $\theta = \{\pi, A, B\}$:

$$P_\theta(\mathbf{O}, \mathbf{S}) = \prod_{k=1}^{K} \pi_k^{\mathbb{1}(o_1=k)} \prod_{t=1}^{T} \prod_{k=1}^{K} \prod_{v_g \in \mathscr{O}}^{M} b_k(m)^{\mathbb{1}(o_t=v_g, s_t=k)} \prod_{t=2}^{T} \prod_{k=1}^{K} \prod_{k'=1}^{K} A_{kk'}^{\mathbb{1}(s_t=k', s_{t-1}=k)}.$$

We then build the associated Q-function, equation (1.15), by taking the expected value of complete-data w.r.t. the posterior of the hidden variables:

$$\begin{aligned} \mathscr{Q}(\theta, \theta^{old}) = & \sum_{k=1}^{K} P_{\theta^{(old)}}(S_1 = k|\mathbf{O}) \log \pi_k + \sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{v_g \in \mathscr{O}} P_{\theta^{(old)}}(S_t = k|\mathbf{O}) \log b_k(m) \\ & + \sum_{t=2}^{T} \sum_{k=1}^{K} \sum_{k'=1}^{K} P_{\theta^{(old)}}(S_t = k', S_{t-1} = k|\mathbf{O}) \log A_{kk'}. \end{aligned} \tag{1.17}$$

noting that the expectation of a binary random variable is just the probability that it takes 1, i.e. $\mathbb{E}(\mathbb{1}(o_t = v_g, s_t = k)|\mathbf{O}) = P(S_t = k|\mathbf{O})\mathbb{1}\{o_t = v_g\}$ and so $\sum_{v_g \in \mathscr{O}} \mathbb{E}[\mathbb{1}\{o_t = v_g, s_t = k\}|\mathbf{O}] = P(S_t = k|\mathbf{O})$

The Q-function highlights the quantities to be estimated in the E-step:

$$P_{\theta^{old}}(S_t = k|\mathbf{O}), \tag{1.18}$$

$$P_{\theta^{old}}(S_t = k', S_{t-1} = k|\mathbf{O}), \tag{1.19}$$

while the M-step optimizes the Q-function w.r.t. θ by computing the partial derivatives using the Lagrangian in a similar fashion to the MLE of the Multinomial distribution, equation (1.2):

$$\pi_k = \frac{P_{\theta^{old}}(S_1 = k|\mathbf{O})}{\sum_{k' \in \mathscr{S}} P_{\theta^{old}}(S_1 = k'|\mathbf{O})},$$

$$A_{kk'} = \frac{\sum_{t=2}^{T} P(S_t = k', S_{t-1} = k|\mathbf{O})}{\sum_{k'' \in \mathscr{S}} \sum_{t=2}^{T} P(S_t = k'', S_{t-1} = k|\mathbf{O})},$$

and similarly for the emission distribution

$$b_k(j) = \frac{\sum_{t=1}^{T} P_{\theta^{old}}(S_t = k|\mathbf{O})\mathbb{1}(O_t = j)}{\sum_{t=1}^{T} P_{\theta^{old}}(S_t = k|\mathbf{O})}.$$

Inference. State inference is used in E-step to compute both the expected sufficient statistics given by the equations (1.18) and (1.19). The original idea of the Forward-Backward algorithm, which performs exact inference, is the standard inference procedure used in HMM and is simply an application of the message passing algorithm.

The idea is to break one of the equation into two pieces using the Bayes theorem and conditional independence:

$$\begin{aligned} P(S_t = k|\mathbf{O}) &\propto P(S_t = k, O_1, ..., O_t)P(O_{t+1}, ..., O_T|S_t = k) \\ &= \alpha_t(k)\beta_t(k) \end{aligned} \tag{1.20}$$

where both $\alpha_t(k)$ and $\beta_t(k)$ are computed by induction:

$$\begin{aligned} \alpha_t(k) &= \sum_{k' \in \mathscr{S}} P(S_t = k, S_{t-1} = k', O_1, ..., O_t) \\ &= \sum_{k' \in \mathscr{S}} P(O_t|S_t = k)P(S_t = k|S_{t-1} = k')P(S_{t-1} = k', O_1, ..., O_{t-1}) \\ &= \sum_{k' \in \mathscr{S}} b_k(o_t)A_{k'k}\alpha_{t-1}(k'), \end{aligned}$$

with

$$\begin{aligned} \alpha_1(k) &= P(S_1 = k, O_1 = o_1) \\ &= P(O_1 = o_1|S_1 = k)P(S_1 = k) \\ &= b_k(o_1)\pi_k. \end{aligned}$$

Note that we omit the denominator since it is just a way to normalize the distributions so that they sum to 1 and can easily be computed by summing over S_t. It should also be noticed that $\alpha_t(k) \propto P(S_t = k|O_1, ..., O_t)$ corresponds to what we defined to be the

filtered probabilities. Moreover:

$$\begin{aligned}\beta_t(k) &= \sum_{k' \in \mathscr{S}} P(S_{t+1}=k', O_{t+1},...,O_T|S_t=k)\\ &= \sum_{k' \in \mathscr{S}} P(O_{t+1}|S_{t+1}=k')P(S_{t+1}=k'|S_t=k)P(O_{t+2},...,O_T|S_{t+1}=k')\\ &= \sum_{k' \in \mathscr{S}} b_{k'}(o_{t+1})A_{kk'}\beta_{t+1}(k'),\end{aligned}$$

with $\beta_T(k)=1, \forall k \in \mathscr{S}$. These quantities are again reused for the computation of the second expected sufficient statistic given by equation (1.19):

$$\begin{aligned}P(S_t=k', S_{t-1}=k|\mathbf{O}) &\propto P(\mathbf{O}, S_t=k', S_{t-1}=k)\\ &= P(O_{t+1},...,O_T|S_t=k')P(O_t|S_t=k')P(O_1,...,O_{t-1},S_{t-1}=k)P(S_t=k'|S_{t-1}=k)\\ &= \beta_t(k')b'_k(o_t)\alpha_{t-1}(k)A_{kk'}.\end{aligned} \tag{1.21}$$

In pratice, $\alpha_t(k)$ and $\beta_t(k)$ are different at every EM iteration, they are intermediate quantities which should be stored in memory for each iteration since they are reused several times in inference. So are both the expected sufficient statistics. In the literature, equation (1.18) is often referred as the $\gamma_t(k)$ variables while (1.19) is referred as $\xi_t(k,k')$ variables. We can also note that the $\gamma_t(k)$, equation (1.20), corresponds to the **smoothed probabilities** whereas $\xi_t(k,k')$ is called the **double smoothed probabilities**.

With known parameters, we can also perform a **prediction** using the same recursive inference technique:

$$\begin{aligned}P(O_{T+\tau}|O_{1:T}) &= \sum_{S_{T+\tau}} P(O_{T+\tau}|S_{T+\tau})P(S_{T+\tau}|O_{1:T})\\ &= \sum_{S_{T+\tau}} P(O_{T+\tau}|S_{T+\tau}) \sum_{S_{T+1}} ... \sum_{S_{T+\tau-1}} P(S_{T+\tau}|S_{T+\tau-1})...P(S_{T+1}|S_T)P(S_T|O_{1:T})\\ &= \sum_{S_{T+\tau}} b_{s_{T+\tau}}(o_{T+\tau}) \sum_{S_{T+1}} ... \sum_{S_{T+\tau-1}} A_{s_{T+\tau}s_{T+\tau-1}}...A_{s_{T+1}s_T}\alpha_T(s_T).\end{aligned} \tag{1.22}$$

State sequence restoration. Once parameters are learned, the state sequence restoration **S** can be performed to find the "best" state sequence, or the most "optimal" state sequence. There exists several definitions and therefore solutions to this problem. A first solution consists in maximizing the sequence as the marginally most probable states and is called the Maximizer of the posterior marginals (MPM) whereas the second solution is to maximize the most likely state sequence and is called the Maximum A posteriori (MAP). The MPM writes as

$$\mathbf{s}^*_{MPM} = \left(\arg\max_{s_1} p(s_1|\mathbf{o}), \ldots, \arg\max_{s_T} p(s_T|\mathbf{o}) \right) \tag{1.23}$$

where the star upper script (*) stands for the optimal sequence. The MPM, equation (1.23), can be easily computed for s^*_t at each time $t \in [\![1,T]\!]$ by reusing the quantities computed at the last E-step: $s^*_t = \arg\max_{s_t} p(s_t|\mathbf{o})$ by multiplying both forward and backward variables. In the induction procedures of these quantities, the other state variables are summed out and therefore each state is computed by averaging its neighbors. This approach, called **sum-product**, can therefore be seen as a robust one as stated by Marroquin et al. (1987). However, this approach does not take into account the likelihood of the entire optimal path. Even though, each single state is locally maximal, the entire sequence may occur with a probability of 0. The MAP arise as a solution to this problem for which the optimal sequence is:

$$\mathbf{s}^*_{MAP} = \arg\max_{\mathbf{s}} P(\mathbf{s}|\mathbf{o}). \tag{1.24}$$

On the other hand, the MAP is not as straightforwardly computed as the MPM from the forward backward variables considering that it maxes out the other states at each time t s.t.

$$\mathbf{s}^*_t = \arg\max_{s_t} \max_{s_1,\ldots,s_{t-1},s_{t+1},s_T} P(\mathbf{s}|\mathbf{o}), \tag{1.25}$$

deserving its the name of **max-product** procedure. Computing the MAP efficiently involves dynamic programming which keeps a traceback in memory in order to recover the most likely path. In the context of HMM, this algorithm, introduced in 1967, is known as Viterbi's (Viterbi, 1967). First, note that $\arg\max_{\mathbf{s}} p(\mathbf{s}|\mathbf{o}) = \arg\max_{\mathbf{s}} p(\mathbf{s},\mathbf{o})$ because the max over $\mathbf{z}$ does not depend on $p(\mathbf{o})$.

We define the probability of ending up in state s_t at time t given that we took the most probable path:

$$\begin{aligned}
\delta_t(s_t) &\equiv \max_{s_1,\ldots,s_{t-1}} P(s_1,\ldots,s_t,o_1,\ldots,o_t) \\
&= \max_{s_{t-1}} \left(p(o_t|s_t) p(s_t|s_{t-1}) \max_{s_1,\ldots,s_{t-2}} p(s_1,\ldots,s_{t-1},o_1,\ldots,o_{t-1}) \right) \\
&= \max_{s_{t-1}} \delta_{t-1}(s_{t-1}) A_{s_{t-1}s_t} b_k(o_t)
\end{aligned} \tag{1.26}$$

with the initialization

$$\delta_1(s_1) \equiv \max_{s_1} p(s_1, o_1) = \max_{s_1} b_{s_1}(o_1)\pi_{s_1}$$

and the termination

$$\delta_T(s_T) \equiv \max_{s_1,\ldots,s_{T-1}} p(\mathbf{o}, \mathbf{s})$$

which gives us s_T^* s.t.

$$s_T^* = \arg\max_{s_T} \delta_T(s_T).$$

Once we know s_T^*, the main idea is that the most probable path to state s_t at t must be built on the most probable path at time $t-1$ to some other state s_{t-1} followed by a transition from s_{t-1} to s_t. At each time t, we keep the trace associated with $\delta_t(s_t)$

$$a_t(s_t) \equiv \arg\max_{s_{t-1}} \delta_{t-1}(s_{t-1}) A_{s_{t-1}s_t} b_{s_t}(o_t)$$

The most probable state sequence is then computed recursively using the **traceback**:

$$s_t^* = a_{t+1}(s_{t+1}^*).$$

Another alternative to state sequence restoration is called the **N-best list** which is an extension of the Viterbi algorithm and returns the N most likely state sequences. The N-best list was introduced by Schwarz and Chow (1990), and the algorithm's complexity was powered up by Nilsson and Goldberger (2001). Once the N-best state sequences are obtained, one can then use a discriminative method in order to rerank them according to the application. However, the authors state that the algorithm often provides similar results and that N should be very large in order to provide more versatile solutions.

Some authors, such as Foreman (1992); Brushe et al. (1998); Barbu and Zhu (2005); Porway and Zhu (2011); Tu and Zhu (2002), have proposed to use **sampling methods** to provide more versatile solutions. The idea is to sample state sequences from the posterior $p(\mathbf{s}|\mathbf{o})$ by sampling recursively from $s_t^* \sim p(s_t|s_{t-1}^*, \mathbf{o})$ where the quantity is obtained using a forward-backward pass along with another forward pass. After generating multiple optimal sequence, a procedure can be performed in order to check for solution diversity and keep the most relevant. The main drawback of this family of solution is the computational cost as well as the dependency of an ad-hoc procedure for the choice of the most diverse solutions.

More recently, Batra et al. (2012); Kulesza et al. (2012), proposed another family of solutions for the diverse N-best problem, **generalizing the N-best list** algorithm, and consists in optimizing a linear combination of the probability and the dissimilarity of the state sequences.

Guédon (2007)

Applications of HMMs. In pratice, the state sequence $\{S_t\}_{t\in[\![1,T]\!]}$ is unknown and shall be recovered. On the first hand, HMMs are used in the unsupervised case to estimate the density of sequences. On the other hand, it also allows to model long range dependencies between observations mediated via latent variables.

For instance, Obermaier et al. (2001a) used HMM to model multi-channel EEGs where changes in latent states express physiological changes in the spatio-temporal patterns. The main goal of their study was to classify either a subject was imagining turning his head left or right. For a given training set, they computed two HMMs, one for the left turn, another one for the right turn. Finally, they classified the trials from the testing set using the maximal probability of the restored state sequence of each HMM.

Another notable application of HMMs was achieved by Simola et al. (2008) in order to discover reading strategies, the latent states, given eye-movement features which are the observed variables. Plus, the reading strategies have been characterized using model parameters. Moreover, they embedded several HMMs into a discriminative HMM in order to classify the task type that the subjects were performing, showing that task types can be discriminated given eye movement features.

2.6 Various DBNs to overcome HMM's limitations

As we stated before, Hidden Markov Models are the simplest form of Dynamic Bayesian Networks and are usually either used for recovering and characterizing the latent state structure of sequential data, or used for forecasting with long term dependencies. However, sometimes the data is not fitted for HMMs. For example, the phenomenon modeled by the latent structure may not have a geometric sojourn state distribution which is the case in HMM as we showed in Lemma (1). To overcome this aspect, **Hidden semi-Markov Models** (HSMMs) have been introduced. Its goal is to relax the state sojourn duration hypothesis. Since HSMMs are core to this thesis, they are discussed in much more detail in section 3. Another instance is the **Hierarchical HMM**

(HHMM) proposed by Fine et al. (1998), which is suited for complex sequential data with multi-scale structure such as the natural language which can be decomposed at the sentence, word and syllable levels. Another well known instance is the **Factorial HMM** (FHMM), Ghahramani and Jordan (1996), which answer the need of distributed state representation in HMMs by decoupling the dynamics of a single process or multiple independent processes generating multiple time series. Smyth (1997) also introduced **mixtures of HMMs** in a same framework in order to cluster sequences. Bellow, we discuss several methods which are of high interest in the context of this thesis.

Coupled signals. A first DBN called **Coupled HMM** (CHMM) for coupling related data streams was developed by Brand (1997). In such a model, each observed sequence has its own Markov chain and each of them interact with its neighbors. Assume the hidden state is composite of C different channels s.t. $\mathbf{S}_t = \{S_t^{(1)}, ..., S_t^{(C)}\}$, the assumption on the CPD of the hidden states is as follows:

$$P(\mathbf{S}_t|\mathbf{S}_{t-1}) = \prod_{c=1}^{C} P(S_t^{(c)}|\mathbf{S}_{pa(t-1,c)}) \tag{1.27}$$

where $pa(.)$ is the parent function and $\mathbf{S}_{pa(t-1,c)}$ denotes the parents of $S_t^{(c)}$ at $t-1$ which should represent the neighborhood or a spacial dependency between the channels.

CHMM have been successfully applied in diverse areas showing significant improvement compared to other classes of HMM. Kwon and Murphy (2000); Murphy and Russell (2002) applied CHMM to freeway traffic modeling. They had multiple detectors recording the car speed at different locations which was their observations. Each of this sequence of observations had an underlying hidden Markov chain where the state was representing a Boolean of either it is jammed or not. Each Markov chain was then coupled to its spatial neighbor using equation (1.27). In a classification framework, Brand et al. (1997) used CHMM to model human activity recognition where the observations were the tracking data of different limbs. A CHMM was learnt on a training set for each kind of activities while the performance was evaluated on a testing set for which they used the Viterbi algorithm to find the maximum likelihood model and classify the activity accordingly. Nefian et al. (2002) also applied various DBNs to speech recognition by jointly modeling audio and video. They showed that CHMM outperformed most of the other models, especially when the noise was low. They also mentioned the CHMM was still efficient even though the signals were asynchronous.

However, CHMM still suffers from high parameter specification especially if one wants to enlarge the interactions between the neighborhood of each channel. Asavathiratham (2001); Zhong and Ghosh (2001) proposed a variant called **influence model**, or **distance coupled HMM** (DCHMM), which uses fewer parameters and has the following assumption on the hidden states CPD:

$$P(S_t^{(c)}|\mathbf{S}_{t-1}) = \sum_{c'=1}^{C} w_{c',c} P(S_t^{(c)}|S_{t-1}^{(c')}) \tag{1.28}$$

where $w_{c',c}$ represent the coupling weight between model c' and c s.t. $\sum_{c'=1}^{C} w_{c'c} = 1$, describing how much $S_{t-1}^{(c')}$ affects the distribution of $S_t^{(c)}$. It acts as an approximation of the joint dependency by linear combination of all the marginal dependencies. Also note that in case there is a spacial dependency between the channels, we can make each $w_{c',c}$ function of distance between channels. The influence model therefore has $C^2 + CK^2$ transition parameters while the standard CHMM has K^C transition parameters at worst, i.e. in the fully coupled case.

Influence model has been applied by Basu et al. (2001) to quantify, through coupling parameters, human interaction in conversational settings. Zhong and Ghosh (2002) also applied the distance coupled HMM to classify if subject had genetic predisposition to alcoholism or not given EEG data. A DCHMM was learned for each type of patient on a training set and the classification performance was evaluated on a testing set. Surprisingly DCHMM performed much worst than standard HMM on this task. The authors pleaded for an insufficient amount of channels and not good enough approximate inference for the DCHMM to perform well.

Asynchronous signals. The most generic instance of DBN in the literature built to handle asynchronous signals of different nature describing the same event is the **asynchronous Hidden Markov Model** (AHMM) introduced by Bengio (2003) along with an application to audio-visual speech recognition. Given two streams represented by a series of random variables that might be of different length $\{O_t^{(1)}\}_{t\in[\![1,T]\!]}$ and $\{O_t^{(2)}\}_{t\in[\![1,T']\!]}$ respectively, with $T' \leq T$, the main difference compared to the standard HMM with two output processes lies in the introduction of a new set of random variable $\{D_t\}_{t\in[\![1,T]\!]}$ which represent the probability of emitting $O_t^{(2)}$ at time t and can be seen as the alignment between both the signals. This leads to the introduction of a new set of parameters in the model:

$$\varepsilon_t(k,t') \equiv P(D_t = t' | D_{t-1} = t' - 1, S_t = k, O^{(1)}_{1:t}, O^{(2)}_{1:t'}),$$

which means that the alignment at t depends on the alignment at $t-1$ but also from the current hidden state as well as the previous observations from both the sequences. Several assumptions can be made on this CPD. For example, if $\varepsilon_t(k,t') \equiv P(D_t = t' | S_t = k)$, the widely used **pair HMM** in DNA sequences alignment (Durbin et al., 1998) can be recovered. This instance works well with categorical variable. With continuous data streams, a more common assumption used Bengio (2004), is $\varepsilon_t(k,t') \equiv P(D_t = t' | D_{t-1} = t' - 1, S_t = k)$ and is simply modeled by a Binomial distribution.

Another series of similar model have been introduced namely **Input Output HMM** (Bengio and Frasconi, 1995) and **Asynchronous Input Output HMM** (Bengio and Bengio, 1996) which are similar to AHMM except that the arcs direction have been reversed between $\{O^{(1)}_t\}_{t \in [\![1,T]\!]}$ and $\{D_t\}_{t \in [\![1,T]\!]}$. The first one is then called the control signal. It is naturally more discriminant and performs better from real time predictions tasks of the output signal given the input one. It also allows the dynamics of the latent Markov chain to evolve since it is conditioned by the input signal and is therefore better suited for non homogeneous Markov chains, i.e. for long term predictions. There have been several domain of application of these models such as speech recognition (Bengio and Frasconi, 1996; Bengio, 1999), finance (Bengio et al., 2001) or human authentication (Chiappa and Bengio, 2003).

3 Hidden semi-Markov Models

Introduced in the 1980 by Ferguson (1980), the Hidden semi-Markov Model (HSMM) has, since then, widely been studied as an extension of the Hidden Markov Model (HMM), notably by Guedon and Cocozza-Thivent (1990); Guédon (1999, 2003) for developing fast and real-life-oriented inference algorithms, by Yu and Kobayashi (2003, 2006); Yu (2010, 2015) for contributions in inference algorithms and for a state of the art, and finally by Murphy (2002); Murphy and Russell (2002) for proposing a clear alternative formulation of the problem. Barbu and Limnios (2009) proposed a book to treat the subject with its use in DNA analysis. Another book (Yu, 2015) is more algorithmic and implementation oriented.

Similarly to HMM, an HSMM is composed of two stochastic processes. The former is a finite-state homogeneous semi-Markov chain (SMC) which is latent, while it influences the latter which produces observations. A SMC is like a Markov Chain (MC) except that the within-state sojourn time is not necessarily geometric and can therefore be of any form (tabular or parametric).

Therefore, on top of the traditional parameters involved in HMM, i.e. initial, transition, and emission probabilities, a HSMM is also described by within-state sojourn duration, also called dwell times.

3.1 General definition

Let us assume the following notations :

- the set of hidden states $\mathscr{S} = [\![1, K]\!]$ where S_t is the state at time t. $S_{1:T}$ is the hidden state sequence, $s_{1:T}$ is the realization associated to the hidden state sequence
- the random state duration d is either bounded $\in [\![1, D]\!]$ or set to $d \in \mathbb{N}$ and naturally upper bounded by the length of the sequence,
- $S_{t_1:t_2} = k$ means staying in state k from time t_1 to t_2 without any constraints on S_{t_1-1} and S_{t_2+1}
- $S_{[t_1:t_2]} = k$ means staying in state k from time t_1 to t_2 with the constraints that $S_{t_1-1} \neq k$ and $S_{t_2+1} \neq k$
- $S_{[t_1:t_2} = k$ means staying in state k from time t_1 to t_2 with the constraint $S_{t_1-1} \neq k$
- $S_{t_1:t_2]} = k$ means staying in state k from time t_1 to t_2 with the constraint $S_{t_2+1} \neq k$
- the set of observable values $\mathscr{O} = \{v_1, ..., v_G\}$ where $O_t \in \mathscr{O}$ is the observed variable at time t. $O_{1:T}$ is the observed state sequence, $o_{1:T}$ is the realization associated to the observed state sequence

The most general HSMM model assumes the following set of parameters $\theta \equiv \{a_{(k,d')(k',d)}, b_{k',d}(v_{k_1:k_d}), \pi_{k,d}\}$ such that:

- the state transition probability from (k,d') to (k',d), $k \neq k'$: $a_{(k,d')(k',d)} \equiv P(S_{[t+1:t+d]} = k' \mid S_{[t-d'+1:t]} = k)$
- the emission probability $b_{k',d}(o_{t+1}) \equiv P(o_{t+1:t+d} \mid S_{t+1:t+d} = k')$
- the initial distribution : $\pi_{k',d} \equiv P(S_{[1:d+1]} = k')$

3.2 Relaxing the main hypothesis

Sojourn time assumptions. Based on this general model, several simplifying assumptions models have been suggested in the literature regarding the sojourn time:

- **Marhasev** (Marhasev et al., 2006) $d \perp d', k'$. The transition probabilities are expressed as $a_{(k,d')(k',d)} = a_{(k,d')k'}\ p_{k'}(d)$ such that $a_{(k,d')k'} \equiv P[S_{[t+1} = k' \mid S_{[t-d'+1:t]} = k])$ and $p_{k'}(d) \equiv P[S_{t+1:t+d} = k' \mid S_{[t+1} = k'$ is the probability of the duration d of the state k'.

- **Residential time HMM** (Yu and Kobayashi, 2003) the state transition is independent to the duration of the previous step : $a_{(k,d')(k',d)} = a_{k(k',d)}$ such that $a_{k(k',d)} = P[S_{[t+1:t+d]=k'} \mid S_{t]} = k]$.

- **Variable transition HMM** (Vaseghi, 1991, 1995) : the self-transition is allowed and independent to the previous step : $a_{(k,d')(k',d)} = a_{k(k',d)} = a_{(k,d')k'} \prod_{\mathscr{S}=1}^{d-1} a_{k'k'(\mathscr{S})}[1 - a_{k'k'(d)}]$ where $a_{k'k'(d)} \equiv P[S_{t+d+1} = k' \mid S_{[t-d'+1:t]} = k,\ S_{[t+1:t+d]=k'}] = P[S_{t+d+1} = k' \mid S_{[t+1:t+d]=k'}]$ is the self-transition probability when state k' has lasted for d time units. $1 - a_{k'k'(d)} = P[S_{t+d]} = k' \mid S_{[t+1:t+d=k']}]$ is the probability that state k' ends with duration d.

- **Explicit duration HMM** (Ferguson, 1980; Mitchell and Jamieson, 1993; Sin and Kim, 1995), transition to the current state is independent to the duration of the previous state and the duration is only conditioned by the current state: $a_{(k,d')(k',d)} = a_{kk'} p_{k'}(d)$ where $a_{kk'} \equiv P[S_{[t+1} = k' \mid S_{t]} = k]$.

Then $p_{k'}(d)$ can either be multinomial (nonparametric) or take any (parametric) discrete distribution. See section 1.3.3 for more details.

$b_{k',d}(v_{k_1:k_d})$ can also be parametric or non-parametric, discrete, continuous, dependent, or independent on the state durations. It can also be a mixture of distributions.

State sequence censoring. There exist several assumptions concerning the survival of the semi-Markov chain.

Assumption 3. *The* ***general assumption*** *supposes that the process starts at $-\infty$ and ends at $+\infty$ even though the observations are done from time 1 to T. An inference procedure in this case is described in Yu (2010).*

Assumption 4. *The **simplifying assumption** assumes that the process starts at* 1 *and finishes at* T*. Most of the literature uses this assumption since it makes inference more straightforward. We subsequently describe this procedure.*

Assumption 5. *The **right-censored** hypothesis can be useful in many real life applications. It assumes that the process started at time 1 and ends at time* $+\infty$*. This has been studied by Guédon (2003), who developed the corresponding inference procedures.*

3.3 Representation of EDHMM

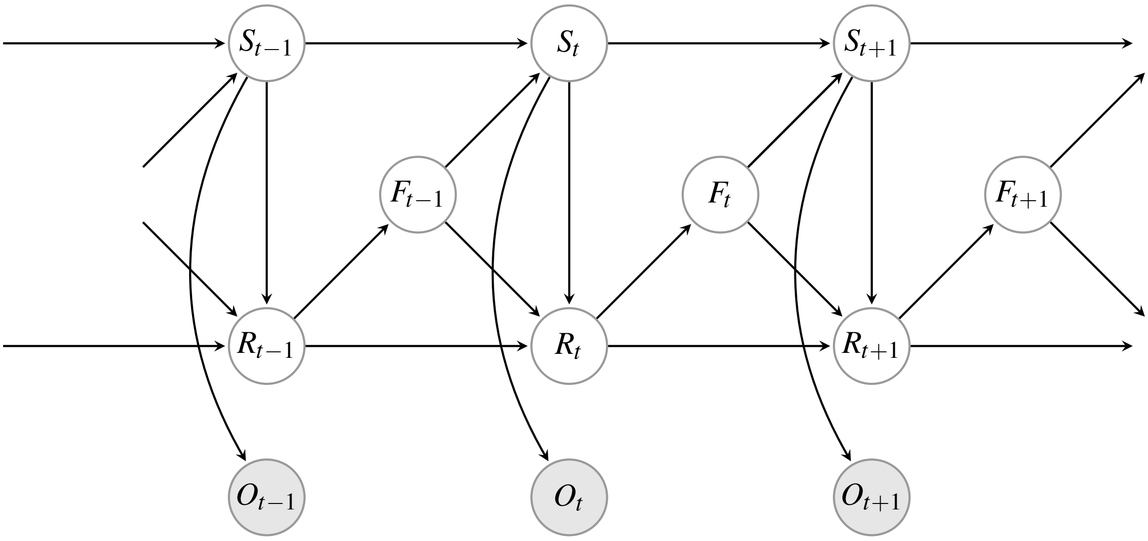


Figure 1.2: Graphical model corresponding to a EDHMM

Using the formalism proposed in Murphy (2002) which is Dynamic Bayesian Network-oriented (DBN), we describe a SMC by:

- $S_{1:T}, \forall t \in \{1,...,T\}$ $S_t \in \mathscr{S} = \{1,...,K\}$, the discrete and latent process. Note that $S_{1:T}$ stands for $\{S_t\}_{t=1}^{T}$.

- $R_{1:T}, \forall t \in \{1,...,T\}$ $R_t \in \{1,...,\mathscr{D}\}$, a discrete and latent process, encoding the residual time R_t in the current state S_t at time t. At the beginning of a new state, a new duration is randomly sampled from an arbitrary distribution p_{s_t} and then counts down deterministically to 1, and so on.

- $F_{1:T}, \forall t \in \{1,...,T\}$ $F_t \in \{0,1\}$, a discrete and latent process, which acts as a binary switch which is turned on when $R_{t-1} = 1$ and off else. Even though it is redundant with $R_{1:T}$, it is useful in order to simplify later notations and equations. Initialization is performed s.t. $F_0 = 1$ and $R_0 = 1$, and we also have $F_T = 1$ which means that the process starts at time 1 and will end at time T and is related to the simplifying assumption 4

Follows the Conditional Probability Distributions (CPD) parameters associated to a SMC :

$$P(S_1 = k) = \pi_k$$

with $\pi \in \mathscr{S}$, a vector representing all the initial probabilities,

$$P(S_t = k | S_{t-1} = k', F_{t-1} = f) = \begin{cases} \mathbb{1}\{k = k'\} & \text{if } f = 0 \\ A_{k'k} & \text{if } f = 1 \end{cases}$$

with $\mathbb{1}$ the indicator function, and $A \in \mathscr{S} \times \mathscr{S}$, a matrix representing the transition probabilities. We also have,

$$P(R_t = d | R_{t-1} = d', S_t = k, F_{t-1} = 1) = p_k(d)$$

with $p_k(d')$ being an arbitrary probability distribution on $\mathbb{N}^*$, representing the sojourn distributions for each state k while entering a new state at time t and then sampling a new value $d' \geq 1$ for R_t. Finally,

$$P(R_t = d | R_{t-1} = d', S_t = k, F_{t-1} = 0) = \begin{cases} \mathbb{1}\{d = d' - 1\} & \text{if } d > 1 \\ \text{undefined} & \text{if } d = 1 \end{cases}$$

and,

$$P(F_t = f | R_t = d) = \begin{cases} \mathbb{1}\{d = 1\} & \text{if } f = 1 \\ \mathbb{1}\{d > 1\} & \text{if } f = 0 \end{cases}$$

define the countdown process, i.e. the residual time in the current state.

In conclusion, the process can be described as the transition from a latent state to another at time t triggered the following changes: the finishing node switches on $F_{t-1} = 1$, requiring a transition to a new state, $S_t = k$, from the previous one, $S_{t-1} = k'$, with $k \neq k$. Finally, given this state k, a new sojourn duration is sampled, $R_t \sim p_k \geq 1$.

Discrete Observed Process. The observed process can be discrete and is emitted from $S_{1:T}$:

- $O_{1:T}, \forall t \in \{1, ..., T\}$ $O_t \in \mathscr{O} = \{v_1, ..., v_G\}$.

and the associated CPD is as follows:

$$P(O_t = v_g | S_t = k) = b_k(v_g)$$

where $b_k(v_g)$ can either represent a tabular distribution and be a matrix of size $K \times G$, or K parametric distributions.

Continuous Observed Process. The observed process can also be continous, Gaussian for example, and described by the CPD:

$$P(O_t | S_t = k) = \mathscr{N}(\mu_k, \Sigma_k)$$

μ_k being the mean vector of size K and Σ_k, the covariance matrix of size $K \times K$.

Joint Probability Distribution. Combining all the CPDs, we can define the following Joint Probability Distribution with one single discrete output process:

$$\begin{aligned}
&P(\{S_t, O_t, R_t, F_t\}_{t=1}^{\mathscr{S}}; \theta = \{\pi_{k'}, a_{ik'}, b'_k(v_k), p_{k'}(d)\}) \\
&= P(S_1) \prod_{t=2}^{T} P(S_t | S_{t-1}, F_{t-1}) \prod_{t=1}^{T} P(O_t | S_t) P(R_t | S_t, R_{t-1}, F_{t-1}) P(F_t | R_t) \\
&= \prod_{k=1}^{K} \pi_k^{\mathbb{1}\{s_1 = k\}} \prod_{t=2}^{T} \prod_{k=1}^{K} \prod_{k'=1}^{K} \left(\mathbb{1}\{k = k'\}^{\mathbb{1}\{s_t = k, s_{t-1} = k', f_{t-1} = 0\}} A_{k'k}^{\mathbb{1}\{s_t = k, s_{t-1} = k', f_{t-1} = 1\}} \right) \\
&\quad \prod_{t=1}^{T} \prod_{k=1}^{K} \left(\prod_{v_g \in \mathscr{O}} b_k(v_g)^{\mathbb{1}\{o_t = v_g, s_t = k\}} \prod_{d=1}^{\mathscr{D}} \prod_{d'=1}^{\mathscr{D}} \left\{ \mathbb{1}\{d = d' - 1\}^{\mathbb{1}\{r_t = d, s_t = k, r_{t-1} = d', f_{t-1} = 0\}} \right. \right. \\
&\qquad p_k(d)^{\mathbb{1}\{r_t = d, s_t = k, r_{t-1} = d', f_{t-1} = 1\}} \\
&\qquad \left. \left. \mathbb{1}\{d > 1\}^{\mathbb{1}\{f_t = 0, r_t = d\}} \mathbb{1}\{d = 1\}^{\mathbb{1}\{f_t = 1, r_t = d\}} \right\} \right)
\end{aligned}$$

3.4 Inference and learning

In order to apply EM, we compute $\mathscr{Q}(\theta, \theta^{old})$,

$$
\begin{aligned}
\mathscr{Q}(\theta, \theta^{old}) &= \mathbb{E}[\log P(\{S_t, R_t, F_t, O_t\}_{t=1}^{T}; \theta) | \{O_t\}_{t=1}^{T}, \theta^{old}] \\
&= \sum_{k=1}^{K} P(S_1 = k | \{O_t\}; \theta^{old}) \log \pi_k \\
&+ \sum_{t=2}^{T} \sum_{k=1}^{K} \sum_{k=1}^{K} \Bigg(P(S_t = k, S_{t-1} = k', F_{t-1} = 0 | \{O_t\}; \theta^{old}) \log \mathbb{1}\{k = k'\} \\
&\qquad + P(S_t = k, S_{t-1} = k', F_{t-1} = 1 | \{O_t\}; \theta^{old}) \log A_{k'k} \Bigg) \\
&+ \sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{v_g \in \mathscr{O}} \Bigg[P(S_t = k | \{O_t\}; \theta^{old}) \log b_k(v_g) \\
&\qquad + \sum_{d=1}^{\mathscr{D}} \sum_{d'=1}^{\mathscr{D}} \Bigg(P(R_t = d, S_t = k, R_{t-1} = d', F_{t-1} = 0 | \{O_t\}; \theta^{old}) \log \mathbb{1}\{d = d' - 1\} \\
&\qquad\qquad + P(F_t = 0, R_t = d | \{O_t\}; \theta^{old}) \log \mathbb{1}\{d > 0\} \\
&\qquad\qquad + P(R_t = d, S_t = k, R_{t-1} = d', F_{t-1} = 1 | \{O_t\}; \theta^{old}) \log p_{k'}(d) \\
&\qquad\qquad + P(F_t = 1, R_t = d | \{O_t\}; \theta^{old}) \log \mathbb{1}\{d = 0\} \Bigg) \Bigg]
\end{aligned}
\tag{1.29}
$$

which highlights the Expected Sufficient Statistics (ESS) to be evaluated in the E-step:

$$P(S_t = k | \mathbf{O}; \theta^{old}), \tag{1.30}$$

$$P(S_t = k, S_{t-1} = k', F_{t-1} = 1 | \mathbf{O}; \theta^{old}), \tag{1.31}$$

$$P(R_t = d, S_t = k, R_{t-1} = d', F_{t-1} = 1 | \mathbf{O}; \theta^{old}), \tag{1.32}$$

A core challenge related to dynamic Bayesian networks is the computation of the posteriors in the E-step since variables are not i.i.d. For this purpose we need a to use inference algorithms. More particularly, for exact inference, we use an algorithm called Forward-Backward.

First, we define the following intermediate probabilities:

$$P(S_t = k, F_t = 1, \{O_t\})$$

A naive application of the message passing algorithm Koller et al. (2009) would consist in computing:

$$\begin{aligned}
\alpha_t(k', d, f) &= P(S_t = k, R_t = d, F_t = f, O_{1:t}) \\
&= \sum_{k=1}^{K} \sum_{d'=1}^{\mathscr{D}} \sum_{f'=0}^{1} P(S_t = k, R_t = d, F_t = 1, S_{t-1} = k', R_{t-1} = d', F_{t-1} = f', O_{1:t}) \\
&= \sum_{k=1}^{K} \sum_{d'=1}^{\mathscr{D}} \sum_{f'=0}^{1} P(O_t|S_t = k) P(S_t = k|S_{t-1} = k', F_t = f) P(R_t = d|S_t = k, R_{t-1} = d', F_t = f) \\
&\qquad P(F_t = f|R_t = d') P(o_{1:t-1}, R_{t-1} = d', F_{t-1} = f', S_{t-1} = k') \\
&= \sum_{k=1}^{K} \sum_{d'=1}^{\mathscr{D}} P(O_t|S_t = k) \alpha_{t-1}(k', d', f') \\
&\qquad \left(\mathbb{1}\{k = k'\} \mathbb{1}\{d = d' - 1\} \mathbb{1}\{d' > 0\} + A_{k'k} p_k(d) \mathbb{1}\{d' = 0\} \right)
\end{aligned} \tag{1.33}$$

and then marginalizing out d and f:

$$P(S_t = k, F_t = 1, O_{1:t}) = \alpha_t(k) = \sum_{d=1}^{\mathscr{D}} \sum_{f=0}^{1} \alpha_t(k, d, f)$$

which has complexity $O((TK\mathscr{D})^2)$. Though, one can intuitively see that the recursion requires way more computations than it should since most of the probabilities are modeled as indicators. Hence, we formalize the intuitions proposed in Mitchell et al. (1995); Murphy (2002); Guédon (2003) and define $V_t = \max_{t'}\{t' < t | S_{t'} \neq S_t\}$, the previous transition instant. By convention, $V_t = 0$ if $\{t' < t | S_{t'} \neq S_t\} = \emptyset$, and if $V_t > 1$, a transition has already occured. This definition is particularly useful when $F_t = 1$ and $V_t = t'$ because it implies $F_{t'-1} = 1$, $R_{t'} = t - t'$ and therefore that $S_{t':t}$ is constant for duration $t - t'$. The forward variables are computed the following way:

$$\begin{aligned}
\alpha_t(k) &= P(S_t = k, F_t = 1, O_{1:t}) \\
&= \sum_{t'=0}^{t-1} P(V_t = t', S_t = k, F_t = 1, O_{1:t}) \\
&= P(V_t = 0, S_t = k, F_t = 1, O_{1:t}) + \sum_{t'=1}^{t-1} P(V_t = t', S_t = k, F_t = 1, O_{t-t':t}, O_{1:t-t'-1}) \\
&= P(R_1 = t, S_1 = k, F_t = 1, O_{1:t}) + \sum_{t'=1}^{t-1} P(R_{t'} = t - t', F_{t'-1} = 1, S_{t'} = k, F_t = 1, O_{t-t':t}, O_{1:t-t'-1}) \\
&= P(O_{1:t} | S_1 = k, R_1 = t, F_t = 1) P(R_1 = t | S_1 = k) P(S_1 = k) \\
&\quad + \sum_{t'=1}^{t-1} \left(P(O_{t-t':t} | S_{t'} = k, R_{t'} = t - t') P(R_{t'} = t - t' | S_{t'} = k, F_{t'-1=1}) P(S_{t'} = k, F_{t'-1} = 1, O_{1:t-t'}) \right) \\
&= \pi_k p_k(t) \prod_{u=1}^{t} b_k(O_u) + \sum_{t'=1}^{t-1} \left(p_k(t-t') \alpha^*_{t-t'}(k) \prod_{u=t'}^{t} b_k(O_u) \right)
\end{aligned} \tag{1.34}$$

with,

$$\begin{aligned}
\alpha^*_t(k) &= P(S_{t+1} = k, F_t = 1, O_{1:t}) \\
&= \sum_{k'=1}^{K} \alpha_t(i) A_{k'k}.
\end{aligned} \tag{1.35}$$

This method simply relies on the computation by induction of two sets of forward variables around transition instants, increasing storage space by 2 compared to standard HMM. The complexity of this forward recursion is $O(TK^2\mathscr{D})$.

Following a similar schema for the backward variables, we firstly have:

$$\begin{aligned}
\beta_t(k) &= P(O_{t+1:T} | S_t = k, F_t = 1) \\
&= \sum_{k'=1}^{K} \beta^*_t(k') A_{k'k},
\end{aligned} \tag{1.36}$$

we also define $W_t = \max_{t'}\{t' > t | S_{t'} \neq S_t\}$, the next transition instant, implying that if $F_t = 1$ and $W_t = t'$, then $R_t = t' - t$ (and therefore $R_{t+t'-1} = 1$), $F_{t'-1} = 1$, $S_{t:t'-1}$ constant for duration $t' - t$. If $\exists t$ s.t. $W_t > T$, then this is the right-censored sojourn time assumption 5. Comes:

$$\begin{aligned}
\beta_t^*(k) &= P(O_{t+1:T}|S_{t+1}=k,F_t=1)\\
&= \sum_{t'=t+2}^{T} P(O_{t+1:t'},O_{t'+1:T},W_t=t'|S_{t+1}=k,F_t=1)\\
&= \sum_{t'=t+2}^{T} P(O_{t+1:t'},O_{t'+1:T},R_{t+1}=t'-t,F_{t'}=1|S_{t+1}=k,F_t=1)\\
&= \sum_{t'=t+2}^{T} \Big(P(O_{t+1:t'}|R_{t+1}=t'-t,S_{t+1}=k)P(O_{t'+1:T}|S_{t'-1}=k,F_{t'-1}=1,R_{t+1}=t'-t)\\
&\qquad P(F_{t'}=1|R_{t+1}=t'-t)P(R_{t+1}=t'-t|S_{t+1}=k,F_t=1)\Big)\\
&= \sum_{t'=t+2}^{T} \left(\prod_{u=t+1}^{t+d} b_k(o_u)\beta_{t'-t}(k)p_k(t'-t) \right).
\end{aligned} \tag{1.37}$$

The computation of ESS (1.31) can then easily be derived:

$$\begin{aligned}
P(S_t=k,S_{t-1}=k',F_{t-1}=1|O_{1:T}) &\propto P(S_t=k,S_{t-1}=k',F_{t-1}=1,O_{1:T})\\
&= P(S_t=k,S_{t-1}=k',F_{t-1}=1,O_{1:t-1},o_{t:T})\\
&= P(O_{t:T}|S_t=k,F_{t-1}=1)P(S_t=k|S_{t-1}=k',F_{t-1}=1)\\
&\quad P(O_{1:t-1},S_{t-1}=k',F_{t-1}=1)\\
&= \beta_{t-1}^*(k)A_{k'k}\alpha_{t-1}(k')
\end{aligned} \tag{1.38}$$

where the normalization term $P(O_{1:T})$, which is ommited here, can easily be computed so that the probabilities sum to one. The computation of ESS (1.32) is calculated in a similar manner:

$$\begin{aligned}
P(R_t=d,S_t=k,F_{t-1}=1|O_{1:T}) &\propto P(R_t=d,S_t=k,F_{t-1}=1,O_{1:T})\\
&= P(O_{t:T}|S_t=k,F_{t-1}=1)P(R_t=d|S_t=k,F_{t-1}=1)\\
&\quad P(O_{1:t-1},S_t=k,F_{t-1}=1)\\
&= \beta_{t-1}^*(k)p_k(d)\alpha_{t-1}^*(k).
\end{aligned} \tag{1.39}$$

For the computation of (1.30), we first define more intermediate quantities:

$$\gamma_t(k) = P(S_t = k, F_t = 1|O_{1:T}) \propto \alpha_t(k)\beta_t(k)$$

and,

$$\gamma_t^*(k) = P(S_{t+1} = k, F_t = 1|O_{1:T}) \propto \alpha_t^*(k)\beta_t^*(k),$$

then we can rewrite (1.30) by making clear the scheme proposed by Guédon (2003),

$$\begin{aligned} P(S_t = k|O_{1:T}) &= \sum_{k'=1}^{K} P(S_{t+1} = k', S_t = k|O_{1:T}) \\ &= P(S_{t+1} = k, S_t = k|O_{1:T}) + P(S_{t+1} \neq k, S_t = k|O_{1:T}) \\ &= P(S_{t+1} = k|O_{1:T}) - P(S_{t+1} = k, F_t = 1|O_{1:T}) + P(S_t = k, F_t = 1, O_{1:T}) \\ &= P(S_{t+1} = k|O_{1:T}) - \gamma_t^*(k) + \gamma_t(k) \\ &= \sum_{t'=T}^{t} \gamma_{t'}(k) - \gamma_{t'}^*(k). \end{aligned} \tag{1.40}$$

And hence, we have $P(S_{t+1} = i|O_{1:T}) = P(S_t = k|O_{1:T}) + \gamma_t^*(k') - \gamma_t(k')$, which can be computed via induction with the first term being $P(S_1 = k|O_{1:T}) = P(S_1 = k, F_0 = 1|O_{1:T}) = \gamma_0^*(k)$, and which works in the case were the SMC process starts at time $t = 0$.

The M-step maximizes $\mathcal{Q}(\theta, \theta^{old})$ w.r.t. the parameters. The updated parameter formulas are computed using the ESS as follows:

$$\hat{\pi}_k = P(S_1 = k|O_{1:T}; \theta^{old}),$$

$$\hat{A}_{k'k} = \frac{\sum_{t=2}^{T} P(S_t = k, S_{t-1} = k', F_{t-1} = 1|O_{1:T}; \theta^{old})}{\sum_{k=1}^{K}\sum_{t=2}^{T} P(S_t = k, S_{t-1} = k', F_{t-1} = 1|O_{1:T}; \theta^{old})},$$

$$\hat{b}_k(v_g) = \frac{\sum_{t=1}^{T} P(S_t = k|O_{1:T}, \theta^{old})\mathbb{1}\{O_t = v_g\}}{\sum_{t=1}^{T} P(S_t = k|O_{1:T}, \theta^{old})},$$

$$\hat{p}_k(d) = \frac{\sum_{t=1}^{T} P(R_t = d, S_t = k, F_{t-1} = 1|O_{1:T}; \theta^{old})}{\sum_{d'=1}^{\mathcal{D}}\sum_{t=1}^{T} P(R_t = d', S_t = k, F_{t-1} = 1|O_{1:T}; \theta^{old})},$$

where $\hat{p}_k(d)$ is estimated here as a non parametric, or multinomial distribution, but we can obviously fit various discrete distributions. More details about this subject were provided in section 1.3.3 on the fit of Geometric, Poisson or Negative Binomial distributions. Another discussion can be found in Guédon (2003).

3.5 Asymptotic properties

Proposition 1. *Under some border constraints on the transition matrix, an EDHMM's parameters are identifiable up to K permutations.*

Proof. An EDHMM can be seen as an HMM where latent variables lie in the state space $\{1,...,K\} \times \{1,...,D\}$, that is the cross product between state values and their durations. This property is particularly interesting since HSMM may then inherent some of the HMM properties such as identifiability which has been determined for HMM (Leroux, 1992; Douc et al., 2011). □

Moreover, under the following assumptions:

Assumption 6. *The SMC is irreducible.*

Assumption 7. *The conditional sojourn time distributions have finite support.*

Assumption 8. *There exist a right censored observed sequence s.t. its Fisher information matrix is invertible.*

Barbu and Limnios (2006, 2009) proved that:

- all the estimators are strongly consistent as the sequence length tends to the infinity, assumptions (6) and (7),
- all the parameters are asymptotically normal, assumptions (6), (7) and (8).

Note that these properties hold for a single observation sequence.

3.6 State sequence restoration

Similarly to HMM, the state sequence restoration consists in finding the best state sequence given an observed sequence. Different approaches have been discussed in section 2.5 concerning HMM. Here, we directly focus on the most popular one: the MAP computed using the **Viterbi HSMM algorithm**. The MAP is the same as HMM, given by equation (1.24). What differs is the recursive max product equation, i.e. the probability to end up in state k at time t and to transit at time $t+1$ given that the most likely path was previously taken

$$\begin{aligned}\delta_t(k) &\equiv \max_{s_1,\dots s_{t-1}} P(F_t = 1, S_{1:t-1} = s_{1:t-1}, S_t = k, O_{1:t} = o_{1:t}) \\ &= \max_{s_{t-1}} \left\{ \max_{1\le t'\le t} \left[\left(\prod_{u=t'}^{t} b_k(o_u) \right) p_k(t-t') \max_{k'\neq k} \left(A_{kk'} \delta_{k'}(t') \right) \right], p_k(t)\pi_k \prod_{u=1}^{t} b_k(o_u) \right\}.\end{aligned} \tag{1.41}$$

The Viterbi HSMM algorithm, like the Forward-Backward is much more complex than the HMM's since we need to find the best transitions instants t' and therefore we need to max over all of them. This can be seen in the left term of the max of equation (1.41) while the right term computes the max in the case that no transition has happened yet.

The optimal sequence is computed using the traceback. At each time t and for each state k, two backpointers should be recorded. The first one should store the optimal previous state while the second one should record the optimal preceding time of transition from each optimal preceding state, i.e. the optimal state duration.

Note that equation (1.41) only holds for the simplifying assumption. See Yu (2010) for the general assumption and Guédon (2003) for the simplifying assumption. Moreover, there has been plenty of Viterbi HSMM algorithms regarding the sojourn time assumptions. See the following papers for Viterbi algorithms on variable transition HMM (Ljolje and Levinson, 1991; Ramesh and Wilpon, 1992; Chen et al., 1993), for explicit duration (Burshtein, 1996), for Marhasev (Marhasev et al., 2006), for residential time HMM (Yu and Kobayashi, 2003).

Guédon (2007) provides a N-best list of restored state sequences.

3.7 Model Selection

Model selection refers as setting hyperparameters of the model or of the algorithms used. For a HSMM computed with exact inference, there are three main issues: choosing the right number of states, seeking the global maximum of the likelihood function, and choosing the right topology of the transition matrix and assumptions. Hereunder, we give insights on the two first issues while we consider that the third one should be chosen according to the specifications of the data, see Stolcke and Omohundro (1993) and Brand (1999) if interested by these questions.

Number of clusters. Up to this point, we have considered the number of clusters, i.e. K fixed. However, in some applications, this number is unknown and should be determined. One of the main issues when optimizing the likelihood of the data is that we can always improve it by adding another component and fit the noise of the data. It is therefore crucial to also take into account the complexity of the model, given by its number of parameters. Possible solutions are:

- a **grid-search** over a set of values for K with a given a goodness-of-fit versus complexity tradeoff-based objective function such as AIC (Akaike, 1987), BIC (Schwarz et al., 1978), ICL (Biernacki et al., 2000), Entropy (Durand and Guédon, 2016), cross-validated likelihood (Celeux and Durand, 2008).
- use an ensemble learning algorithm to decrease the number of components as EM iterates, see (MacKay, 1997) for the HMM case, or variational Bayes, see (Beal et al., 2003) for the HMM case.
- a **Bayesian nonparametric** HSMM framework based on the hierarchical Dirichlet process proposed by Johnson and Willsky (2012, 2013),

Seek of the global maximum: the Holy Grail ? We showed in Lemma (1.14) that the EM algorithm optimizes a lower bound of the log-likelihood. This lower bound is a local maximum of the likelihood function and there exists plenty of it. A key question is therefore, how to approach or get closer to the global maximum. There is no theoretical result to this question yet, however there has been empirical studies, notably for HMMs and Mixture Models (MMs).

- Juan et al. (2004) proposed an empirical study of the comparison of 6 different initialization techniques for HMM with Bernoulli observed process. It turned out that a simple **parameter jitter in the hypercube center** gave the best results. This is the current initialization technique used in the python package *hmmlearn*[1]. The author proposed a novel initialization technique, performing slightly worse than the jittered hypercube center, which they called **random prototypes** and aims at computing parameter estimates on a subsample, adding jitter, and using it as a starting value for EM. See Karlis and Xekalaki (2003) for a review of the existing methods for mixture models.

[1] https://github.com/hmmlearn/hmmlearn

- Biernacki et al. (2003) proposed a framework called **Search/Run/Select** (SRS) for finding the best initial estimates of a Gaussian mixture model using EM variations such as **Stochastic EM** (SEM) or **Classification EM** (CEM). The SRS technique consists in firstly running few iterations of EM, SEM or CEM with random data points as initial centers. This gives us a starting value of EM which is run until convergence. Finally, the solution providing the best likelihood among all starting values is selected. In practice, there was no sensible difference between all the methods, but the standard EM initialization strategy was sometimes performing slightly worse. The authors warn users that this heuristic framework may sometimes lead to spurious local maximizers when sometimes it may be more interesting to select a local maximizer with a larger domain of attraction because it can be seen as a more stable one.

- some recent works have tried to take advantage of the increasingly popular Wasserstein distance from optimal transport. It has already shown promising results in terms of likelihood as well as the stability of the cluster in the Gaussian Mixture Models (Kolouri et al., 2018) by using a sliced Wasserstein distance (Kolouri et al., 2017).

Chapter 2

Eye-movement analysis using Hidden semi-Markov Models

Contents

1 Introduction to eye-movement data

In this section, we give an insight on eye-movement features, then describe the experimentation, which was first proposed by Frey et al. (2013), subsequently relate eye-movement segmentation with reading strategies by discussing the results found in Simola et al. (2008). We then describe the data preprocessing chain used in order to build an observed process to feed to HSMMs.

1.1 State of the art

Let us first define a couple of eye-movement-related concepts.

Definition 1. *A **fixation** is an immobilization of the visual gaze during a few milliseconds.*

Definition 2. *A **saccade** is a brief movement of the eyes between two fixations.*

Definition 3. *A **scanpath** is a series of fixations and saccades, with their positions and durations, recorded for a certain amount of time (e.g. during a given task).*

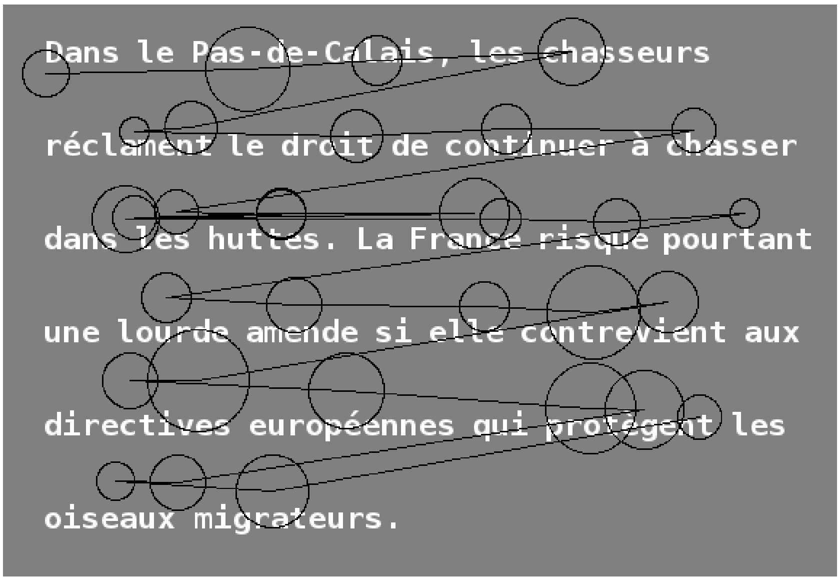
Example 1. *Figure 2.1 provides an example of a scanpath. Fixations are illustrated by circles, which radius is proportional to the duration, whereas, saccades are represented by the lines between two fixations.*

Definition 4. *A **refixation** is the action to perform consecutive fixations on the same word.*

Definition 5. *A **regression** is the action to perform a saccade, and therefore a fixation, on a preceding word in the text. In latin languages, the saccade can be backward or upward.*

Definition 6. *A **progression** is the action to perform a saccade, and therefore a fixation, on a word succeeding in the text. In latin languages, the saccade can be forward or downward.*

Information provided by eye-movements. Since the eye-tracker was invented in 1948 by Hartridge and Thompson, the reading and information processing research has risen, notably from the 70s onwards. Empirical studies have shown that eye-movement itself holds information about the reading process. For example, longer fixations have been observed on misspelled or less common words, see Rayner (1998); Rayner et al. (2012). More, recent studies discuss much deeper topics such as the characteristics of eye movements, the perceptual span, the information integration across saccades, the eye movement control and lastly, individual differences.



Dans le Pas-de-Calais, les chasseurs
réclament le droit de continuer à chasser
dans les huttes. La France risque pourtant
une lourde amende si elle contrevient aux
directives européennes qui protègent les
oiseaux migrateurs.

Figure 2.1: Example of a scanpath. A line corresponds to a saccade. A circle matches to a fixation and its radius is proportional to the duration of the fixation. The larger radius of the circle, the longer the fixation.

Reading strategies. Carver (1992, 2000) has shown that reading processes are mainly affected by the type of task being performed by the subject. The processes were characterized as reading strategies. The engendered effect is noticeable through different types of indicators such as the **reading rate**, the seriality of the words processed, or if they are processed more than once. Carver argued that strategies could simply be clustered by comparing reading rates. He also advocated that reading strategies are different cognitive processes from which readers transit more or less efficiently according to their skill. As a consequence, switching to a higher speed gear implies: decreasing the mean fixation duration, decreasing the mean number of fixation, decreasing the mean number of regressions and increasing the length of forward saccades. Reading strategies are described subsequently:

- **Scanning** is the quickest reading strategy and is generally used for tasks which require a lexical access only such as word search. The given reading rate is 600 words per minutes (wpm).
- **Skimming** is 25% slower than scanning with 450 wpm and consists of adding a semantic access to words. It generally allows the reader to get just enough information to know what the text is about.

- **Rauding** is the contraction of "reading" and "auding", is achieved at a rate of 300 wpm and is basically the default reading strategy which implies sentential integration.
- **Learning** is much deeper than rauding; it implies idea remembering and the ability to answer text comprehension questions. It is performed at 200 wpm.
- **Memorizing** is performed at 138 wpm is by far the slowest reading mechanism and consists of memorizing information by re-reading sentences in order to rehearse facts in a longer term than the previous strategies.

Carver (1992) and Freese (1997) mention that proficient readers demonstrated flexibility by shifting to the appropriate strategy when required. Rauding is used as a central process and users adjust their reading rate when encountering difficulties. Consequently, proficient readers are better at adapting their reading speed by lowering it if the text comprehension is difficult or by increasing it if the text does not provide any information regarding a task. This highlights the fact that the reading rate is closely related to the comprehension of the text. Additionally, the studies showed that there was no significant difference regarding reading speeds but text comprehension between proficient and unsuccessful readers. Authors also put forward individual differences due to an individual's own thinking rate, working-memory capacity, cognitive speed, age, practice; see Hyönä et al. (2002) for more information about individual differences.

Eye-movement segmentation. There has been a wide variety of reading models in information search which can be distinguished in two classes: experimentally-driven models and data-driven models. The former is the most common and consists in building a simulation model which decomposes algorithmically the process of information search, adjusting the model structure, its parameters and evaluating its goodness-of-fit by comparing it to real human experiments. For instance, are the E-Z reader (Reichle et al., 2012) tries to evaluate when and where will the next fixation land using a decomposition of the microprocesses of reading. The model of Lemaire et al. (2011) proposes to predict eye-movement positions using a linear combination of well-defined parameters, such as a word's probability to be fixed, and which vary according to the task type. The latter class of reading models we focus on in this thesis, is based on reading strategies segmentation through statistical modeling. To our knowledge, there is only one instance, based on HMMs, by Simola et al. (2008). The authors modeled the scanpath as a time series by extracting four output processes: the log of the fixation duration in milliseconds (ms) modeled by a Gaussian, the log of the saccade amplitude in px modeled by a Gaussian, the outgoing saccade direction modeled by a Multinomial, and a Bernoulli indicating if the word currently fixed has already been fixed or not. Their experiments were based on three different tasks, namely, word search, question/answer and title choice for a text. They to discriminate the three tasks using a discriminative HMM with one sub-HMM per task type. The authors showed that HMMs were performing well not only to discriminate task but that it was also able to uncover reading strategies which they identified as rauding, scanning, and decision. In conclusion, they suggested tracks of improvements noting that the naturally geometric state sojourn distribution was not fitted for this kind of data.

HMMs have also been used in non-reading tasks to segment eye movements, notably in face recognition (Chuk et al., 2014) and scene exploration (Hayashi, 2003; Coutrot et al., 2018), visual processing control (Rimey and Brown, 1991), fixation-saccade separation (Salvucci and Goldberg, 2000), visual attention (Liechty et al., 2003), implicit feedback relevence (Salojärvi et al., 2005), eye gaze prediction in video streaming (Feng et al., 2011).

In conclusion, HMMs seem to be perfectly suited tools for modeling eye movements due to their changes in dynamics within a same task. Each hidden state is linked with a cognitive process that is indirectly observed through eye-movement features. Hence, the conditional probability distribution (CPD) of the observed eye-movement feature at time t only depends on the hidden cognitive state at time t which only depends on the hidden cognitive state at time $t-1$. Each eye-movement features distribution is different per state. Therefore, a change of state is characterized by a change of dynamics in the eye movements. Moreover, HSMM generalizes HMM and proposes to adjust a parametric sojourn distribution for each state. This perspective is also stated as a perspective in the work of Simola et al. (2008) as Geometric distribution does not seem to be suited to model the duration of reading strategies.

1.2 Material and methods

In this section, we give a brief highlight of the material and methods of the experiment which are necessary to understand the choices made in the statistical modeling. For more information concerning the process, refer to Frey et al. (2013). Note that, small data preprocessing changes were made compared to the referenced experiment in order to be better adapted to a HSMM kind of modeling.

Participants. Initially, there were 21 participants. We rejected 6 of them because they did not follow the rules of the experiment thoroughly or data was too noisy during the acquisition with the eye tracker.

Textual Material. Texts were presented to the participants. These texts, in French, are extracted and corrected from the French newspaper LeMonde, edition 1999. Texts were given a topic and were constructed around 3 types: ones which were highly related (HR) to the topic, moderately related (MR) to the topic and unrelated (UR) to the topic. There were 60 texts of each type, hence 180 in total. The semantic relatedness of the text to the topic was controlled by Latent Semantic Analysis (LSA), Deerwester et al. (1990). LSA is a natural language processing method which consists in building a term-document matrix which counts occurrences of words within a document and performs a single value decomposition in order to reduce the dimension among the document axis and project words in a smaller space. In this space, words that have a closer semantic are also closer given a similarity measure, usually the cosine similarity. All the texts were composed of an average of 5.18 ± 0.7 (mean plus or minus standard deviation) sentences and 30.1 ± 2.9 words. Each word was composed of an average of 5.34 ± 3.24 characters. The average number of lines was 5.18 ± 0.68. In average, the text was displayed with 40.1 ± 5.4 characters per line.

Experimental Procedure The experimental protocol is presented in Figure 2.2. The goal of the experiment was to assess either the text was related to a given topic or not. First the topic was presented to the readers and then they clicked to start the experiment. Then a fixation cross was presented to them to indicate the location of the beginning of the text. The duration of this step was random so that the user cannot anticipate the starting moment. They also did not know whether the text is HR/MR/UR so that he cannot plan on a search strategy mechanics in advance. When the text was displayed, readers needed to answer as soon as possible. The task was then repeated for the 180 texts with breaks in-between. The text were also randomly ordered for each subject. This given task was closely related to information search and decision making. Consequently, we expected subjects to mainly use rauding and skimming but also seldom scanning.

EEG and Eye tracking acquisition Along the experiment, electrical cerebral activity was measured through a 32-channel electroencephalogram (EEG) with 1000 Hz

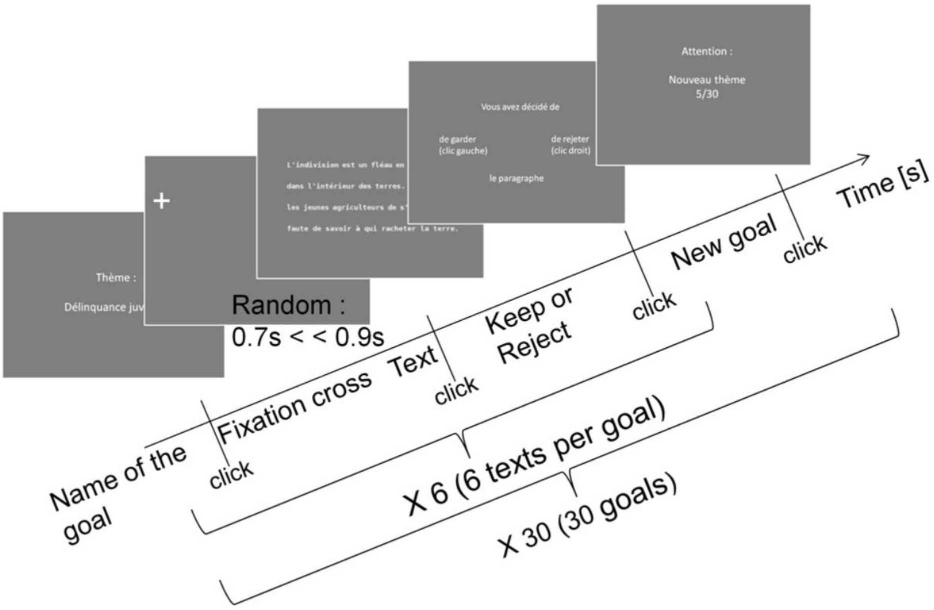


Figure 2.2: Experimental protocol from Frey et al. (2013).

sampling rate. X/Y eye positions on screen were collected using an eye tracker. The minimum fixation duration threshold was set to be 80ms whereas the maximum duration was 600ms. The font size / eye-to-screen distance ratio such that the fovea area (the sharp central vision) was composed of 3.8 characters. Both the measures were upper-bounded by 10 seconds for each text.

Data enrichment: from fixations to words. The eye tracker gave the position of the fixations on the screen. A posteriori, it was necessary to know which word was being processed by the participant. First, the **word identification span** was defined as the necessary area from which a word can be identified. This span varies according to the direction of the lecture, the alphabet, or the language, but can also be micro-context related as it was for several reading models such as EZ-Reader Reichle et al. (1998, 2003) or the SWIFT model Engbert et al. (2005). For simplicity, we used a fixed span, that is considered for most of Latin languages (Rayner, 1998), an asymmetrical window of 4 characters left and 8 characters right to the fixation. Moreover, a word may not entirely be located in the word identification span. Based on the study of Farid and Grainger (1996), we considered a word to be processed if at least 1/3 of its beginning or 2/3 of its end was inside the window. This result was obviously language sensitive, only valid in French, and considers that the important root of the word necessary to its understanding is located at the beginning of the word. Finally, another hypothesis

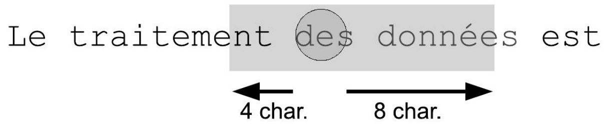


Figure 2.3: The fixation (circle) and the word identification span (rectangle).

had to be made on the processed word within the window since several words might be captured. For this, we assumed that only one word could be processed at the same time and that this word was chosen to be the one which was the closest to the fixation center and that was not a stop word. A stop word is a word that is so common that it does not provide any semantic information. An example of word identification is provided in Figure 2.3. The circle represents the fixation while the rectangle is the word identification span. Here the French word "des", "some" in English, is a stop word, therefore the processing affectation is made with "données", "data" in English, since at least 1/3 of its characters are inside the window.

1.3 Building the output process for HSMM

Up to this point, we know what word is being processed at each fixation, the fixation duration, and similarly to Simola et al. (2008), we can compute several other variables such as the outgoing saccade amplitude, or the saccade direction. The goal was to find variables that are discriminant enough through states and that represent, at least partially, reading strategy. These variables must also be suited regarding a set of possible distributions. Subsequently, we state and discuss some of the preprocessing and modeling choices that were made.

Forward selection strategy. There are many different possibilities to preprocess the data, select the model, find the right filters on the data, define the output process itself, the number of hidden states, the random initialization strategy, or the model selection through different criterion. Each of them could lead to a different model with its own interpretation. Is is straightforward that all combinations cannot be tried out in polynomial time. Therefore, we set up a forward selection heuristic strategy which consists in selecting every possible preprocessing feature one by one and testing its goodness compared to the previous set of preprocessing features, and then to keep or reject it accordingly. The model is assessed at the end of every cycle and repeats until

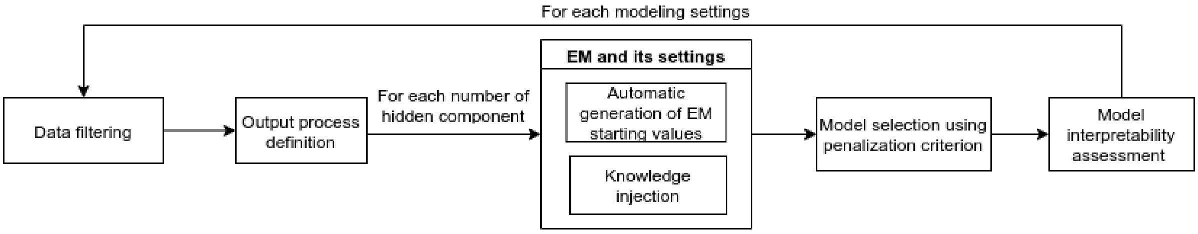

For each modeling settings
Data filtering
Output process definition
For each number of hidden component
EM and its settings
Automatic generation of EM starting values
Knowledge injection
Model selection using penalization criterion
Model interpretability assessment

Figure 2.4: Eye-movement data preprocessing pipeline.

the addition or rejection of every preprocessing element has been tested. The proposed pipeline is shown in Figure 2.4.

Model assumptions. In section 3.2, we presented diverse assumptions regarding the sojourn time and the state sequence censoring. Firstly, we chose to focus on the explicit duration HMM because it is the simplest HSMM, with the least number of parameters. Plus, it allows to fully characterize a state by its initial probabilities, transition probabilities to another state and then, its within-state sojourn distribution. We wanted to highlight the importance of having a transition matrix which diagonal is filled with 0 so that its sojourn distribution characterize a state. If not, as in the variable duration HMM, the sojourn distribution would have been non explicit since characterized by its distribution together with a probability of returning in the same state. Secondly, we considered the simplifying time assumption 4 since we made sure that the experiment started at a given time without any prior information such that the reader was agnostic to which strategy to start with, and that the ending time of the experiment corresponded to the decision. Finally, we considered that changes in reading strategies were expressed through changes of semi-Markovian regime, hence hidden states represented reading strategies.

Data filters. Unlike simulated data, real data is not always as neat as we would like it to be. Plus, we noticed that results could be sensitive to changes in the data which lead us to search for the right filters in order to reduce the uncertainty of the model, expressed through the uncertainty of the state sequence restoration. We refer to Durand and Guédon (2014) where the authors provided a local entropy-based tool for quantifying the uncertainty of a restored state sequence in HMMs, Markov trees setting and HSMMs. From this, a global entropy measure can straightforwardly be computed. We tested the following filters:

- **short sequences** were removed because they were be non characteristic of the given task. The acceptance threshold was set to four or more fixations.
- **double human filter**: manual (human) scanpaths rejection with double check which for scanpaths with acquisition issues such as:
  - top drift, too many fixations between the lines (uncertainty), line skip leading to regression when it looks like a readjustment fixation,
  - the eye tracker estimated too many wrong positions,
  - the return sweep was pathologically pointing at the current line instead of the next line leading to backward movements rather than downward movements.
- **subject filter**: we tried to remove subjects whose behaviour was too atypical or who did not respect the "game" rules, that is, they did not try to reply as soon as possible but re-read the text several times to increase their answer's accuracy.

Time step. Before we introduce the output process, the granularity of the information should be cleared out, i.e. the measure of processing time. Oculometric data is conveniently analyzed at the **fixation step**. A duration was computed, see Rayner (1998) for discussions about the fixation duration computation, as well as characteristics of the outgoing saccade such as its direction or amplitude and hence, text-related features.

Choice of variables. In accordance with Simola et al. (2008), several possible output processes were tested, some were tried separately while some others were combined:

- the fixation duration (in ms) modeled by a log-Normal distribution,
- the outgoing saccade amplitude (in px) modeled by a log-Normal distribution,
- the outgoing saccade direction (upward, forward, downward, backward) plus a factor indicating if it is the last fixation, modeled by a Multinomial distribution,
- the number of characters skipped in the outgoing saccade,
- the number of words skipped in the outgoing saccade,

- the **readmode**, a categorized measure of the number of words skipped during the outgoing saccade. Note that the saccade is the key to segment reading phases. Measuring the saccade in pixel present the inconvenient that it is text layout-sensitive. As a matter of fact, the saccade is always longer after every like break in order to go to the next line and these changes could, alas, be interpreted as Markovian regime changes. Hence, measuring the number of words skipped is a more text layout-robust approach. As we discussed in the beginning of the Chapter, saccades can be characterized by progressions, regressions and refixations but we could also imagine differentiating short regressions or progressions with long regressions or progressions respectively. To this end, several readmode factors were tested:

    1. {< -1, -1, 0, 1, >1}: a decomposition considering each factor has the same importance to decompose eye-movement dynamics. < -1 represents long regressions, -1 short regressions, 0 refixation, 1 short progressions and > 1 long progressions,
    2. {<-1, -1, 0, 1, 2, >2}: since it is mostly a forward saccade experience, we tried to gain a higher detail in the decomposition by adding a third level of progressions,
    3. {<-2, -2, -1, 0, 1, 2, >2}: we also tried to distinguish regressions by adding one more level,
    4. {<-1, -1, {0, 1}, 2, >2}: we tried to merge refixations and short progressions to make a rauding strategy emerge from this state.

Parameter learning, model selection and interpretability. For a given output process, we performed parameter learning for a various number of latent states. For each number of states, we focused on a high likelihood search with random initializations of the EM algorithm. This part is discussed in section 2. Then, model selection was performed in order to choose the correct number of states along with the correct set of preprocessing features. Finally, model interpretability was assessed as validation and is discussed in section 3.

Descriptive statistics. Refer to Appendix A for descriptive statistics on the dataset. We presented the average number of fixations, fixation duration and saccade amplitudes per subject in Table A.1, but also readmode frequencies in Table A.2 and good answer rate in Table A.3. We also provided per text type statistics such as readmode frequencies in Table A.5 but also few indicators in Table A.4.

2 Search of the global maximum likelihood

The hidden semi-Markov models has received a lot of attention in the literature as shown in Chapter 1, section 3. Its framework being generic, modeling assumptions have been proposed, mainly focused around the dependencies between the state and its sojourn time, as well as the latent process time censoring Guédon (2003), leading to a wide variety of inference algorithms. Barbu and Limnios (2009) proved the asymptotic convergence and normality of the estimators, but did not provide any detail on the convergence speed or on the multiple sequence framework. The Expectation-Maximization algorithm finds a local maximum of the likelihood and is known to be extremely sensitive to starting values. Plus, in practice, working with a finite amount of data along with multiple short categorical sequences is an encouraging reason to question the optimality of the local maximum found by the Expectation-Maximization algorithm. To our knowledge, most of the contributions of this kind have been done around the independent Mixture Models (MMs), see Biernacki et al. (2003) for Gaussian MMs or Juan et al. (2004) for Bernoulli MMs. While for HMM, what seems to work best is a simple jitter of the parameters around their centers as implemented in the python library hmmlearn[1]. In this section, we tackle the problem with two different strategies. The former consists in giving random initial parameters to the EM algorithm while the latter resides in injecting human knowledge and expertise over EM iterations.

2.1 Choosing EM starting values for a higher likelihood

We propose a new strategy that we call **sequence breaking framework** (SB), which aims at finding high local maxima of the likelihood by choosing starting values for HSMM's EM, for which the randomness is controlled by the observed sequences in order to restrict the search space. The idea is to prevent EM to start with initial parameter values that are independent from the data and that can therefore lead to very low or almost null likelihood values. This strategy is compared to the standard HMM strategy, the jittered-center parameters.

[1] https://hmmlearn.readthedocs.io

2.1.1 Experimental strategy

Multiple sequence framework. So far, we have written down the EDHMM considering only one sequence of observation $\{O_t\}_{t\in[\![1,T]\!]}$ for notation convenience. We now consider that we have multiple observed sequences $\mathbf{O} = \{\{O_t^{(1)}\}_{t\in[\![1,T_1]\!]},\dots,\{O_t^{(M)}\}_{t\in[\![1,T_M]\!]}\}$.

Proposition 2. *For an observed semi-Markov chain $\{S_t,R_t,F_t\}_{t\in[\![1,T]\!]}$ with a corresponding output process $\{O_t\}_{t\in[\![1,T]\!]}$, the MLE of each set of parameters is given by the conditional empirical frequencies.*

Proof. Given that an EDHMM can be expressed as a DBN, see section 2, and since the likelihood of a DBN can be expressed as a global decomposition of the local-likelihood of each node given its parents, see equation 1.13, then the parameters of the EDHMM can be independently optimized by MLE using the conditional empirical frequencies. □

Choosing starting values with the sequence breaking framework. The main idea is to choose a subset of the observed sequences, generate the associated hidden states by sampling and compute the parameter by MLE using proposition 2, as if we were considering that all the random variables were observed. These parameters are then fed as an initial value to EM, which is run on all the observed data. The proposed strategy relies on two intertwined algorithms:

- Algorithm 2 **HighLikelihodSearch**: describes the global framework, it randomly chooses α observed sequences $\mathbf{O}^{(Q_\alpha)}$ from $\mathbf{O}$, generates the corresponding state sequences $\mathbf{S}^{(Q_\alpha)}$ using **SequenceBreaking**, computes the parameters θ^{init} by MLE using Proposition 2 and injects it as a starting value for the EM algorithm which finds the a local maximum of the likelihood for all data $\mathbf{O}$. The goal of sampling sequences randomly from $\mathbf{O}$ is to generate starting values related to the observation process while keeping only a subset to maintain the randomness of the starting values. Note that **Sample**(.) is a function which samples uniformly on the given set.

- Algorithm 3 **SequenceBreaking**: randomly generates a hidden state sequence. Given each observed sequence $\mathbf{O}^{(q)} \in \mathbf{O}^{(Q_\alpha)}$ with its length, it randomly chooses a number of transitions J as well as transition instants I, which "break" the sequences into pieces, and then affects a state randomly to each piece of sequence with the constraints that two consecutive states should be different due to the EDHMM assumption on the transition matrix.

Algorithm 2: HighLikelihoodSearch: High local maximum of the likelihood search by sequence breaking

```
Input: $\alpha \in [\![1,M]\!]$ the number of sequence to sample,
       $N$, the number of initialization
$\hat{\theta} \leftarrow \emptyset$;
for $n \leftarrow 0$ to $N$ do
    Sample $\mathbf{O}^{(Q_\alpha)} \subset \mathbf{O}$ observed sequences s.t. $Q_\alpha \subset [\![1,M]\!]$;
    $\mathbf{S}^{(Q_\alpha)} \leftarrow$ SequenceBreaking($\mathbf{O}^{(Q_\alpha)}$);
    $\theta^{init} \leftarrow \arg\max_\theta \mathscr{L}(\theta; \{\mathbf{O}^{(Q_\alpha)}, \mathbf{S}^{(Q_\alpha)}\})$; # MLE provided by Proposition 2
    $\hat{\theta} \leftarrow \hat{\theta} \cup$ ExpectationMaximization($\theta^{init}, \mathbf{O}$);
end
$\hat{\theta}^* \leftarrow \arg\max_{\hat{\theta}} \mathscr{L}(\hat{\theta}; \mathbf{O})$
Output: $\hat{\theta}^*$, a high local maximum of the log-likelihood.
```

Algorithm 3: SequenceBreaking

```
Input: $\mathbf{O}^{(Q_\alpha)}$, an observed sequence subset of size $\alpha$
for $q \in Q^{(\alpha)}$ do
    $J \leftarrow$ Sample($[\![1, T_q - 1]\!]$); # number of transitions
    $I \leftarrow \emptyset$;
    for $j \leftarrow 0$ to $J$ do
        $i \leftarrow$ Sample($[\![1, T_q]\!]$) s.t. $i \notin I$; # transition instant
        $I \leftarrow I \cup i$
        $\{\mathbf{S}_t^{(q)}\}_{t \in [\![I_{j-1}, i]\!]} \leftarrow$ Sample($[\![1, K]\!]$) s.t. $S_{I_{j-1}}^{(q)} \neq S_{I_{j-1}-1}^{(q)}$; # choose state
    end
    $\{\mathbf{S}_t^{(q)}\}_{t \in [\![i, T_q]\!]} \leftarrow$ Sample($[\![1, K]\!]$) s.t. $S_i^{(q)} \neq S_{I_{j-1}-1}^{(q)}$; # choose final state
end
Output: $\mathbf{S}^{(Q_\alpha)}$, a randomly sampled state sequences
```

Choosing starting values with jittered-center parameters. We compare the proposed methodology with the default strategy used for HMM which consists in selecting slightly perturbed parameters around their centers, i.e. s.t. each event has equal probability, similarly to Juan et al. (2004).

Example 2. *With $K = 2$, we wish to randomly sample π s.t. $\pi_1 = \textbf{\textit{Sample}}([\varepsilon, 1-\varepsilon])$ and $\pi_2 = 1 - \pi_1$, with $\varepsilon \in]0, 0.5]$.*

Example 2 is a specific instance with a Bernoulli distribution. In order to generalize with $K \geq 2$, i.e. to the Multinomial distribution, one solution is to sample from its conjugate, the Dirichlet distribution. Therefore, we apply a Dirichlet sample for each set of parameters except the sojourn distribution which we initialize with a Geometric distribution of parameter $p = 0.1$.

2.1.2 Results

Not only do the experiments consist of numerical comparison of both methods in finding the highest likelihood, but also to compare the convergence speed of EM, for three different datasets, artificial, artificial with noise, and real. The real dataset corresponds to readmode 1 sequences of the experiment described in the previous section.

Datasets. The first dataset $\mathscr{D}^{(a)}$ is artificial, composed of 100 sequences of length 100 each, with $K = 5$ clusters, $G = 5$ factors for the observed variable, and parameters $\pi = (0.2, 0.2, 0.2, 0.2, 0.2)$, $\{p_k(d)\} = (\mathscr{G}(0.2), \mathscr{G}(0.05), \mathscr{N}\mathscr{B}(8, 0.5), \mathscr{P}(4), \mathscr{N}\mathscr{B}(5, 0.1))$ where $\mathscr{G}$ stands for the Geometric distribution, $\mathscr{N}\mathscr{B}$ Negative Binomial and $\mathscr{P}$ Poisson,

$$A = \begin{pmatrix} 0 & 0.5 & 0.3 & 0.1 & 0.1 \\ 0.5 & 0 & 0.1 & 0.3 & 0.1 \\ 0.25 & 0.25 & 0 & 0.25 & 0.25 \\ 0.2 & 0.2 & 0.2 & 0 & 0.4 \\ 0.1 & 0.1 & 0.4 & 0.4 & 0 \end{pmatrix} \quad B = \begin{pmatrix} 0.1 & 0.2 & 0.4 & 0.2 & 0.1 \\ 0.25 & 0.2 & 0.1 & 0.2 & 0.25 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.35 & 0.3 & 0.2 & 0.1 & 0.05 \\ 0.05 & 0.1 & 0.2 & 0.3 & 0.35 \end{pmatrix}$$

The second dataset $\mathscr{D}^{(an)}$ is generated from the first one by replacing 20% of its observations at random. The third dataset $\mathscr{D}^{(r)}$ consists of the bounded number of words skipped during an ocular saccade by different subjects for a reading tasks for which $G = 5$ and we assume $K = 5$. There are 2390 sequences of different lengths, an average of 17 with a standard deviation of 8.

	$\mathscr{D}^{(a)}$	$\mathscr{D}^{(an)}$	$\mathscr{D}^{(r)}$
Sequence Breaking	$\mathbf{-15448 \pm 2.3}$	$\mathbf{-15785 \pm 4.5}$	-50567 ± 236
Jittered-centers	-15452 ± 2.9	-15782 ± 2.6	-50592 ± 274

Table 2.1: Means and standard deviations of maximum likelihood. Significant (<5%) mean differences are boldfaced. $\mathscr{D}^{(a)}$ is the artificial dataset. $\mathscr{D}^{(an)}$ is the artificial dataqet with noise. $\mathscr{D}^{(r)}$ is the real dataset.

Global results. For a global analysis, we compute the mean and standard deviation of all optimal likelihoods for each method and dataset over 100 initializations and a large number of 1000 iterations of EM. Results are presented in table 2.1.

Local results. For a local analysis, we split the 100 initializations into 10 blocks of 10 and compute the max per block. The goal of this analysis is to assess the performance of both the methods on finding a high maximum likelihood with the fewest initializations. For $\mathscr{D}^{(a)}$, sequence breaking performed better than the jittered-centers 8 times out of 10. For $\mathscr{D}^{(an)}$, **9/10**, and for $\mathscr{D}^{(r)}$, 6/10. Significant (<5%) to binomial test results are boldfaced.

Convergence speed. For $\mathscr{D}^{(a)}$, on average, it took the sequence breaking initialization **133** less iterations to converge than the jittered-centers. For $\mathscr{D}^{(an)}$, it took **305** less, and for $\mathscr{D}^{(r)}$, 44 less iterations on average. Significant (<5%) to t-test results are boldfaced.

2.1.3 Discussion and perspectives

Results discussion. Preliminary results seem to indicate that the initializations provided by the sequence breaking framework converge more quickly while being stable with only few initializations. This result is encouraging considering that parameter estimation with exact inference in Hidden semi-Markov Models is already slowed down because of the sojourn distribution estimation compared to HMM's. Indeed, the inference complexity difference is $O((M^2+MD^2)T)$ for HSMM vs $O(M^2T)$ for HMM. At the moment, the proposed strategy has been tested only for a few datasets, with only discrete observations and in the case $K=G=5$. Performing segmentation in this case is considered to be difficult as parameter identifiability is shown to be easier for small values of K and large values of G in Allman et al. (2009) and hence, we expect more local maxima of the likelihood. Also, we have tried the proposed strategy on different sequence lengths and different numbers of sequences to try to establish ideal scenarios of initialization strategies but we still need to provide datasets with much more variety to establish reliable conclusions.

Sequence Breaking framework expected behavior. Firstly, α, the number of sequences to sample can be seen as a control of the randomness of the initial parameters since the higher it is, the more the observed process will be close to the real one. In practise, we have found that a large value of α has higher odds to lead to a large attraction domain of the likelihood than a small one. On the contrary, a small value of α has less odds to be attracted by a large attraction domain and might discover more diverse solutions, better or worse. Secondly, we can explain the quicker convergence of the sequence breaking framework because the initialization provided is already more likely since it takes into account observed values.

Early detection of bad candidates. From a time and energy saving aspect, Biernacki et al. (2003) proposed a framework called *emEM* which aims at running few iterations of EM with different starting values, selecting the one with the highest likelihood before running a big number of EM iterations, using it as a starting value. Similar strategies have been applied using short runs of Stochastic EM, Celeux and Diebolt (1987), and Classification EM, Celeux and Govaert (1992), called *semEM* and *cemEM* respectively. These three methods have shown significant improvements compared to the standard EM for equal processing time. However, this framework has only been tested for Gaussian mixture models. In practise, we have failed to apply *emEM* to the dataset $\mathscr{D}^{(r)}$, Figure 2.5 illustrates a track of explanation. It represents the likelihood over iterations for a selection of EM initializations with 1000 iterations. The selection was performed such that we track the entire run which had the best likelihood after x iterations, $\forall x \in \{20, 50, 150, 400, 1000\}$. The two solutions which turned out to have the highest likelihood after 400 (purple line) and 1000 (red line) EM iterations had a very poor likelihood in the first iterations compared to solutions which had higher likelihood in the beginning but lesser at the end of 1000 EM iterations (blue, sky blue and green lines). Possible explanation is the gradient differences of the local maximum in the likelihood. It also highlights two clear attraction domains which might be due to the bimodality of the reader's HSMMs as we further mention in section 3.

Perspectives. As shown by Meilă and Heckerman (2001), initialization techniques are data dependent, i.e. we should not expect to find an initialization strategy that

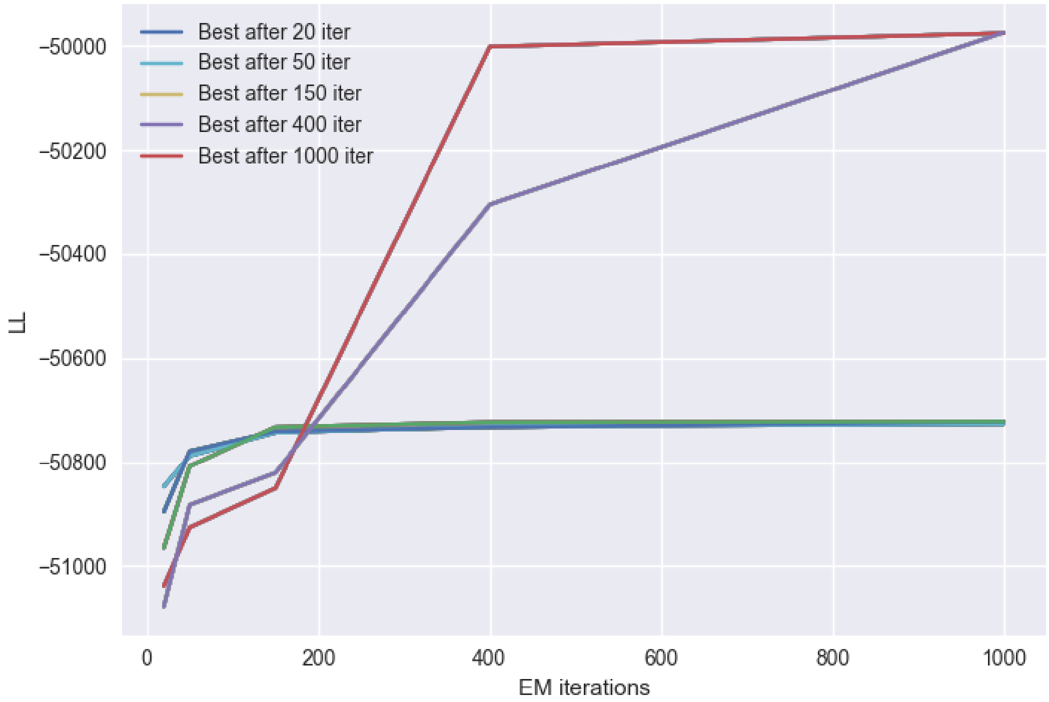


Figure 2.5: Likelihood over iterations for a selection of EM initializations with 1000 iterations. The selection was performed such that we track the entire run which had the best likelihood after x iterations, $\forall x \in \{20, 50, 150, 400, 1000\}$.

outperforms all others on all datasets, but a strategy that works well for a large class of situations arising in practice. We still need to inspect more datasets but also more initialization techniques such as SEM or CEM, to find out the relevance of the sequence breaking framework. Finally, we plan on testing the strategy on more datasets, with multiple output processes, and on continuous data as well.

Additional remarks. Primarily, we would like to note the similarity of the Algorithm 3 with the stick-breaking process view of the Dirichlet process. Similar to the stick breaking process, the probability of having breaks increases with the number of breaks. The transitions instants which are sampled uniformly can be related to a uniform base measure in the stick breaking process. The only difference is that two non consecutive segments could be of the same cluster since the data is modelled by an HSMM. Additionally, Biernacki et al. (2003) recalls that in some cases, local maximizers with a larger attraction region might be preferable because it can be seen as more stable. In return, local maximizers with smaller attraction region might be spurious and not preferred to the previous one even though it could lead to a higher likelihood.

Contribution summary. We proposed a new strategy to search for improved maximum likelihood of HSMM with multiple sequences categorical data which is a significant improvement considering that the current Python implementation of HSMM under the virtual plants library *sequence analysis*[2], as well as the R package *mhsmm* by O'Connell et al. (2011) do not provide a random start option.

2.2 Knowledge injection in parameters

In the previous section, we have focused on automatic strategies to provide higher likelihood values which is what we commonly find in the literature. Rightfully, automatic procedures belong to a much global framework and can be adapted to a wide variety of datasets. However, as we have already mentioned, some strategies work better for some datasets while other strategies work better for other datasets. In this context, and more precisely when modeling is application-oriented, we propose a much more manual approach to perform knowledge injection in the set of model parameters in order to help the model to find a high likelihood. We distinguish two kinds of knowledge, the expert knowledge which is application and data related, and the statistician knowledge which is model related.

Data-related knowledge. The study conducted in Simola et al. (2008) identified 3 states, each with different dynamics. However, since our task is somewhat different, subjects have no prior knowledge about the topic of the text and hence, their reading processes is more likely to be guided by the gathered information. We therefore expect to find more dynamics, and more particularly a strategy slower than the normal reading strategy. Overall, we can expect to recover 3-5 different reading strategies.

Statistician knowledge. The goal of the statistician knowledge is to help EM to converge quicker, or to skip local maximum or simply to dismiss inflection points, i.e. when the gradient cancels out, of the likelihood. In practise with HSMM, it may corresponds to several techniques:

[2] https://github.com/openalea/StructureAnalysis

- set the support of sojourn distribution. In HSMM, the sojourn distributions are discrete. Hence, there are no distinct localization and scaling parameters. To counter this in practise, adhoc procedures are used which introduce a shift parameter and test for the most likely distribution for each shift of each distribution. In practise, we found that such procedure often leads to states with very restricted sojourn support and it might be more useful to set larger support bounds manually.
- set probabilities to 0. In practise, EM converges combinatorial and therefore NP-hard from an automatic point of view

3 Model selection, parameters, restoration and uncertainty

3.1 Selection

Definition 7. *A distribution $P_{\mathbf{O},\theta}$ is characterized by its graphical structure $\mathscr{G}$, the set of hidden states $\mathscr{S} = [\![1, K]\!]$, the family $\mathscr{P} = \{\mathscr{P}_b\}_{b\in B}$ of emission distributions and by the set of parameters $\theta \in \Theta, \Theta \subset \mathbb{R}^{K+K^2+KG}$, we call model $\mathscr{M}$, the set of family distributions s.t.*

$$\mathscr{M}(\mathscr{G}, \mathscr{S}, \mathscr{P}) = \{P_{\mathbf{O},\theta} | \theta \in \Theta(\mathscr{G}, \mathscr{S}, \mathscr{P})\}$$

Concretely in HSMM the graphical structure $\mathscr{G}$ is fixed while $\mathscr{P}$ and $\mathscr{K}$ should be determined. In this part, we address the problem of selection in general. Most commonly, the task is to perform model selection. This area is well-defined and mostly consists in selecting the number of hidden states, the graphical structure, the emission distribution family and structure, or the constraints on the transition matrix. While the last model selection issues are fixed for us, regarding the data and the knowledge about the application, the primary problem of selection of the number of hidden states is addressed hereafter using information theory-based criterion. Another less common approach is to perform a likelihood ratio test to assess a model's superiority, we refered to Giudici et al. (2000) in the context of HMM, but we did not find the existence of such test for HSMM. Additionally, as it was previously introduced, the proposed forward selection strategy requires selecting data filters as well as output processes. On the first hand, assessing a preprocessing token in a supervised context is straightforward since the end goal is always assessed by a performance measure. On the other hand, the assessment of a preprocessing token in an unsupervised task is lead to pure interpretation. As a matter of fact, if changes occur in the data, the criterion cannot be compared. After recalling some model selection criterion, we propose heuristics to face these kind of issues.

3.1.1 Model selection criterion

Quantitative model selection criterion. The **Bayesian Information Criterion** (BIC) of a model $\mathscr{M}$ selects the most likely model conditionally to the observations $\mathbf{O} = \{O_n\}_{n=1}^{N}$ and can be seen as an approximation of the integrated likelihood. We follow Koller et al. (2009) and define the BIC as:

$$BIC_{\mathbf{O}}(\mathscr{M}) = \mathscr{L}_{\mathscr{M},X}(\hat{\theta}) - \frac{\log(N)}{2} d_{\mathscr{M}},$$

where N is the number of observations in $\mathbf{O}$ and $d_{\mathscr{M}}$ is the dimensionality, i.e. the number of free parameters associated with the model $\mathscr{M}$.

The **Integrated Completed Likelihood** (ICL) introduced by Biernacki et al. (2000) originates from a classification perspective, its goal is to find the model which separates best the hidden states.

$$\begin{aligned} ICL_{\mathbf{O}}(\mathscr{M}) &= \mathscr{L}_{\mathscr{M},\mathbf{O},\hat{\mathbf{s}}}(\hat{\theta}) - \frac{\log(N)}{2} d_{\mathscr{M}} \\ &= \mathscr{L}_{\mathscr{M},\mathbf{O}}(\hat{\theta}) + H_{\hat{\theta}}(\hat{\mathbf{s}}|\mathbf{O}) - \frac{\log(N)}{2} d_{\mathscr{M}} \end{aligned} \tag{2.1}$$

where $H_{\hat{\theta}}(\hat{\mathbf{s}}|\mathbf{O})$ is the conditional entropy which measures the disorder (uncertainty) of $\hat{\mathbf{s}}$ conditionally to $\mathbf{O}$, and $\hat{\mathbf{s}}$ being the restored state sequence via the Viterbi algorithm.

Qualitative selection criterion. As stated in Burnham and Anderson (1998), if a model makes no sense regarding the application, it should not be a part of the solutions. As a consequence, we set up practical data-dependent as well as parameters-dependent interpretation criterion:

- **emission distribution** should have an interpretable features regarding reading strategies and information search strategies that can be found in the literature while also taking into account the task specificities.
- **transition and initialization distributions** should be coherent to the task. In an information search task, with rather short texts, we do not expect users to go back to a normal reading strategy after being in a decision state. Similarly, we do not

expect subjects to start with a decision state. Therefore we expect the transition matrix and initialization distributions to have some (almost) zero values.

- **state sojourn distributions** should be coherent with the length of sequences and the expected number of dynamic changes within the same scanpath. As a consequence, we rejected models which had very short state duration (1-2 fixations on average) together with the ones which had very long durations and almost no transition. It should be noted that models within the range of the last case should be treated with particular attention since long state duration encourages less transitions and therefore less uncertainty, which is directly linked to the quantitative criterion such as entropy and ICL. In other words, models with long state duration have a lesser entropy and ICL. Both quantitative and qualitative criterion should work in harmony for model selection.
- in general, parameters should not be affected and sensitive towards data changes. For example, a state should not be created to satisfy a specific behavior which occurs in only few scanpaths.

3.1.2 Selecting data filters, output processes and models: methodology

Comparing data filters. The goal is to choose experimentally whether a data filter should be applied or not. Let us first denote $\mathbf{O}_F$ the dataset with all the data, and $\mathbf{O}_Q \subset \mathbf{O}_F$, a subset of the data for which a filter has been applied. Note that $\mathbf{O}_F$ may already be a filtered subset. The proposed experimental procedure to decide if $\mathbf{O}_Q$ should be chosen over $\mathbf{O}_F$ is bootstrap-based (MacKinnon, 2009) and is described by Algorithm 4. The strategy consists in learning the parameters θ_Q, θ_F on training samples, $\mathbf{O}_{Q_{train}}, \mathbf{O}_{F_{train}}$ respectively, where $\mathbf{O}_{F_{train}}$ contains some additional data that has not been filtered and that is not present in $\mathbf{O}_Q$ (and hence in $\mathbf{O}_{Q_{train}}$), and then computing the likelihood on a testing sample $\mathbf{O}_{test}$ that has been unobserved from both the training sets. The procedure is repeated B times and the filter is accepted and kept if it exceeds the threshold α.

Comparing datasets with different output processes. Comparing different output processes leads to comparing different and disjoint datasets. Thereupon, comparing unequal datasets, models, and parameters but aiming at modeling the same observed process. For this task, we mainly rely on qualitative interpretation criterion which were

Algorithm 4: SelectDataFilter

```
Input: $\mathbf{O}_F$, the full dataset,
       $\mathbf{O}_Q \in \mathbf{O}_F$, a filtered dataset,
       $B \in \mathbb{N}$, a number of repetition,
       $\alpha \in [0,1]$, an acceptance threshold.
$i \leftarrow 0$ ;
for $b \leftarrow 0$ to $B$ do
    $\mathbf{O}_{Q_{train}} \leftarrow$ SampleWithReplacement$(\mathbf{O}_Q)$
    $\mathbf{O}_{F_{train}} \leftarrow \mathbf{O}_{Q_{train}} \cup$ Sample$(\mathbf{O}_{F \smallsetminus Q})$
    $\mathbf{O}_{test} \leftarrow \mathbf{O}_{F \smallsetminus Q \smallsetminus F_{train}}$
    $\hat{\theta}_F \leftarrow$ ExpectationMaximization$(\mathbf{O}_F)$
    $\hat{\theta}_Q \leftarrow$ ExpectationMaximization$(\mathbf{O}_Q)$
    if $\mathscr{L}_{\mathbf{O}_{test}(\theta_Q)} > \mathscr{L}_{\mathbf{O}_{test}(\theta_F)}$ then
        $i \leftarrow i + 1$
    end
end
if $i/B > \alpha$ then
    $\mathbf{O}^* \leftarrow \mathbf{O}_Q$
else
    $\mathbf{O}^* \leftarrow \mathbf{O}_F$
end
Output: $\mathbf{O}^*$, the best dataset.
```

cited before. Moreover, we still use conditional entropy of the restored state sequence which we use as a measure of disorder with respect to the hidden states segmentation.

Selecting the number of hidden states. An experimental comparative study of the selection of the hidden state number in HMM using information theory-based criterion has already been proposed by Celeux and Durand (2008). The authors showed that BIC and ICL were performing well and had similar behaviors in mixture models, that is ICL favors models that partition the data with the greatest evidence from the hidden states whereas BIC has a tendency to overestimate the complexity of the model. They also showed promising results regarding likelihood cross-validation criterion. However, the likelihood cross-validation was omitted in our comparative study since it is much less computationally efficient.

Comparing different initialization strategies. The comparison of different initialization strategies relies on the search of a higher likelihood discussed in section 2.

However, further we discuss some tracks of comparison about information theory-based criterion.

3.1.3 Application to Eye-movement data

Data, output process and model selection results are reported in Table 2.2. The first set of columns represent meta-data about the model and the second set represent quantitative criterion. Each line corresponds to a combination of data, output process, initialization technique and number of states. Hence, the table is split in four sets of rows that aim at finding:

1. the best data,
2. the best output process,
3. the right number of states,
4. the right initialization method.

We started with the most simple set of preprocessing and modeling tokens and tried to complexify at every step. If it was improving some quantitative and/or qualitative criterion, we kept the preprocessing token otherwise we rejected it and tried some other one, and so on. Hence, we started by using all the data, the simplest output process, 5 states. The number of states can be justified combining Carver's reading strategies along with Simola's reading processes in information search task, we expected 4-5 states: a slow processing strategy between learning and rauding, rauding, skimming, scanning and decision.

Datasets.

- Table 2.2 - row 1: we started with **All data**: 42491 fixations over 2565 sequences.
- Table 2.2 - row 2: since in practise, some scanpaths were irrelevant, see section 1.3, we applied a **Double human filter** (DHF). 175 scanpaths were rejected. The test by bootstrap presented in Algorithm 4 showed that the double human filtered dataset performed better on the test set 82% of the time. Hence, we decided to keep this data filter. There were 2390 sequences and 39564 fixations left.

ID	DF	Output process	HLS	#Seq	N	K	LL	BIC	ICL	Entropy	$d_{\mathscr{M}}$
1	All data	Readmode1	KI	2565	42491	5	-53749	-107808	-132248	12220	29
2(*)	DHF	Readmode1	KI	2390	39564	5	**−49745**(**)	**−99798**(**)	-121341	10771	29
3	DHF ⟍ s04	Readmode1	KI	2245	35062	5	-43500	-86904	-106390	9743	29
4	DHF	Simola et al. (2008)	KI	2390	39564	5	-386717	-774123	-794332	10104	65
5	DHF	Readmode2	KI	2390	39564	5	-61061	-122504	-145666	11581	36
6	DHF	Readmode3	KI	2390	39564	5	-62658	-125741	-148072	11165	40
7	DHF	Readmode4	KI	2390	39564	4	**-49040**	**-98335**	-114390	8027	24
8	DHF	Readmode1	KI	2390	39564	3	-50837	-101876	-126728	12426	19
9	DHF	Readmode1	KI	2390	39564	4	-50776	-101848	-123493	10822	28
10	DHF	Readmode1	KI	2390	39564	6	-50549	-101501	-121635	10067	38
11	DHF	Readmode1	SB	2390	39564	3	-50837	-101876	-126728	12426	19
12(*)	DHF	Readmode1	SB	2390	39564	4	-50769	-101804	**−121321**(**)	9758	26
13(*)	DHF	Readmode1	SB	2390	39564	4	-50744	-101711	**-112578**	**5433**	21
14	DHF	Readmode1	SB	2390	39564	5	-50643	-101636	-120146	9255	33
15	DHF	Readmode1	SB	2390	39564	6	**-49567**	**-99578**	-122744	11583	42

Table 2.2: Models learned with different data filters, output processes and EM High likelihood search settings, number of states, and their corresponding quantitative criterion. Best criterion are bold-faced. Models 12 and 13 can be distinguished by two different runs of EM with different initial parameters. Note that the 3 first rows differs in the number of fixations and therefore should not be compared with the given criterion but with the Algorithm SelectDataFilter 4. (*): models presented in section 3.2. (**): Best criterion among the qualitatively interpretable model class. DF: data filter, DHF: "Double human filter", HLS: high likelihood search, KI: Knowledge Injection, SB: sequence breaking.

- Table 2.2 - row 3: on top of the DHF, we added a **subject-related filter** in order to filter out subject 4, motivated by its atypical behavior. The subject 4 used much more fixations than the average, 31.1 ± 9.3 vs. 16.5 ± 8.5. Hence 135 scanpaths were removed. The bootstrap test showed that the additional filter performed worse 94% of the time. Even though, this subject was taking much more fixations than the others for the same task, this result highlights the stability of HSMM learned on the DHF dataset and shows that such specific reading mechanisms are already included in the current Markovian regimes that only affect the model with more transitions and/or longer sojourn state durations. Subject 4 was therefore kept.

Output process. We tried the output processes presented in section 1.3: a set of output processes based on low-level features presented in Simola et al. (2008) (Table 2.2 - row 4), and a single high-level output process namely the Readmode with 4 different sets of levels in order to handle different aspects of the modeled task (Table 2.2 - row 5-7).

- With a much more complex model, 65 free parameters (row 4) versus 29 (row 2), the output processes used in Simola et al. (2008) showed to have a slightly better discriminant power when comparing entropies (10104 vs 10771). Moreover, the model were poorly interpretated since all sojourn durations were of 1-2 fixations on average.

- The goal of Readmode 2-3 (section 1.3) (rows 5-6) was to extend the Readmode factors. However, it was shown to have a lesser discriminant power in terms of entropy (11581 and 11165 vs 10771) despite having many more parameters (36 and 40 vs 29).

- The Readmode 4 (row 7) was designed to overcome a drawback of Readmode 1 (row 2): the rauding strategy could only be explained by manually merging two states (see section for a more detailed explanation 3.2), hence Readmode 4 regrouped both refixations and short forward saccades in order to simulate the rauding state dynamics. This solution showed large increases in log-likelihood as well as BIC and ICL, however, the model was not interpretable from qualitative criterion. There were redundant reading dynamics and the merging of refixation and short forward saccade made it impossible to dissociate strategies with backward and refixation saccades from strategies with backward and short forward saccades. Hence we decided to keep Readmode 1.

Number of states. 3 to 6 number of hidden states were tested (rows 2, 8-10). The model with 5 states performed better on all criterion (log-likelihood, BIC and ICL) together with the qualitative interpretation of the model. However, as we will discuss subsequently, the true reading processes seemed to indicate that two states are closely linked and should be merged even though they could not be recovered by learning a 4 states model. Models with 6 states did not lead to models with any possible interpretability.

EM initialization strategy. Finally, the knowledge injection (KI) and sequence breaking (SB) initialization strategies were compared (rows 2, 11-15). Moreover, we also tried to learn 3-6 states models with the sequence breaking strategy as a validation on the number of states. First, we found a better likelihood and BIC for a 6-state model, whereas the data and task did not seem to indicate many reading strategies in accordance with Celeux and Durand (2008): BIC might not always be enough penalized. Additionally, we found 2 models (rows 12,13) with 4 states with a lesser entropy and a higher ICL than the model learned with knowledge injection (row 2). More particularly, the model row 13 outperformed all others based on the entropy and ICL. Nevertheless, this model is presented section 3.2 as an example of a spurious maximizer with no interpretation power. The model row 12 resulted from a large attraction domain of the likelihood and is also presented subsequently and has a high interpretation power such as the model row 2. In conclusion, both BIC and ICL performed well and bad on some cases. Sometimes BIC was not penalized enough. Sometimes ICL was too penalizing. We kept models corresponding to rows 2 and 12 that both had a high BIC and ICL. They are subsequently named "Model 1" and "Model 2" respectively.

3.2 Model parameters

In this section, we describe the two models that were considered plausible (Table 2.2 rows 2,12), both from quantitative and qualitative point of view. We also briefly mention a model (row 13) with good quantitative criterion but poor qualitative criterion as an example of a spurious maximizer.

Parameters interpretation. First and foremost, it should be noted that parameter interpretation relies on the asymptotic property that the MLE estimators of the parameters converge to the real parameter. In practise, the presented models and parameters are therefore an approximation of the truth, they do not claim to fit the data perfectly or to describe the true reading processes. The proposed modeling and analysis aims at modeling reading processes/strategies through the hidden states of the HSMM. Therefore, each hidden state represents a reading strategy, and each strategy has its own probabilities to start, given by the initial probabilities, its own probabilities to transit to other states given by the transition matrix, its own sojourn duration given by the state sojourn distribution and its own Readmode (reading dynamics) pattern, given by the emission probabilities. Moreover, transition probabilities should be interpreted with caution. For example, a probability of 0.8 to transit from a scanning strategy to a rauding strategy does not necessarily means that it will happen for 80% of the scanpaths. Indeed, most of the time the trial could just end with scanning strategy. For a complementary indicator, we also provide counts based on hidden state restoration.

States		1 & 2 (NR)		3 (SR)	4 (IS)	5 (SC)
Initial probabilities		.53 (1219)	.25 (324)	.22 (847)	0	0
Transition probabilities	NR	0	1 (6558)	0	0	0
		.74 (5339)	0	0	.21 (1190)	.05 (353)
	SR	0	0	0	0	1 (162)
	IS	0	.07 (3)	0	0	.93 (23)
	SC	0	0	0	0	1
Sojourn	Distribution	G(.13)		NB(33,.77)	NB(3.3,.40)	G(.12)
	Mean ± Std	7.7 ± 3.3		11.1 ± 3.6	5.4 ± 3.2	8.3 ± 7.8
Readmode	Bwd++	.03 (195)	.01 (26)	.04 (366)	.05 (280)	.21 (1715)
	Bwd+	.02 (190)	.01 (76)	.03 (274)	.01 (74)	.05 (352)
	Refixation	.65 (5989)	.03 (62)	.10 (1109)	.26 (1738)	.18 (1302)
	Fwd+	.30 (2259)	.25 (1765)	.25 (2690)	.18 (1082)	.13 (875)
	Fwd++	0 (0)	.70 (5470)	.58 (5611)	.49 (3050)	.43 (3013)
Total counts	8633	7399	10050	6224	7258	
Final state		0	1	685	1166	538

Table 2.3: HSMM parameters for model 1, the hand-crafted local maximum, with counts in parenthesis. NR: normal reading, SR: speed reading, IS: information search, SC: slow confirmation, Bwd++: long regression, Bwd+: short regression, Fwd+: short progression, Fwd++: long progression.

Observation distributions and latent states. Over the 39564 fixations, 7% of them were long regressions, 2% short regressions, 26% were refixations, 22% were short progressions and 43% were long progressions. It should be noted that these statistics are slightly different than what is found in the reading literature where usually 10-15%

States		1 (NR)	2 (SR)	3 (IS)	4 (SC)
Initial probabilities		.72 (1317)	.26 (1068)	.02 (5)	0 (0)
Transition probabilities	NR	0	.23 (53)	.77 (463)	0
	SR	0	0	0	1 (425)
	IS	.76 (141)	0	0	.24 (71)
	SC	0	0	0	1
Sojourn	Distribution	NB(1.24, 0.15)	NB(67,0.85)	NB(1.22, 0.22)	NB(0.34, 0.14)
	Mean ± Std	8.1 ± 6.9	13.1 ± 3.7	5.2 ± 4.3	12.2 ± 9.2
Readmode	Bwd++	0 (0)	.05 (557)	.11 (547)	.21 (1478)
	Bwd+	.01 (245)	.04 (449)	0 (0)	.05 (272)
	Refixation	.33 (6183)	.14 (1750)	.33 (1080)	.18 (1187)
	Fwd+	.31 (4818)	.24 (3010)	.04 (22)	.14 (821)
	Fwd++	.34 (5766)	.53 (7228)	.52 (1843)	.42 (2308)
Total counts		17012	12994	3502	6066
Final state		942	696	256	496

Table 2.4: HSMM parameters for model 2, the local maximizer with large attractivity. NR: normal reading, SR: speed reading, IS: information search, SC: slow confirmation, Bwd++: long regression, Bwd+: short regression, Fwd+: short progression, Fwd++: long progression.

		1 (NR?)	2 (SR?)	3 (?)	4 (SC)
Initial probabilities		.11 (165)	.43 (1044)	.45 (1177)	.01 (4)
Transition probabilities	NR?	0 (0)	0 (0)	.05 (4)	.95 (82)
	SR?	0	0	0	1 (610)
	?	.62 (941)	.38 (236)	0	0
	SC	0	0	0	1
Sojourn	Distribution	NB(17, 0.41)	NB(1.92, 0.17)	1	NB(0.36, 0.15)
	Mean ± Std	26.2 ± 7.9	10.3 ± 7.4	1 ± 0	12.8 ± 9.2
Readmode	Bwd++	.02 (413)	.02 (170)	0 (0)	.17 (1999)
	Bwd+	.01 (237)	.03 (353)	0 (0)	.04 (376)
	Refixation	.35 (6233)	.18 (1430)	.63 (940)	.19 (1597)
	Fwd+	.23 (4099)	.28 (3371)	.25 (237)	.13 (964)
	Fwd++	.38 (7147)	.49 (5976)	.12 (0)	.47 (4022)
Total counts		18129	11200	1177	8958
Final state		1018	670	0	702

Table 2.5: HSMM parameters for model 3, the spurious local maximizer. NR: normal reading, SR: speed reading, IS: information search, SC: slow confirmation, Bwd++: long regression, Bwd+: short regression, Fwd+: short progression, Fwd++: long progression.

of fixations are regressions, Rayner (1998), and only 15% are refixations, O'regan et al. (1984). A possible explanation is that texts present more acronyms and area-specific words with a low word frequency than standard tasks in literature. Such words are known to be factors of refixations, Sereno and Rayner (1992); Rayner and Well (1996). Another possible explanation is our fixation-to-word implementation discussed in section 1.2 which keeps the word with the highest interest in case two words are in the same window. This word is most likely the word with the lowest frequency. Therefore, it is possible to wrongly assign a refixation when it might not be one. However, since the proposed modeling focuses on the eye-movement dynamics as a whole, the identification of the processing states should not affect but simply be taken into consideration for reading strategy identification.

Model 1 , presented in table 2.3 (row 2), was found to have 5 states using quantitative criterion. The two first states, presented a different Markovian dynamics, one was mainly composed of refixation (0.65), short forward saccades (Fwd+) (0.30), while the second was composed of short forward saccades (0.25) and mainly long forward saccades (Fwd++) (0.70). Both had very few backward saccades (Bwd+) (0.04) combined on average, and more interestingly. Both models had a short duration but very high probability to loop, 1 from state 1 to state 2 and 0.74 from state 2 to state 1. Even though they had different Markovian dynamics, from a eye-movements dynamics point of view, the states clearly describe what we termed as **normal reading** (NR) or rauding, using the terminology found in Carver (1990). Moreover, the combination of the two states is corroborated by the model 2 (row 12), presented in table 2.4, which also recovered a state with very similar parameters, a probability of 0.72 to begin the assignment, close to 0.78 for model 1. This state was almost equally (≈ 0.33) composed of refixations, short forward saccades, long forward saccades and very few backward saccades. It can be related to reading word-by-word.

In both models 1 and 2, readers began in the second state with a probability of 0.25. They had more backward fixations than in the normal reading state, fewer refixations, compared to the total number of backward fixations and refixations, and essentially long forward saccades (0.58 and 0.53) rather than short forward saccades (0.25 and 0.24). As a result, we labelled the state **speed reading** (SR).

The dynamics of the third state were slightly different for both the models. Neither had any significant short backward saccade, but mainly refixations (0.26 and 0.33) and long forward saccades (0.49, 0.52). Model 1 had many more short forward saccades than model 2 (0.18 vs 0.04) while model 2 had much longer backward saccades

(Bwd++) (0.05 vs 0.11). Following the idea of the dynamics mainly expressed by model 2 and lightly by model 1, we labelled the state **information search** (IS) because of the low probability of short backward and forward fixations. The process can be opposed to normal reading.

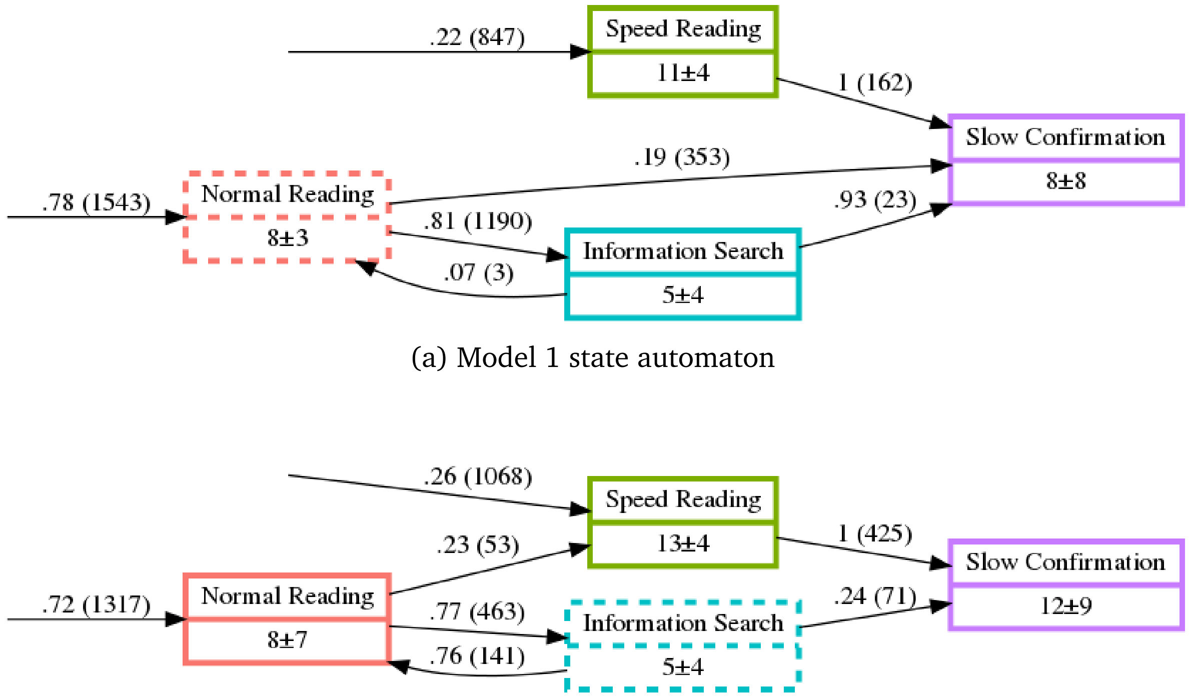
Finally, the fourth state was the same in model 1 and 2. It had many long backward saccades (0.21), and long forward saccades (0.42, 0.43) and not many short backward saccades (0.05), refixations (0.18), or short forward saccades (0.13, 0.14). Plus, as it was used as a final state, it was referred as **slow confirmation** (SC).

Transition probabilities. Let us first be reminded that, processing states last for several fixations and their influence survives across saccades, as pointed out by the study of Yang and McConkie (2005) which encourages us to model their associated duration.

Both the models recovered a sojourn distribution for normal reading state with about 8 fixations but different standard deviations, 3.3 (model 1) vs 6.9 (model 2). They also indicated longer sojourn duration in speed reading state, 11.1 and 13.1 fixations on average with a small standard deviation of 3.6 and 3.7 respectively. Both found information search state to be the shortest with around 5.3 fixations. Slow confirmation had a mean sojourn of 8.3 for model 1 and 12.2 for model 2. Both had large standard deviation (≈ 8.5), indicating very versatile uses.

Figure 2.6 represents the automaton of the HSMM states transitions for model 1, figure 2.6a and model 2, figure 2.6b. Both transition matrices indicate that the process had a left-to-right tendency, i.e. it starts in normal reading or speed reading, may go to information search if it started in normal reading, to finish in slow confirmation state. While it rarely goes backward, except for the information search to normal reading in model 2 with a probability of 0.76 but few occurrences, 141 times. Another key difference is that in model 1 state transitions from normal reading are much more frequent whereas it may be terminal in model 2 due to the higher standard deviation of the strategy's duration. As a consequence, the information search strategy is much more terminal in model 1 than 2, noting that in both cases, the subject takes his decision in almost every state. Hence, the slow confirmation state does not characterize a decision state but rather a state when decision is ambiguous and requires many more fixations to reach the final decision.

Main differences with the study conducted by Simola et al. (2008) lie in the way strategies are used and their duration. Unlike the study of Simola, the subjects of the



(a) Model 1 state automaton

(b) Model 2 state automaton

Figure 2.6: Automatons representing hidden states parameters for model 1 and 2. Each state is represented by a box of different color with its label and the mean and standard deviation of the dwell times below. Arcs between two states represent the probability of transit from one state to another and the associated count in parenthesis. A solid-contoured box indicate that the state is usually terminal whereas a dashed-contoured box indicate that the state is rarely or not terminal. Note that for model 1, both transition probabilities and sojourn duration were recomputed after merging the two corresponding states.

current study had no prior information about the difficulty of the task. Hence they most likely started with a normal reading strategy until they assessed its difficulty, whereas subjects in Simola's study mainly started with a scanning strategy. This may also explain the differences in dwell times. The study of Simola suggested shorter durations for the scanning state (2.8s $\pm$ 2.3 for a Q&A task) whereas more than two time longer (6.1s $\pm$ 5.1) reading states were documented for the same task. On the contrary, our study suggested that speed reading was longer (2.4s $\pm$ 1.5) than normal reading (1.7s $\pm$ 1.3). Besides, global differences on the reading strategy duration may simply be explained by the length of the texts and language specifications, 58 words and 580 characters on average in the study of Simola where texts were in written Finnish, when it was only 30 words and 161 characters on average in the present study that was conducted in French.

Model 3 - the spurious local maximum. Table 2.5 represents the model parameters of the spurious local maxima. The spuriousness can be assessed when focusing on the transition matrix and sojourn distribution. First, it can be seen that state 3 was an initial state with a starting probability of 0.45, then lasted for only one fixation (not random) before transiting to state 1 or 2. State 1 had a very long duration, 26.2 $\pm$ 7.9 and was rarely exited. State 2 was exited more often (610/1628) to end up in state 4. The states could not be labeled into meaningful reading strategies based on their readmode factors except slow confirmation. The low amount of transitions and the weak possibilities offered by the model 3 explain the low uncertainty regarding the state choice and therefore the low entropy and ICL. This analysis highlights that ICL, in the context of HSMM, might be too penalized and it could be beneficial to perform model selection using BIC instead.

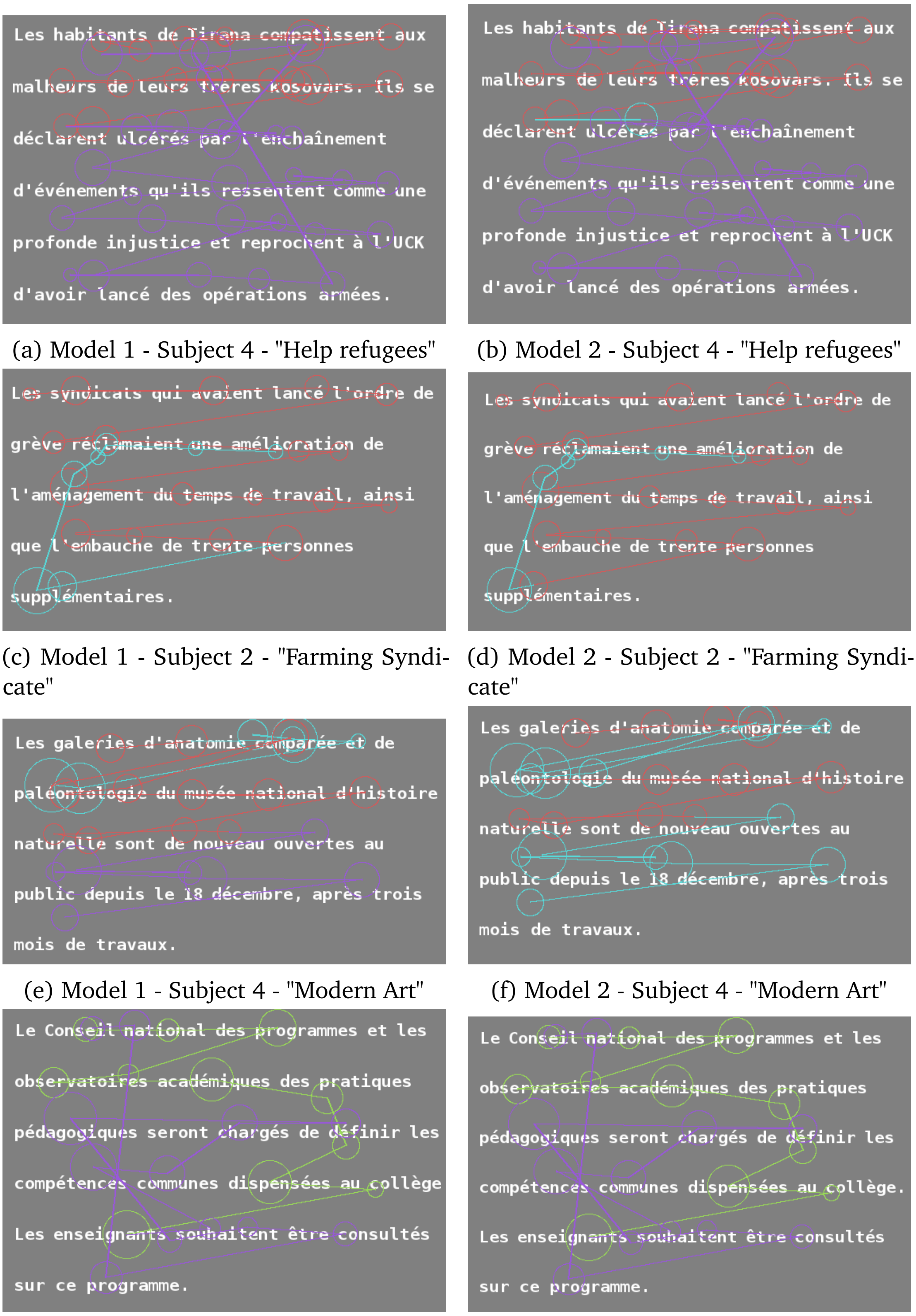
3.3 Restorations

In this section, we provide examples of restored scanpaths for both model 1 and 2, using the Viterbi algorithm presented in Chapter 1, section 3.6, to discuss practical uses of reading strategy before showing concrete effects of uncertainty on restoration. Note that gross patterns of strategies were already highlighted by counts in Tables 2.3 2.4 and discussed in the previous section.

Scanpaths restorations. Figure 2.1 provides a comparison of restoration with model 1 vs model 2 for several scanpaths, with several behaviors. The first scanpath restored with model 1, figure 2.7a, and model 2, figure 2.7b, presents a subject who started with a normal reading strategy where words were processed one by one with several refixations before transiting into a slow confirmation strategy from which backward (short and long) fixations are typical. The main difference between the two models lies in the necessity of an intermediate information search fixation because the model 2 forbids fixations from normal reading to slow confirmation. The second scanpath restored is presented in figures 2.7c and 2.7d. Both the models recovered the same hidden states, the readers started with a word-by-word normal reading process, before transiting to an information search strategy and performing few refixations with a long backward fixation on a past location of the text which probably helped them to take the decision. Indeed, Shimojo et al. (2003) showed that participants tended to look more often at the target before they made their decisions. This may also be corroborated by the studies of Frazier and Rayner (1982); Ehrlich and Rayner (1983); Blanchard and Iran-Nejad (1987) in which authors assessed backtracked eye-movement is performed on misunderstood area which has been memorize. In this case, the participant goes back to the beginning of the second line stating "strike claims" when the topic "Help refugees" was not related. The third scanpath restored, figures 2.7e and 2.7f shows four different state transitions, notably made around target words such as "paleontology" when the topic is "Farming syndicate". Both models detected changes in dynamics and therefore state transition at almost the same places. However, model 2 did not end with a slow decision state regarding that there was no arc from normal reading to slow decision. Finally, the fourth scanpath, figures 2.7g and 2.7h show identical state restorations for both the models. The participant started with a speed reading state in which long progressions and no refixations were typical, before ending in a slow confirmation state in which he re-passed on most of the text, probably due to the lack of information gathering in the previous state.

Uncertainty and state profile exploration. In order to assess the uncertainty of a state sequence restoration along with the potential candidates, Guédon (2007) proposed a tool in order to compute the max posterior state probability at each time t, i.e. at each fixation, for each state k, given by:

$$s_t^{(k)} = \max_{s_{1:t-1}, s_{t+1:T}} P(S_{1:t-1} = s_{1:t-1}, s_t = k, S_{t+1:T} = s_{t+1:T} | O_{1:T}).$$



Les habitants de Tirana compatissent aux malheurs de leurs frères kosovars. Ils se déclarent ulcérés par l'enchaînement d'événements qu'ils ressentent comme une profonde injustice et reprochent à l'UCK d'avoir lancé des opérations armées.

(a) Model 1 - Subject 4 - "Help refugees"

Les habitants de Tirana compatissent aux malheurs de leurs frères kosovars. Ils se déclarent ulcérés par l'enchaînement d'événements qu'ils ressentent comme une profonde injustice et reprochent à l'UCK d'avoir lancé des opérations armées.

(b) Model 2 - Subject 4 - "Help refugees"

Les syndicats qui avaient lancé l'ordre de grève réclamaient une amélioration de l'aménagement du temps de travail, ainsi que l'embauche de trente personnes supplémentaires.

(c) Model 1 - Subject 2 - "Farming Syndicate"

Les syndicats qui avaient lancé l'ordre de grève réclamaient une amélioration de l'aménagement du temps de travail, ainsi que l'embauche de trente personnes supplémentaires.

(d) Model 2 - Subject 2 - "Farming Syndicate"

Les galeries d'anatomie comparée et de paléontologie du musée national d'histoire naturelle sont de nouveau ouvertes au public depuis le 18 décembre, après trois mois de travaux.

(e) Model 1 - Subject 4 - "Modern Art"

Les galeries d'anatomie comparée et de paléontologie du musée national d'histoire naturelle sont de nouveau ouvertes au public depuis le 18 décembre, après trois mois de travaux.

(f) Model 2 - Subject 4 - "Modern Art"

Le Conseil national des programmes et les observatoires académiques des pratiques pédagogiques seront chargés de définir les compétences communes dispensées au collège Les enseignants souhaitent être consultés sur ce programme.

(g) Model 1 - Subject 18 - "Computer science training"

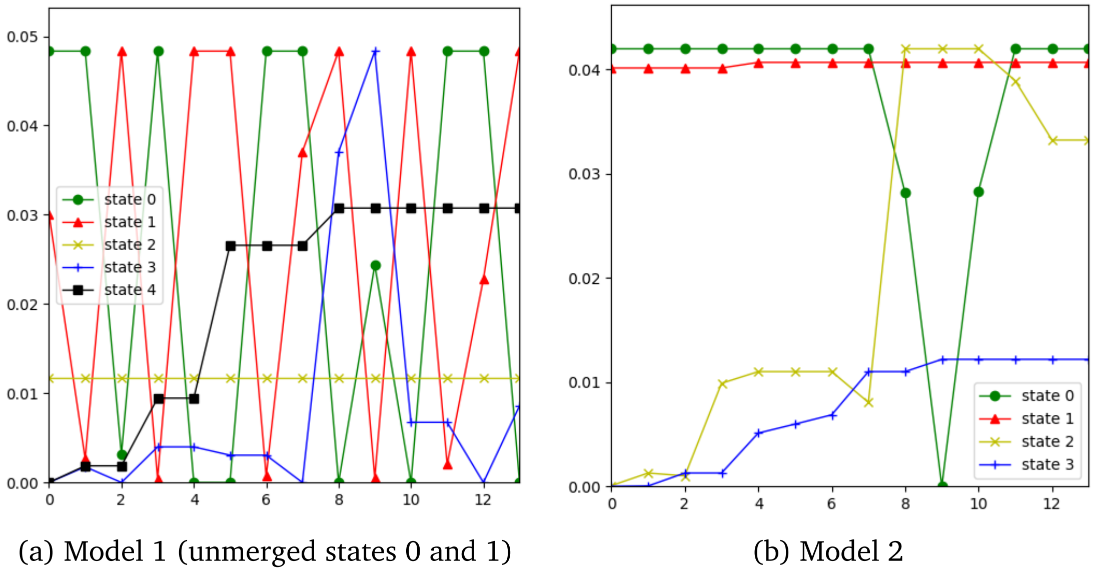
Le Conseil national des programmes et les observatoires académiques des pratiques pédagogiques seront chargés de définir les compétences communes dispensées au collège. Les enseignants souhaitent être consultés sur ce programme.

(h) Model 2 - Subject 18 - "Computer science training"

Figure 2.7: Scanpath restoration samples. Left column scanpaths are restored with model 1 while right ones are restored with model 2. Red: normal reading, green: speed reading, teal: information search, purple: slow confirmation.

In other words, the methodology is based on keeping the most likely state sequence conditionally to the observed sequence, restored by Viterbi, with the additional particularity of investigating the probability of all the states for a fixed time t and repeating the process for each $t \in [\![1, T]\!]$.

An application of this state profile exploration is showed in Figure 2.8 for model 1, 2.8a and model 2, 2.8b. Note that for model 1, state 0 and 1 could not be merged into one state due to software specifications. Besides, it should be clear that, the bigger the difference is at every time index between the max posterior and its candidate, the better it is. Hence, considering Figure 2.8a where both states 0 and 1 (green and red) are merged, we can see that candidates are not likely along the trial even though state 4 (in black) becomes a likely candidate after fixation 5. Nonetheless, for model 2, Figure 2.8b shows that along the entire trial, state 1 (in red) was a very likely candidate; a source of high local entropy for the corresponding trial.



(a) Model 1 (unmerged states 0 and 1)

(b) Model 2

Figure 2.8: Max posterior state probability over fixations for a scanpath restoration with model 1 and 2, subject 1 - "Planting flowers" - Unrelated text to the topic.

Choice of the model. Along the study, model 1 and 2 performed almost similarly in terms of quantitative criterion, parameter interpretability and scanpaths restorations. However, model 1 had a better BIC (-99798 vs -101804) which was shown to be more reliable than ICL for HSMM in the previous study. In terms of model parameters, model 1 had a shorter speed reading state and a lesser standard deviation on the duration of the normal reading state, making it a more stable state. Finally, restorations show small improper uses of states due to different constraints on the transition matrix. All these reasons lead us to **keep model 1** for the rest of the study.

Contribution summary. In this chapter, we proposed to identify and characterize reading strategies using HSMM. This process was rigorously tied together with a methodology proposing data selection, output process selection and model selection. Model Parameters were learned on the basis of two different and novel strategies: random EM initialization using the sequence breaking strategy as well as knowledge injection in the model parameters. This approach highlighted that a local maximizer with a large attraction domain might sometimes be preferable rather than a spurious local maximizer with a smaller attraction area. This statement is particularly true regarding the ICL criterion in the context of HSMM which was shown to underfit the data, indicating a preference to use the BIC, or another methodology such as cross validation criterion, Celeux and Durand (2008). Along the study, the two models learned with different strategies were opposed, showing high similarities and encouraging results in terms of interpretation. However, the retained model still needs to be assessed and interpreted using more diverse covariate such as eye-movement indicators, textual information or electroencephalograms, which is precisely the topic of the next chapter.

Chapter 3

A posteriori analysis of covariates

Contents

In this chapter, we propose to use the model retained in the previous chapter to perform scanpath segmentation and, a posteriori, analyze model covariates. The covariates considered are of different forms. In a first part, we discuss simple eye-movement related covariates (internal) such as fixation duration and saccades per reading strategy. In a second part, we use the text read by the participants to enrich reading strategies with semantic information acquisition indicators. Moreover, we explore inter-individual behavior differences regarding reading strategies. Finally, in a third and last part, we propose to use the concomitantly acquired multi-channel EEG and link it with brainwaves by the means of a time-frequency decomposition of the signal in order to relate the strategies with well known neural functions such as memory. Additionally, information diffusion is explored by inspecting inter-channel correlations.

1 Eye movement covariates (interval covariates)

In the previous chapter, we discussed the selection of the output process. We retained a process which we called *readmode* that is a truncated measure of the number of words skipped during an output saccade. We also found out that output processes used by Simola et al. (2008) for a similar task - the fixation duration, the saccade amplitude, its direction and a boolean holding the information if the currently fixated word had previously been or not - did not have any discriminant power on our data. In this section, we compute these indicators per reading strategy after performing a state restoration, which we call a posteriori analysis. We also relate strategies to the one based on reading rates (Carver, 1990).

Assessing reading rate. The reading rate is measured in words per minute (wpm). At a macro level, it can simply be measured by how far (in words) can a person go in a text in how much time. To measure the reading speed in a multi-sequence task like ours, where scanpath are also being segmented, we need to focus on the micro-measurements of the reading rate. At a micro level, we measure the number of words skipped in one saccade plus one, relating to the fixed word, with respect to time. The elapsed time corresponds to the duration of the previous fixation plus the duration of the outgoing saccade. Additionally, words that have already been read (or skipped in a previous saccade) do not increase reading speed. Therefore, if a strategy lasts three fixations, the reading rate is computed as the number of words skipped plus one during each outgoing saccade divided by the duration of the three fixations plus the associated outgoing saccades. However, since we work in a multi-sequence framework and that the average speed is not equal the average of the speeds, number of words and durations were summed over all reading strategies in all scanpaths, then divided.

Character increment / word increment ratio. We also provide a character increment per word increment ratio (CIWIR), a ratio between the number of characters skipped vs the number of words skipped in a saccade. The CIWIR measures either or not the words skipped in a reading strategy are semantically interesting. A low CIWIR means that words read in the reading strategy were short, which often corresponds to stop words, i.e. words which are very common and usually not specific to any topic. A high CIWIR means that words read were long and, by contrast, often meaningful regarding a specific topic.

Results. Eye movements covariates per reading strategies are reported in Table 3.1.

	Normal Reading	Speed Reading	Information Search	Slow Confirmation
Fixation duration (ms)	181 ± 68	178 ± 58	193 ± 58	190 ± 69
Saccade amplitude (px)	119 ± 101	153 ± 95	137 ± 105	143 ± 97
Reading speed (wpm)	353	615	500	280
CIWIR	3.7 ± 3.9	6.3 ± 4.7	5.5 ± 5.1	7.5 ± 5.9
Saccade directions				
Forward	0.74% (11924)	0.62% (6232)	0.51% (3189)	0.44% (3169)
Upward	0.01% (146)	0.02% (213)	0.01% (82)	0.09% (664)
Backward	0.13% (2003)	0.06% (659)	0.08% (499)	0.19% (1368)
Downward	0.12% (1941)	0.23% (2291)	0.21% (1277)	0.21% (1518)
Last	0% (1)	0.07% (684)	0.19% (1166)	0.07% (538)

Table 3.1: Eye-movement indicators per strategy.

Firstly, fixations tended to not last long in **normal reading** (NR) (181ms). Rayner (1998) indicated shorter fixations in association with easier tasks. This word-by-word reading strategy may be confirmed by short saccade amplitudes (119px) as well as the saccade directions, mostly aiming forward (74%) with seldom backward fixations (13%). A low CIWIR of 3.7 suggested that words skipped were essentially stop words. There were also few downward fixations (12%) pointing at the slowness of the process. The reading speed was 353 wpm, close to the 300 wpm suggested by Carver (1990). The strategy is never terminal which may be explained by the fact that is it a central process and therefore used as an initial strategy in information search tasks.

Secondly, **speed reading** was characterized by short fixations (178ms) as well as long saccade amplitudes (153px), symbolizing an easy task, the easiest. This is especially highlighted by the reading speed of 615 wpm, which can be compared to the scanning strategy of Carver that is used for lexical access. The high CIWIR (6.3) pointed out that words skipped were longer than average (5.3 characters per word). Hence, this possibly means that word skipped could be essential to the understanding of the text. The saccade directions were mostly forward and downward (total 85%) promoting a rather fast forward behavior, which is contrary to the scanning state found by Simola et al. (2008) where directions were random (≈25% each). We explain this phenomenon by the difference of the tasks that was asked to readers. In our study, the global saccade behavior was mostly progressions with very few regressions.

Thirdly, **information search** had long fixations (193ms), average saccade amplitudes (137ms) and CIWIR (3.7) but a quick reading speed (500 wpm). We make the analogy with the skimming strategy of Carver, achieved at 450wpm, that consists in semantic access to words, and that gathers just enough information to know what the text it about. It has similar saccade directions as speed reading but with less forward saccades and more last fixations.

Lastly, **slow confirmation** was related to long fixations (190ms) and rather long saccade amplitudes (143px). The reading speed was slow (280wpm), which is explained by mostly re-reading as we do take into account re-read words in the computation of the reading speed. It is slower than normal reading and therefore integrates the ability of learning and answering text comprehension questions as presented by Carver. We relate this to the (slow) decision making process since it is mostly a terminal state. Most of the upward saccades but also a lot of backward saccades were achieved in this state, characteristic of re-reading.

Conclusion. Even though reading strategies were not discriminated with a HSMM and the low-level output processes presented in the study of Simola et al. (2008), we found out that it could be done a posteriori by using a much high-level different output process. The fact that these low-level variables did not have a (semi)-Markovian dynamics is a possible explanation of this successful a posteriori segmentation. This study also corroborates the semantic given to each reading strategy in comparison with other similar studies.

2 Text and Subjects (external covariates)

2.1 Types of readers

Uses of reading strategies. The automaton presented in Figure 2.6a shows a broad range of possibilities concerning strategy usages. For example, the process may start in normal reading (NR) then go to information search (IS), go back to NR to finish in slow confirmation (SC). It may also simply start and finish in NR. What strategies are really used in practice? Are there different clusters of subjects? Both these questions might be answered by performing a factor correspondence analysis (FCA).

Factor Correspondence Analysis. The FCA proposed by Benzécri et al. (1973) is a diagonalization of the contingency table, a matrix representing the factor occurrences of two categorical variables. Each point is then represented in a new space using eigenvectors. In this space, information is conserved and axis are hierarchized by contribution to the inertia in the data.

Results. FCA was performed on the subject-reading strategy occurrence matrix. The projection of subjects and strategies on the first two axis is shown by Figure 3.1. Axis 1 holds most of the inertia, 74.4%, and contrasts readers using normal reading (NR) and information search (IS) vs those using speed reading (SR) and slow confirmation (SC). The second axis comprises 25.6% of the variance and brings into opposition SR vs NR and IS but also SC vs NR and IS. Subjects who are close to a reading strategy tended to use it more often. For example subject 4 mainly used SC. On the contrary, subjects located in the center such as subjects 17, 10, 8 can be seen as more versatile.

Conclusion. This study shows a gradient between fast and careful readers which suggests that readers may be clustered accordingly. Moreover, it also puts forward that not all sequences are independent and identically distributed, a hypothesis in the modeling proposed in Chapter 2.

2.2 Types of texts

Formerly, the results presented in Chapter 2 did not explicitly take into account the effect of the type of texts. In section 1.2, we discussed that the experiments were run on three types of texts: those highly related (HR) to the topic, moderately related to

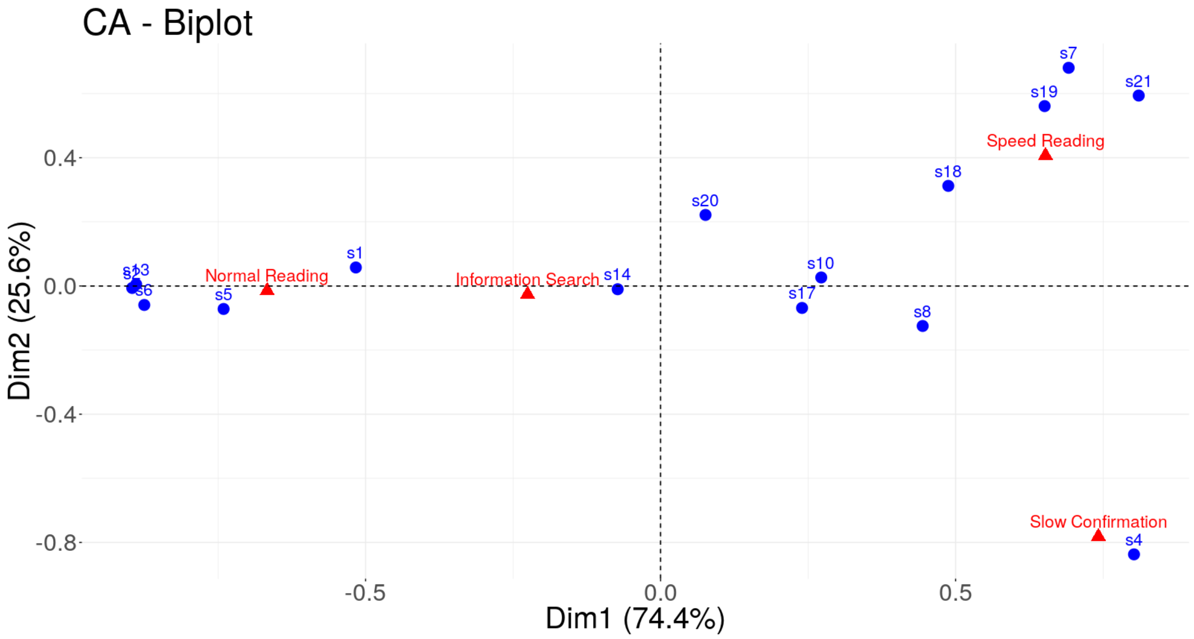


Figure 3.1: Factor Correspondence Analysis of the strategy usage per subject.

the topic (MR) and unrelated (UR) to the topic. 60 of each type were presented to each subject. Moreover, we chose to split the class HR into HR and HR+, where HR+ directly contains words in the topic. HR+ texts should therefore result in an easier task. So is, UR texts in which incongruent words to the topic are easier to spot. In this section, we perform another a posteriori analysis in order to assess quantitatively the effect of the text type and if reading strategy are used differently in the different type of texts.

Semantic represent of words. UR,MR and HR texts were clustered in a previous study, Frey et al. (2013), by using LSA to provide vector representations of words. LSA was trained on a 24 million-word French corpus composed of all the articles published in the newspaper *Le Monde* in 1999. A measure of **cumulated cosine** was used to control the semantic relatedness of the texts to the goals. The cumulated cosine is defined as the cosine similarity between the sum of all words in the text and the sum of all words in the topic. The cosine similarity simply measures the cosine of the angle between two vectors, i.e. the dot product divided by the magnitude of the vectors. Hence, a cosine of 1 shows a great semantic similarity between words while a cosine of 0 means that words are unrelated.

Results on text type indicators. Indicators per text type are presented in table 3.2.

	Unrelated	Highly Related	Highly Related +	Moderately Related
cumulated cosine	< 0.1	> 0.4	> 0.4	$0.15 < . < 0.3$
#trials	802	443	360	785
average #fixations	14.3 ± 7.9	15.5 ± 7.5	15.8 ± 8.6	20.1 ± 8.2
Reading speed (wpm)	485	413	412	399
strategy proportions (#fixations)				
NR	.39 (4492)	.39 (2559)	.40 (2287)	.42 (6677)
SR	.27 (3153)	.29 (1919)	.28 (1557)	.22 (3451)
IS	.18 (3153)	.16 (1093)	.16 (927)	.13 (2041)
SC	.15 (1706)	.16 (1066)	.16 (909)	.23 (3576)
average instantaneous cosine per strategy				
NR	.00 ± .05	.18± .25	.22± .30	.10± .18
SR	.00 ± .06	.20 ± .26	.24 ± .31	.09 ± .17
IS	.00± .07	.25± .29	.27± .32	.11± .18
SC	.00± .06	.20± .26	.22± .31	.09± .16

Table 3.2: Text type indicators.

MR texts had much more fixations on average (20.1) than others. Participants performed one less fixation in UR texts than HR texts, but also had a higher reading speed, suggesting that the task was easier with UR texts containing incongruent words to the topic. HR+ texts did not seem easier in terms of number of fixations, but also reading speed (≈ 410wpm) rather than HR texts, even though they contained words present in the topic. The reading speed was also slightly less for MR texts than HR/HR+.

The second section of the table shows the reading strategy proportions in number of fixations per text. It seems that strategies were used similarly in texts UR, HR and HR+ with almost equal proportions. The main difference arose from text MR, where much more time was spent in SC (23% vs 16%) rather than SR (22% vs 28%).

The last section of the table shows the per-trial-averaged instantaneous cosine between fixed words and the topic of the text for each text type and each reading strategy. For every trial, only fixed words are taken into account through the instantaneous cosine, since semantic information is acquired during fixations and not saccades (Rayner, 1998). A low instantaneous cosine is better for UR texts, whereas a higher instantaneous cosine is better for HR/HR+ texts. It has no special meaning for MR texts since their relation to the topic is usually fuzzy. Globally, results were similar for text types HR and HR+: the IS strategy had the highest instantaneous cosine which means that target words were more often fixed in this reading strategy. Then, SR had a higher instantaneous cosine in HR+ texts than NR/SC (.24 vs .22), whereas NR was less efficient in HR texts (0.18) than other strategies. It should be noticed that the results presented also show high standard deviation.

Assessing state transitions instants. Let us recall the hypothesis of this study: subjects take either positive decisions by detecting target words or negative decisions by detecting incongruent words. A key study aimed at assessing either or not state (reading strategy) transitions occurred around target (for texts HR) and incongruent (for texts UR) words based on the following question: are target and incongruent words (keywords) the triggers of reading strategy changes ? And how quick are the changes triggered according to the text type ? In order to answer these questions, a procedure to detect keywords automatically was first designed.

A new representation of words. In practise, we failed at detecting keywords automatically with a LSA representation, notably incongruent words. This result might be explained by the poor ability of LSA to find good representations for words which are not frequent in the vocabulary and for words which might be out of the vocabulary. To face this issue, we instead used Facebook's **fastText** word representations. FastText is based on a recent neural probabilistic model, namely word2vec, proposed by Mikolov et al. (2013a,b). This method learns an embedding by predicting the surrounding words given the context. The context is the current word. Several extensions were then proposed to come up with memory-efficient representations (Joulin et al., 2016a,b). FastText also present the main advantage of decomposing each word into a bag of n-gram characters and then creates sub-word features related to part of speech, or semantic. Concretely, even words that are less frequent or out of vocabulary get a good representation by analogy to their neighbors. Mikolov et al. (2018); Grave et al. (2018) provided pretrained word vectors on tremendous data, such as Wikipedia and Common Crawl. We used their word vectors in French. Moreover, we used the source code publicly available presented in Bojanowski et al. (2017) to find representations of words out of vocabulary.

Experimental procedure. Target words were detected using fastText. For HR and HR+ texts, we kept the two words which had the highest instantaneous cosine with the topic. For UR texts, the two less related words to the topic weighted by their frequency were kept. For MR texts, the most highly related and the least related words were kept with the purpose of finding words (related or not) which contributed to the decision making. Since in section 3.3 we have discussed the uncertainty of state restorations and transition instant, we measured the number of fixations between the transition instant and the target word to assess the accuracy of the transition. The lesser the distance between the transition and to the target word, the better is it. The minimal distance between the two words and the transition instants was kept.

Results. The results are presented in figure 3.2. Each figure represents the distance (in number of fixations) between the transition word and the reading strategy preceding the transition for each text type. Each point therefore represents a frequency and the regression line per text is shown. A regression with a low slope coefficient typically shows that transitions occurred more frequently around keywords. This effect is particularly noticeable for transition occurring from NR strategy in texts UR, HR and HR+, which seems to point out that beginning with a NR strategy is efficient to find out keywords in an information search task. The MR slope coefficient is almost 0, pointing that strategy transitions are not triggered by keywords in MR texts. The slope coefficient is higher for texts HR and HR+ when transiting from IS strategies than for UR texts. And this effect is even more present when transiting from SR. It shows that SR is particularly adapted for the easiest task.

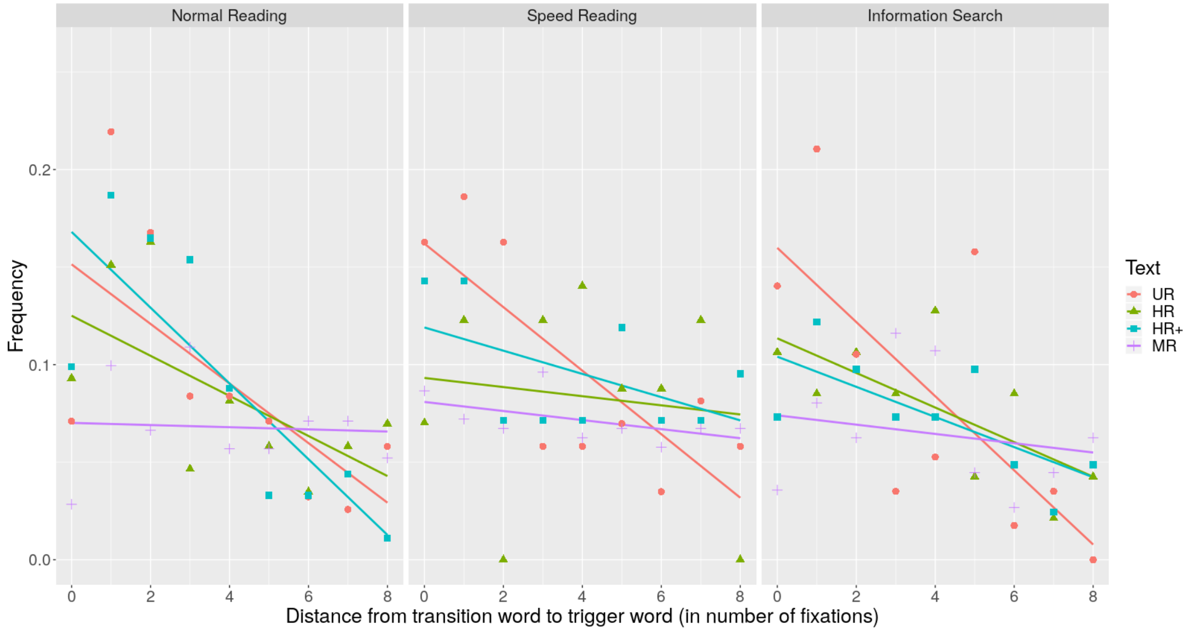


Figure 3.2: Frequencies of the distance between transition word to trigger word in number of fixations.

Conclusion. The previous studies displayed a gradient in complexity between texts UR, HR, HR+, MR characterized by:

- number of fixations,
- reading speed,
- time spent in slow confirmation vs speed reading,
- number of strategy changes,
- effect of trigger words on strategy changes.

Text type differences are shown by several indicators but statistical model (parameters) are still encapsulating these differences and variations.

Different types of text ? In this study, we proposed to split HR into HR and HR+ texts, but it could be interesting to differentiate subject behaviors according to a finer clustering. For example, texts could be clustered using a signal representing the evolution of the semantic relatedness of the text to the topic. A preliminary study has shown that HR texts could be clustered into three profiles: the one having their signal increasing by step (keywords with high instantaneous cosine), the one having their signal increasing with a slope (all words are slightly related) and the one with a saw signal (words are sometimes not related). Soheily-Khah et al. (2016) notably proposed a kernel kmeans method for time series clustering.

Including random effects in the model. In order to deal with the subject and/or text effect, an interesting perspective is to take into account covariates directly in the model and show how they affect parameters, see Chaubert-Pereira (2008); Chaubert-Pereira et al. (2008, 2010) for semi-Markov switching linear mixed models, and Peyhardi et al. (2016) for linear models with categorical response variable. Another possibilities is to model a mixture of HSMM but leads to high increase in terms of number of parameters whereas mixed effect models use some tied parameters.

3 EEGs (external covariates)

3.1 Introduction to EEG analysis

The eye–mind link assumption suggests that the location of an observer's gaze partially reflects what is being processed in his or her mind at that time (Reichle and Reingold, 2013). At a **micro scale**, eye movements represent natural markers for time-locking the ongoing neural activity with respect to a eye movement events such as fixations. Such technique is called eye-fixation related potential (EFRP) (Dimigen et al., 2011) and may be seen at more ecological or natural way to analyze cognitive processes rather than the other well known method called event-related potentials (ERP) which studies brain response with respect to a precise stimuli, see Woodman (2010) for reviews on ERP studies. However, both these techniques rely on time-locking signals and averaging to bring out specific patterns (Luck, 2014).

Moreover, EFRP is an increasingly popular technique and at the moment, little is known about reading in more complex settings such as free text exploration (Dimigen et al., 2011). It relies on the investigation of text comprehension in online task which could lead to the emergence of specific cognitive processes, Leu et al. (2015). Ecological studies have already been tackled in the context of EFRP such as the assessment of working memory with respect to a reading and decide task or a reading and memorize task, both involving different cognitive processes (Frey et al., 2018).

Another method aims at studying brain oscillations on the frequency domain where frequency ranges are related to brain waves. For example, processes related with short-term (episodic) memory may be observed by an increase in the theta band (4-7 Hz), possibly in an anterior limbic system, whereas processes related with long-term (semantic) memory are characterized by a decrease or suppression of power in the upper alpha band (8-12 Hz) in a posterior-thalamic system (Klimesch, 1996). Sauseng et al. (2005) stated that memory is an extremely distributed system with long term memory primarily located in posterior cortices and accessed from prefontal regions. Hanslmayr et al. (2011) also found out alpha oscillations in temporal attention. Seidkhani et al. (2017) observed memory encoding and restitution differences observed in alpha band using a similar wavelet and network-based method. Alpha frequency has been found to be under top-down control to increase or decrease the temporal resolution of visual perception Wutz et al. (2018).

In this study, we make the hypothesis that, at a **macro scale**, the eye-movement semi-Markovian dynamics may also be used to segment brain activity into contrasted reading strategies in terms of EEG patterns. EEGs are time-locked with respect to phases to extract the cognitive process related to reading strategies. Signals are not aggregated with mean but with wavelet cross-correlation between channels during a given phase and a given trial. Wavelet cross-correlations are then aggregated with weighted average. Hence we do not aim to study an eye-fixation relation potential triggered by a given stimuli but a general change of information diffusion in brain through differences of correlations.

EEGs turned out to be too noisy for observing specific patterns on short signals. Instead we used a time-frequency decomposition called maximal overlap discrete wavelet transform (MODWT), Percival and Walden (2006). MODWT is a non orthogonal wavelet transform, compared to the classical discrete wavelet transform (DWT). MODWT is also invariant by translation. Its coefficients may be computed by the pyramid algorithm Mallat (1999). We used MODWT because their estimators of wavelet correlation are supperior to DWT's, Whitcher et al. (2000).

3.2 Introduction to MODWT

In this section, we summarize the work of Whitcher et al. (2000) who provided an unbiased estimator of wavelet cross-correlation and the corresponding confidence interval.

Let us define **X**, a time series of length T. Let

$$\{h_{j,l}\}_{l=0}^{L_j-1}$$

be the wavelet filter (high-pass filter) and

$$\{g_{j,l}\}_{l=0}^{L_j-1}$$

the scale filter (low-pass filter), where $L_j = (2^j - 1)(L-1)+1$ denotes the width of the filter at j-th level and L, the width of the initial filter. The associated MODWT scale and wavelet filters at scale j are respectively

$$\widetilde{h}_{j,l} = \frac{h_{j,l}}{2^{\frac{j}{2}}}$$

and

$$\widetilde{g}_{j,l} = \frac{g_{j,l}}{2^{\frac{j}{2}}}$$

with identical width L_j. The MODWT wavelet coefficients noted $\mathbf{W}_j$, a vector of size T are defined as

$$W_{j,t}^{(X)} = \sum_{l=0}^{L_j-1} \widetilde{h}_{j,l} X_{t-l} \mod T$$

and similarly the scaling coefficients, noted $\mathbf{V}_j$ a vector of size T, are defined as

$$V_{j,t}^{(X)} = \sum_{l=0}^{L_j-1} \widetilde{g}_{j,l} X_{t-l} \mod T.$$

The MODWT has the following property of the energy decomposition

$$\|\mathbf{X}\|^2 = \sum_{j=1}^{J} \|\mathbf{W_j}\|^2 + \|\mathbf{V_J}\|^2,$$

where J is the total number of scales. In other words, MODWT decomposes the variance without loss of information.

Wavelet estimator of the cross correlation. Let us denote $\mathbf{X}$ and $\mathbf{Y}$ two time series that are realizations of size T of Gaussian processes with stationary increments. For $T > L_j$, an unbiased estimator for the covariance at a given scale between $\mathbf{X}$ and $\mathbf{Y}$ is:

$$\gamma_{XY}(\lambda_j) = \frac{1}{T_j} \sum_{L_j-1}^{T-1} W_{j,l}^{(X)} W_{j,l}^{(Y)}$$

for scale $\lambda_j = 2^{j-1}$, and $T_j = T - L_j + 1$. And estimator for wavelet correlation is then

$$\widetilde{\rho}_{XY}(\lambda_j) = \frac{\gamma_{XY}(\lambda_j)}{\nu_X(\lambda_j)\nu_Y(\lambda_j)},$$

with $\nu_X^2 = Var(\mathbf{W}_j^{(X)})/2\lambda_j$ and $\nu_Y = Var(\mathbf{W}_j^{(Y)})/2\lambda_j$ the wavelet variance time series $\mathbf{X}$ and $\mathbf{Y}$ respectively.

Wavelet confidence interval for the cross correlation. Under the hypothesis that $L > 2d$, where d is the max of the orders $\mathbf{X}$ and $\mathbf{Y}$, and that the wavelet coefficients $\mathbf{W}_j^{(X)}$ and $\mathbf{W}_j^{(Y)}$ is a bivariate Gaussian process weakly stationary with square integrable autospectra then the wavelet cross-correlation $\rho_{XY}(\lambda_j)$ is asymptotically normal and unbiased. An approximate confidence interval for the wavelet correlation is therefore

$$IC_\alpha[\rho_{XY}(\lambda_j)] = \left[tanh\left\{ h[\widetilde{\rho}_{XY}(\lambda_j)] - \frac{\phi^{-1}(1-p)}{\sqrt{\hat{T}_j - 3}} \right\}, tanh\left\{ h[\widetilde{\rho}_{XY}(\lambda_j)] + \frac{\phi^{-1}(1-p)}{\sqrt{\hat{T}_j - 3}} \right\} \right], \tag{3.1}$$

with $h(\rho) = tanh^{-1}(\rho)$ being Fisher's z transformation and improves the quality of the confidence interval for small sample sizes, $\hat{T}_j = T_j - L'_j$ and $L'_j = \lceil (L-2)(1-2^{-J}) \rceil$, the number of MODWT coefficients at scale λ_j.

3.3 Methodology

Data acquisition. Electrodes were referenced to head (FCz—ground:AFz). EEG data were amplified with BrainAmp system, sampled at 1000 Hz, and then filtered with a 250 Hz low-pass filter Frey et al. (2013). The montage is provided in Figure 3.3. Each trial had a corresponding sequence of 10 seconds and was truncated if the trial was exceeding this duration. 180 ms of acquisition before each trial is also available. In total, we had 2390 trials, the same number of eye-movement sequences.

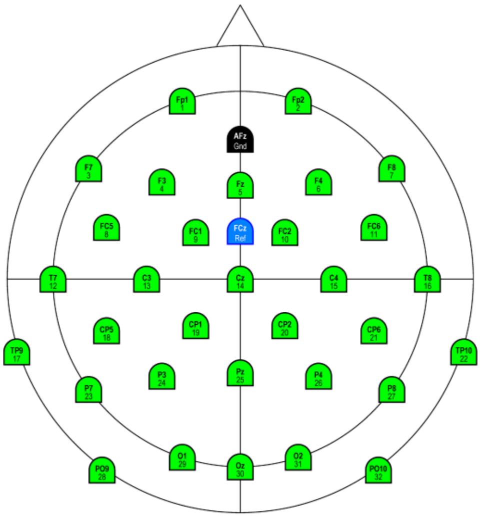


Figure 3.3: EEG montage.

Preprocessing. A first pass of preprocessing was performed in Frey et al. (2013). On top of that, we ran an automatic channel and/or trial rejection method called autoreject Jas et al. (2016, 2017). Autoreject aims at finding by cross-validation the optimal pic-to-pic threshold. Bad channels were then interpolated using spherical spline interpolation with the python software MNE Gramfort et al. (2013, 2014). Finally, we chose to remove the baseline activity for each trial on the time domain under the gain model hypothesis, i.e. we normalized the entire EEG trial using pre-trial start data (duration 180ms) as "resting state" activity Grandchamp and Delorme (2011).

Wavelet analysis. We used MODWT with LA(8) wavelet filter to decompose each trial on the time-frequency domain. The goal is to decompose pairwise correlations patterns that might not be visible on time domain. The correspondence of the wavelet scale to frequency brain and therefore brain wave is shown in table 3.3. Brain oscillations are a widely studied area and the wavelet scale to neural oscillation equivalence should give some light to our results.

Wavelet scale	Wavelet Frequency (Hz)	Brain wave	Brain wave frequency (Hz)
1	256-512		
2	128-256		
3 4	64-128 32-64	γ	32-100Hz
5	16-32	β	12.5-30
6	8-16	α	8-12
7	4-8	θ	4-7

Table 3.3: Wavelet scales, their equivalence in the frequency domain, and their corresponding brain waves.

Correlation analysis. Correlation analysis is an efficient way to analyze information diffusion and activated regions during a given task. As a matter of fact, a highly correlated region may be seen as an entire area working concomitantly. To this end, Bassett and Bullmore (2006); Achard et al. (2006) proposed to use small-world brain networks in fMRI, which relies on graph theory properties (Strogatz, 2001). Small-world networks were shown to have greater local interconnectivity with inferior mean path length between any pair of node than a random network (Watts and Strogatz, 1998). Small-world networks have also been used with EEG data Ferri et al. (2007); Smit et al. (2008).

In Chapter 1 section 2.1, we defined a graph G in the context of dynamic Bayesian network to be a tuple of vertices V and non-oriented edges E, and so $G = (V, E)$. The edges were previously oriented while they are not in a small-world network. Given vertices and edges, the adjacency (square and symmetrical) matrix of the graph can be obtained by setting an element to 1 if two vertices are linked through an edge. The degree of a node is the total number of edges connected to it. The shortest path length between all nodes may also be computed via the well-known Dijkstra algorithm.

Small-world networks allow constructing a sparse anatomical representation of a graph given significant inter-channel correlations, presented as an adjacency matrix.

To this end, each pair of correlations is tested, and significance is tested with the confidence interval provided by equation (3.1) (Whitcher et al., 2000). For each pair of correlations, if it is significant a 1 is set for the same pair in the adjacency matrix, otherwise 0. Achard et al. (2006) proposed to choose the minimal correlation threshold R for the test such that the mean degree of the graph corresponds to the equilibrium of the small network property i.e. the mean degree is equal to the log of the number of nodes (channels). The mean degree is a measure of connectivity in the graph and is the average number of incident for all vertices. In our case, $R = 3.4$.

The correlation threshold R is chosen according to the wavelet scale having the highest amount of significant correlations, and hence the highest amount of edges.

Anatomical representation of graphs and channels. The anatomical graph is constructed given the adjacency matrix of significant correlations and the channel positions, thus an arc is a significant correlation. The maximal number of possible pairs with 30 channels is 435. We used the brainwaver R package to represent sagittal and top view.

The procedure is summarized in figure 3.4 for a given wavelet scale and a given reading strategy.

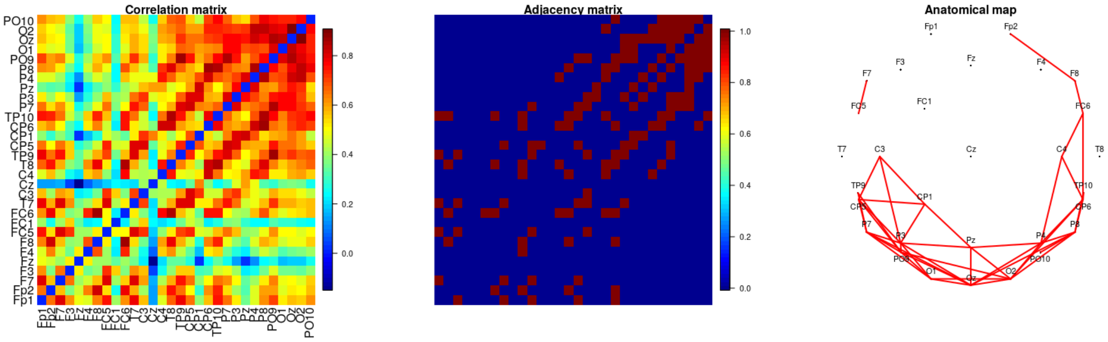


Figure 3.4: Network construction methodology: Correlation matrix, adjacency matrix containing significant correlations and corresponding anatomical graph for a given scale and a given reading strategy.

Correlations per phase. The specificity of our task lies on the decomposition of a trial into phases, i.e. reading strategies. To this end, we computed the wavelet coefficients of each channel for each trial and segmented the wavelet coefficients with respect to phase changes. We then computed cross-correlations for all trials, for a given a phase, before aggregating the correlations per trial with a weighted average, the weight corresponding to the length of the phase.

3.4 Results

Globally, EEG activity was the most salient for wavelet scales 6 (α) and 7 (θ), corresponding to frequency ranges 4-8Hz and 8-16Hz which we relate to theta and alpha bands respectively. This information can be seen on figure 3.5a which represents the mean degree as a function of the correlation threshold R for each scale. The higher the mean degree, the higher the number of arcs, the higher amount of significant correlations there is. The dotted constant line $y = 3.4$ represent the threshold under which small-world properties are not estimable. We hence choose the correlation threshold such that the mean degree is equal to 3.4, a total of 102 edges. This threshold turned out to be around 0.54 for scale 6 (α) and 0.53 for scale 7 (θ). This information can be interpreted as "102 correlations are significantly superior to 0.54 for scale 6". Similarly, 102 correlations are significantly superior to 0.50 for scale 5 (β).

The same information is shown on figure 3.5b, decomposed per phase. It can be seen that normal reading strategy (NR) is equally salient on scales 6 (α) and 7 (θ) and 102 correlations are significant at a threshold $R = 0.58$. Information search (IS) and speed reading (SR) strategy both have a very similar amount of correlations and may be observed on the same scales: 5 (β),6 (α) and 7 (θ) (mainly 6) with 102 correlations at approximately $R = 0.50$. Finally, slow confirmation (SC) is more similar to normal reading and correlations may be equally observed on scales 6 and 7 and a threshold of $R = 0.55$. In order to not complicate interpretability with different threshold per phases, we chose to represent anatomical maps with the same threshold for all reading strategies, $R = 0.54$, corresponding to the general threshold, for which scale 6 has 102 edges. Therefore, for this threshold, there were more correlations in NR and SC rather than IS and SR. We have previously seen that both these sets of strategy were notably contrasted by reading speeds. NR and SC are slower strategies than IS and SR.

Anatomical maps thresholded at $R = 0.54$ for scale 6 (α) are shown on figure 3.6. For each strategy, on the left: the sagittal view, on the right: the top view. For NR strategy, the occipital and parietal regions are highly correlated on each side (left/right). Temporal regions are also correlated with both frontal and parietal regions on each side. Additionally, left and right parietal regions are also very connected. IS and SR strategies information diffusion is mainly localized in the occipital area but also in the parietal regions. Both frontal and temporal regions are very less connected. Finally, SC is more similar to NR with a wide variety of local connections in each area but also towards neighboring areas. The main difference is the lesser amount of connections between the parietal left and right regions. We may notice that the temporal area is more connected with the frontal on the right than of the left side.

Anatomical maps thresholded at $R = 0.54$ for wavelet scale 7 (θ) are shown on figure 3.7. Correlations for NR strategy are almost identical on scales 6 (α) and 7. There are just a few less inter-regional arcs but intra-regional edges remains the same. Both IS and SR strategies have less connections on this scale between occipital and parietal regions. SC's map is identical to scale 6.

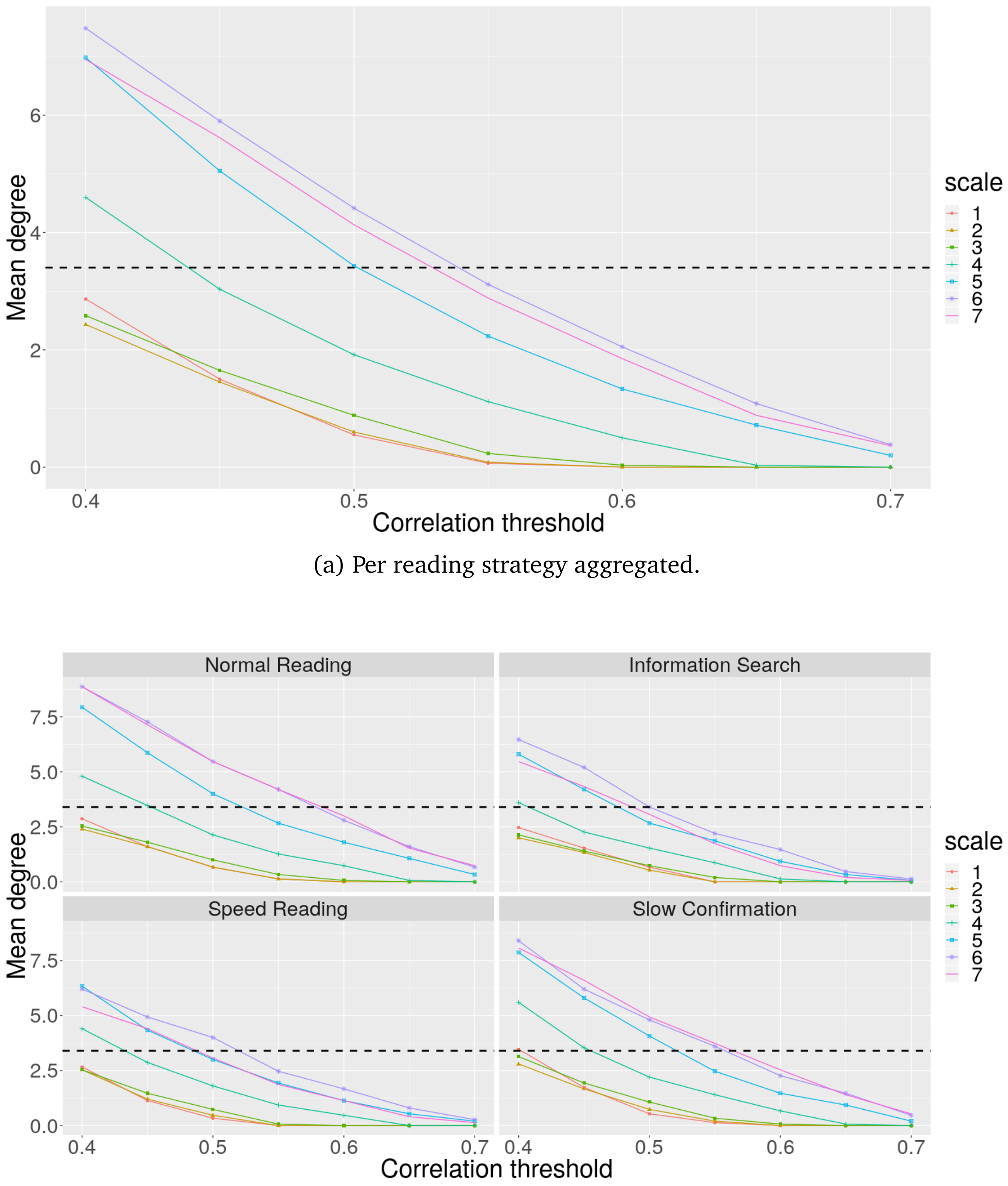
More anatomical maps on other scales but also correlation matrices are provided in Appendix B.

3.5 Discussion

Early conclusion. Strategies which requires a deeper sentential integration and nearly memorization (NR and SC) seem to involve more information diffusion than quicker reading strategies (IS and SR) is both bands theta and alpha. This difference of information diffusion is mainly characterized by more intra-connections in the temporal regions but also inter-connections with both frontal and occipital regions. The right hemisphere seems to be slightly more activated and is contrary to the study of Nagel et al. (2013), where authors observed left hemispheric lateralization for verbal working memory as well as right hemisphere lateralization for spatial working memory.

Thresholding the graph. This work is still ongoing and should be verified with a highest amount of indicators of small-world properties such as clustering ratio, path length ratio, clustering-path length ratio (Bassett and Bullmore, 2006; Seidkhani et al., 2017).

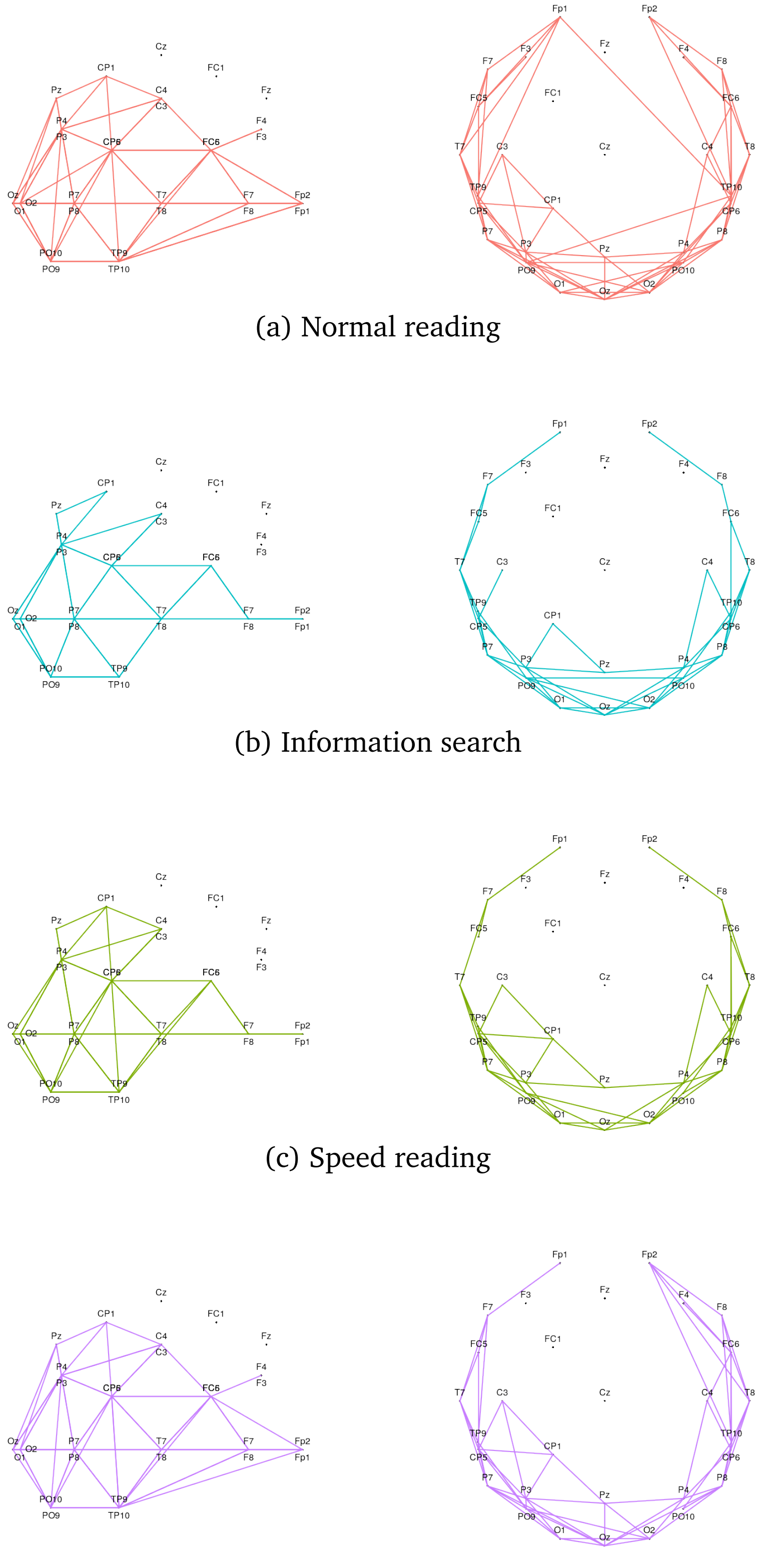
Network random error. We performed a total of 465 dependent hypothesis testing. Therefore, we induced an error in the network. We tried to apply a Bonferroni correction for multiple test but it turned out to be too penalizing for high scales and we did not find any significant correlations even though they were expected. The need of a test for dependent hypothesis testing is primary since a spurious correlation might engender much more, its is a current topic of research. To handle conditional



(a) Per reading strategy aggregated.

(b) Per reading strategy

Figure 3.5: Mean path length of wavelet networks for given a correlation threshold.



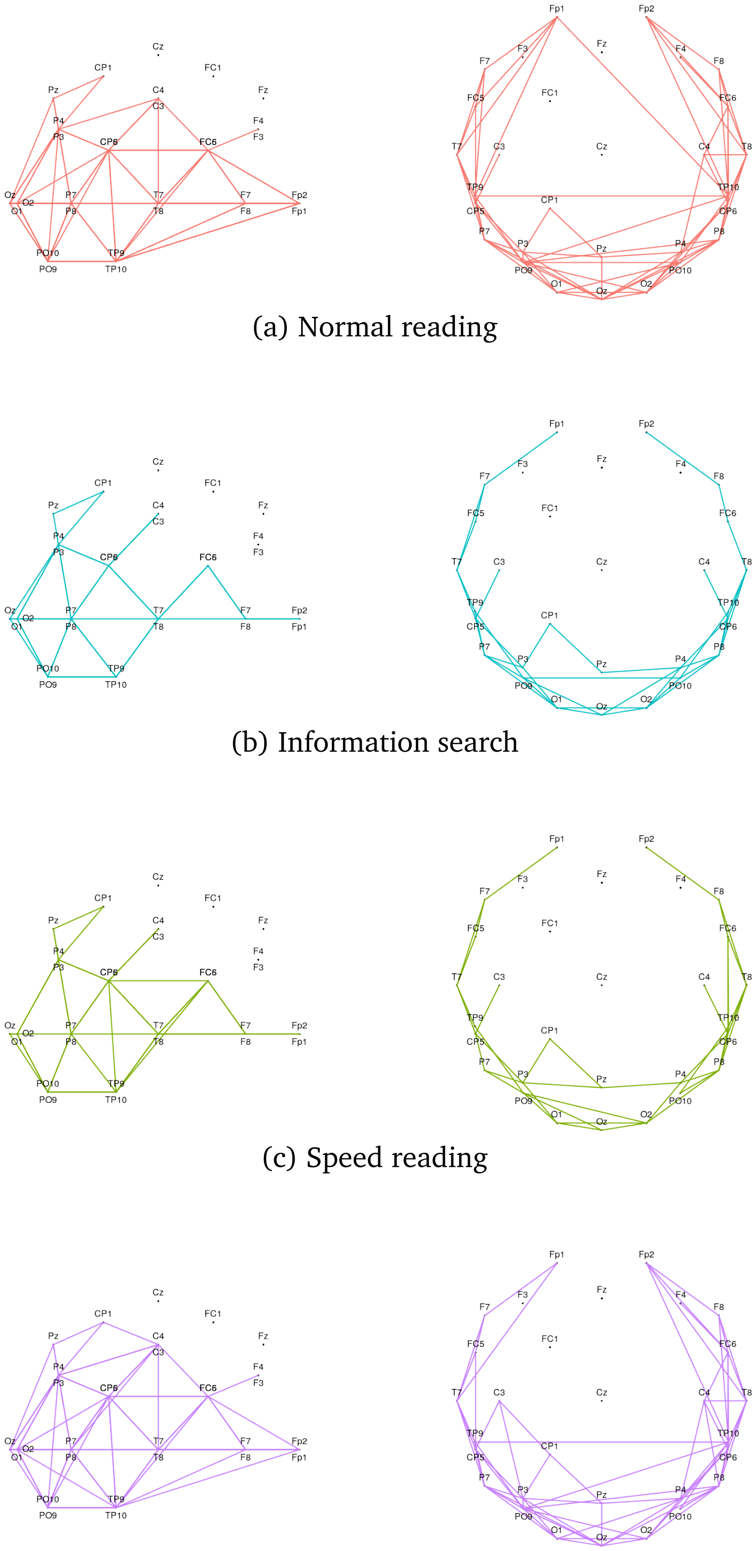
(a) Normal reading

(b) Information search

(c) Speed reading

(d) Slow confirmation

Figure 3.6: Anatomical maps (left: sagittal view, right: top view) per reading strategy for wavelet scale 6 (α band) with thresholded covariance at 0.54. Left map is a sagittal view, right map is a top view.



(a) Normal reading

(b) Information search

(c) Speed reading

(d) Slow confirmation

Figure 3.7: Anatomical maps (left: sagittal view, right: top view) per reading strategy for wavelet scale 7 (θ band) with thresholded covariance at 0.54. Left map is a sagittal view, right map is a top view.

dependence, Barabâsi et al. (2002) notably provided a sampling method to test the effect of correlations one by one on the rest of the network. Park et al. (2014) proposed a new framework to analyze wavelet partial coherence which models direct linear dependence between a pair of signals and therefore removes to linear effect of other observed signals.

Phases overlap. One of the drawback of the usage of MODWT on segmented data lies on the overlap created by downsampling when using MODWT on high scales (low frequency). Indeed, the higher the scale, the larger the filter and the more neighboring information is used. This has the effect of creating an overlap between the signal related to different phases around transitions.

Study of the variance. In an unfruitful study, we tried to perform a wavelet analysis of the variance. Even though, the study highlighted different variance patterns per reading strategies, this variance did not seem superior compared to the variance related to subjects. In an ongoing experiment, we are trying to quantify the contribution to the wavelet variance of different effects such as texts and subjects by using mixed effect models.

EM-EEG Delay. Finally, it is known that the brain activity is a delayed consequence of what the eyes read, the brain then guide the eyes in return with the information acquired Frey et al. (2013). Figure 3.8 shows a salient delay in the brain activity regarding what words are being fixed at what time. Our goal is then to incorporate both eye-movement and EEG data in a single model, taking into account the delay in order to reduce the uncertainty of the segmentation that was discussed in Chapter 2. This topic is the object of Chapter 4.

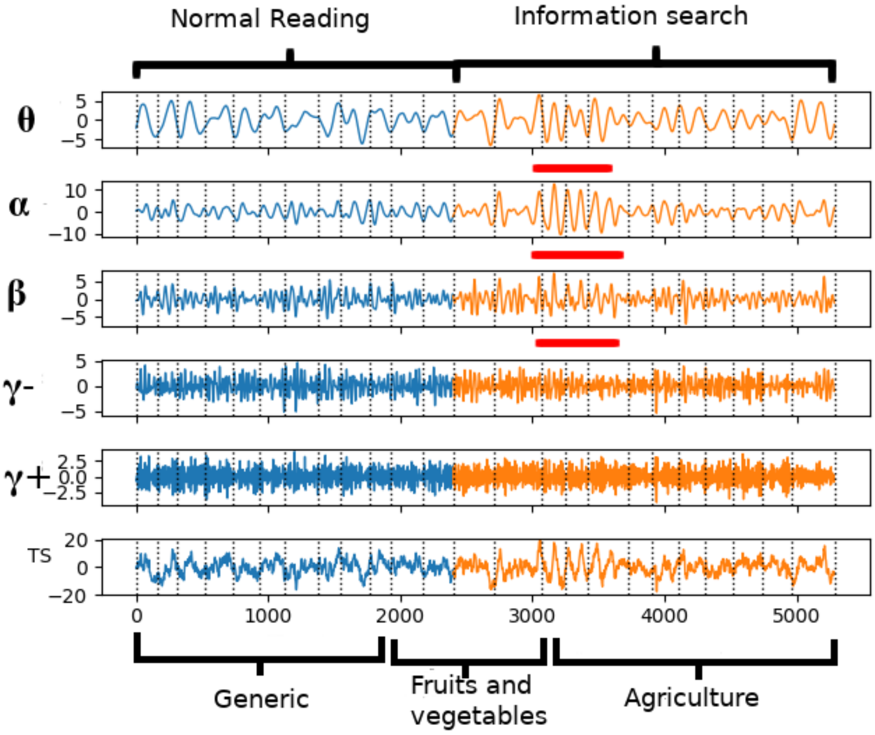


Figure 3.8: EEG recording and its wavelet decomposition given at bands $\theta, \alpha, \beta, \gamma-, \gamma+$ for a given channel and a given trial on unrelated text (UR) "economic growth". The vocabulary read is first generic and then relates to fruits, vegetables and agriculture. The fruits and vegetable lexical field seems to involve a delayed change of activity (underlined in read) in bands θ, α and β.

Chapter 4

Coupling eye-movement and EEG data with Asynchronous Heterogeneous Hidden semi-Markov Models

Contents

In this chapter, we propose to extend the hidden semi-Markov model to segment two (sets of) signals with the two following characteristics: they are asynchronous and heterogeneous. The former supposes that the first signal drives the second one with an additional random delay, which also induces delayed semi-Markovian regimes. The latter proposes to take into account the huge differences in sampling rates of the output processes in model parameter learning. This difference in sampling rates is motivated by a first signal that is a discretized measure over a non-constant time whereas the second one is considered multivariate and continuous. We name the model **Asynchronous Heterogeneous Hidden Semi-Markov Model** (AHHSM). Not only is AHHSMM suited for signals such as eye movements and EEGs but it may also be applied to a broader range of data types such as image and sound.

1 Model description

1.1 Model specifications

Up to this point, we have shown how to segment eye movements into reading strategies by extracting eye-movement features in Chapter 2, based on previous results from the study of Simola et al. (2008). In chapter 3, we used EEGs as model covariates to better interpret phases related to eye-movement from a cognitive point of view. The analyses revealed contrasted EEG patterns per phase, which could be related to changes in cognitive processes. Eye-movement events were also synchronized (time-locked) with EEGs which were segmented a posteriori into reading strategies, highlighting changes in channel covariance per phase and wavelet scale.

There have been plenty of models in the literature which aimed at modeling EEGs with hidden Markov model. Bashashati et al. (2007) proposed a general survey which notably contains an inventory on statistical EEG modeling. Obermaier et al. (2001a,b) proposed to classify EEG signals with HMMs, Obermaier et al. (2001c) proposed to measure information transfer rates in a multiclass brain computer with HMMs, Lee and Choi (2003) used HMMs for supervised learning of EEG sequences, Cincotti et al. (2003) used HMMs as a feature classifier for brain computer interfaces.

Additionally, Rezek et al. (2002); Rezek and Roberts (2000a); Rezek et al. (2000); Rezek and Roberts (2000b) proposed to couple discrete and continuous signals with fixed lag in a Bayesian framework.

None of these models were used for segmentation and interpretation purposes. We propose to couple eye-movement and EEGs into a single model with interpretable hidden states. We make the hypothesis that eye-movement acquire visual and semantical information which is then treated in different locations of the brain with an additional time delay. Hence, the cognitive phase is treated with a delay with respect to eye movements. Concretely, we wish to take into account both signal's information to better characterize and interpret the hidden states linked to reading strategies by taking into account the delay to reduce the uncertainty on states and state transitions. Consequently, each signal is associated with its own latent process, where the first one drives the second one with an additional delay. This engenders the following hypothesis: the cognitive phases are linked to the first discretized signal (eye-movement) and may not change between two fixations but at the start (or end) of a fixation.

Contrarily to the approach by Rezek and Roberts (2000b), the lags introduced in our AHHSMs may be random (or not). Moreover, our models deal with heterogeneous data.

1.2 Global modeling framework

Counting process terminology. Firstly, let us remind the nature of the observed processes. EEGs are the **high-rate sampling processes** at a fixed rate of 1000 Hz while eye-movements is the **low-rate sampling process** at the fixation rate, which is naturally variable. In order to model processes with different sampling rates, let us first define:

- $t \in \{1, ..., \tau\}$, the EEG temporal index in milliseconds,
- N_t, the number of fixations from 1 to t, hence N_τ stands for the total number of fixations, with $N_1 = 1$,
- T_{N_t}, the beginning of the N_t-th fixation, and similarly T_j, the beginning of the j-th fixation,
- $D_j = T_j - T_{j-1}, T_0 = 0$, the time between the j-th and the $j-1$-th fixation (i.e., the duration of the $j-1$-th fixation and associated outgoing saccade).

Assumption 9 (Joint probability distribution sampled at fixation rate)**.** *Let* $\{S_{T_j}\}_{j=1}^{N_\tau}$ *denote any process sampled at a fixation duration level, which implies invariance from* S_{T_j}

to $S_{T_{j+1}-1}$. Therefore we have

$$P(S_{T_j},...,S_{T_{j+1}-1}) = P(S_{T_j})\mathbb{1}\{S_{T_j} = ... = s_{T_{j+1}-1}\}$$

where $P(S_{N_j})$ summarizes the joint probability distribution $P(S_{T_j},...,S_{T_{j+1}-1})$ since states are invariant during fixation/saccade complexes. As a consequence, we subsequently write $P(S_j)$ to refer to the corresponding JPD, assuming that the duration D_j has no influence on the probability $P(S_j)$.

Semi-Markov chain. We also refresh the terminology of the EDHMM's SMC presented in Chapter 1, section 3.3, associated with the low-rate sampling process with the fixation time index j:

- $S^{(1)}_{1:N_\tau}, \forall j \in [\![1, N_\tau]\!], S^{(1)}_j \in [\![1, K]\!]$, the discrete latent state process,
- $R_{1:N_\tau}, \forall j \in [\![1, N_\tau]\!], R_j \in [\![1, \mathscr{D}]\!]$, the discrete latent state duration process,
- $F_{1:N_\tau}, \forall j \in [\![1, N_\tau]\!], F_j \in \{0, 1\}$, the binary latent state duration switch process,

Low-rate sampling output process. Similarly, we refresh the notation of the output process: $O^{(1)}_{1:N_\tau}$, where $\forall j \in [\![1, N_\tau]\!]$, $O^{(1)}_j \in \mathscr{O} = \{v_1, ..., v_G\}$. Note that so far, all the CPDs and parameters remain the same as in the traditional EDHMM. Regarding the data, this corresponds to the fixations in the eye movements.

High-rate sampling output process. We denote the continuous high-rate sampling output process as $O^{(2)}_{1:\tau}$, where $\forall t \in [\![1, \tau]\!]$, $O^{(2)}_t \in \mathbb{R}^{\mathscr{C}}$. In practice, this process corresponds to EEGs or more generally, features of EEGs such as wavelet coefficients. $\mathscr{C}$ is the number of channels (or features). Moreover, to link $O^{(2)}_{1:\tau}$ with $S^{(1)}_{1:N_\tau}$, which have different sampling rates, we define an intermediary set of random variables that correspond to the SMC up to a possible **lag** or delay, $S^{(2)}_{1:\tau}$, where $\forall t \in [\![1, \tau]\!], S^{(2)}_t \in [\![1, K]\!]$. Hence we have the following definition:

$$S^{(2)}_t = S^{(1)}_{N_t - \varepsilon_{N_t}}, \tag{4.1}$$

where $t \in [\![\varepsilon_1, \tau]\!]$ and ε_{N_t} represents the lag at time N_t. In practice, N_t is naturally upper-bounded by τ, the maximal sequence length, but for complexity purposes ε can be both lower- and upper-bounded, say $\varepsilon_{N_t} \in [\![0, \mathscr{L}]\!]$. Note that if $\mathscr{L} = 0$, then there is no lag and it is simply an HSMM with multiple output processes. Also note that $S^{(2)}_{1:\varepsilon_1}$ is considered to be the signal state before acquisition start, and is therefore undefined. Finally, we suppose that the high-rate sampling output process is modeled by a multivariate Gaussian distribution:

$$P(O^{(2)}_t | S^{(2)}_t = k) = \mathscr{N}(\mu_k, \Sigma_k) \tag{4.2}$$

with $\mu_k \in \mathbb{R}^{\mathscr{C}}, \Sigma_k \in \mathbb{R}^{\mathscr{C}\times\mathscr{C}}, \mathscr{C}$ being the dimensionality of the high-rate sampling output process, i.e. the number of channels for EEGs. It is a common practise to use Gaussian distributions to model multivariate EEGs or continuous signals in general, as shown in Obermaier et al. (2001a); Zhong and Ghosh (2002); Chiappa and Bengio (2003). Moreover, note that in equation (4.2), $O^{(2)}_t | S^{(2)}_t = k$ is time invariant. Indeed, temporal information is already encapsulated within the state $S^{(2)}_t = k$ involved in conditional distribution (4.2).

1.3 Specification of the delay distribution

There are plenty of possibilities to model the interaction of the delay between the output processes. Hereafter, we discuss some of the most interesting hypotheses.

Constant lag. If $S^{(2)}_{1:\tau}$ has constant lag regarding $S^{(1)}_{1:N_\tau}$, then we simply rewrite equation (4.1) in the following way:

$$S^{(2)}_t = S^{(1)}_{N_t - \varepsilon} \tag{4.3}$$

$\forall t \in [\![\varepsilon, \tau]\!]$, and $\varepsilon \in [\![1, \mathscr{L}]\!]$ the constant lag. From this relation ensues the following CPD:

$$\begin{aligned}
P(\{S^{(2)}_t\}^{\tau}_{t=1} | \{S^{(1)}_{N_t}\}^{\tau}_{t=1}, \varepsilon) &= \mathbb{1}\{S^{(2)}_{\varepsilon} = S^{(1)}_{N_1}, S^{(2)}_{1+\varepsilon} = S^{(1)}_{N_2}, \ldots, S^{(2)}_{\tau} = S^{(1)}_{N_{\tau-\varepsilon}}\} \\
&= \prod_{t=\varepsilon}^{\tau} \mathbb{1}\{S^{(2)}_t = S^{(1)}_{N_{t-\varepsilon}}\} \\
&= \prod_{l=1}^{\mathscr{L}} \prod_{t=l}^{\tau} \mathbb{1}\{S^{(2)}_t = S^{(1)}_{N_{t-l}}\}^{\mathbb{1}\{\varepsilon = l\}}.
\end{aligned}$$

ε can be either **deterministic**, e.g. given by an expert, estimated by maximum likelihood or **random** and restored using a generalized Viterbi algorithm. In this case, the associated CPD is therefore a discrete Dirac distribution:

$$P(\varepsilon = l) = \mathbb{1}\{\varepsilon = l\}.$$

Non-constant i.i.d. lag. If lag is non-constant, it may vary at different **granularity levels**: fixation or state. In the first case and with an independent and identically distributed hypothesis on the lag, the delay is sampled from $P(\varepsilon_{N_j})$, while in the second case, changes in delays may only occur when the state also transits, i.e. when $F_{N_{j-1}} = 1$. In the rest of the chapter, we consider the fixation level of granularity in order to shorten notations. Considering the relation between the hidden chains from equation (4.1), the associated CPD is:

$$\begin{aligned} P(\{S_t^{(2)}|S_{N_t}^{(1)}, \varepsilon_{N_t}\}_{t=1}^{\tau}) &= \mathbb{1}\{S_{\varepsilon_{N_1}}^{(2)} = S_{N_1}^{(1)}, S_{1+\varepsilon_{N_2}}^{(2)} = S_{N_2}^{(1)}, \dots, S_{\tau}^{(2)} = S_{N_{\tau-\varepsilon_{N_\tau}}}^{(1)}\} \\ &= \prod_{t=\varepsilon_{N_0}}^{\tau} \mathbb{1}\{S_t^{(2)} = S_{N_{t-\varepsilon_{N_t}}}^{(1)}\} \\ &= \prod_{l=1}^{\mathscr{L}} \prod_{t=l}^{\tau} \mathbb{1}\{S_t^{(2)} = S_{N_{t-l}}^{(1)}\}^{\mathbb{1}\{\varepsilon_{N_t}=l\}}. \end{aligned}$$

The lag being discrete, we note the parameters

$$\rho(l) = P(\varepsilon_{N_j} = l)$$

with $\rho(l)$ a tabular distribution of size $\mathscr{L}$. It is then possible to fit any discrete parametric distribution, see section 1.3.3. If we assume that the lag is centered around its mean, $\rho(l)$ can be approximated with a Binomial distribution s.t.

$$P(\varepsilon_{N_j}) = \mathscr{B}(n, p),$$

centered around $\mathbb{E}[\varepsilon_{N_j}] = np$ and ruled by 2 parameters only. However, discrete distribution shapes might sometimes be constraining because of their tied parameters in the expression of the expectation and variance and we could assume a discretization of a continuous distribution, say if $\varepsilon_{N_j} \sim \mathscr{N}(\mu, \sigma^2)$ that can be achieved as follows:

$$\begin{aligned} P(\varepsilon = l) &= P(\varepsilon \in [l-1, l]) \\ &= F_{\mu,\sigma^2}(l) - F_{\mu,\sigma^2}(l-1) \end{aligned} \tag{4.4}$$

then, the difference between the cumulative distribution function can be computed numerically.

Non-constant non-iid lag. The non-constant lag may also have a time dependence, say first order Markovian, which would lead to model

$$\rho_{l'l} = P(\varepsilon_{N_j} = l' | \varepsilon_{N_{j-1}} = l) \tag{4.5}$$

with $\rho_{l'l}$ a transition matrix of size $\mathscr{L} \times \mathscr{L}$, plus the special case at time 1

$$\upsilon_l = P(\varepsilon_{N_1} = l).$$

The main drawback of this hypothesis is that it involves $\mathscr{L} + \mathscr{L}^2$ additional parameters. To overcome this, a possibility is to use an autoregressive process s.t.

$$P(\varepsilon_{N_j} | \varepsilon_{N_{j-1}}) = \mathscr{N}(\varepsilon_{j-1}, \sigma^2),$$

plus the special case $P(\varepsilon_1) = \mathscr{N}(C, \sigma^2)$, with C being a constant, which involves only two parameters. In other words, at each fixation, the new delay is sampled from a normal distribution centered around the previous one signifying that the past dynamics of the delay is captured. We also propose the following discrete approach using a Markov chain with constraints on the transition matrix:

$$P(\varepsilon_j | \varepsilon_{j-1}) = \mathscr{B}(n, p) - \mathbb{E}[\varepsilon_{j-1}], \tag{4.6}$$

with $\mathbb{E}[\varepsilon_{j-1}] = np$. Equation (4.6) describes a noise at time j sampled using a binomial distribution centered around the previous one at time $j-1$ by subtracting the expectation of the binomial distribution. Considering the special case $P(\varepsilon_1) = C + \mathscr{B}(n, p) - np$, where C is a constant, the lag is modeled using only 3 parameters.

Finally, the lag could be modeled on the discrete domain using models for counts data such as Poisson exponentially weighted moving average, see Brandt et al. (2000), or a Poisson autoregressive model, see Brandt and Williams (2001); Fokianos et al. (2009) for more details.

Per state lag. If lags are assumed to have state-dependent distributions, a straightforward extension to equation (4.5) is to add the state to the conditional distribution:

$$\rho_{l'lk} = P(\varepsilon_j = l' | \varepsilon_{j-1} = l, S_j^{(1)} = k), \tag{4.7}$$

plus the special case at initial time 1

$$\upsilon_{lk} = P(\varepsilon_1 = l | S_1^{(1)} = k),$$

leading to increase the number of parameters by a K factor. In a similar fashion as in equation (4.6), the lag can be approximated by a binomial distribution for each state.

Per variable lag. Assuming that $O_{1:\tau}^{(2)}$ corresponds to $\mathscr{C}$ multivariate observations such as a multi-channel EEG or wavelet features of multi-channel EEG, lags may have variable-specific distributions. The $\mathscr{C}$ observation sequences are rewritten $O_{1:\tau}^{(2)} = (O_{1:\tau}^{(2,1)}, \ldots, O_{1:\tau}^{(2,\mathscr{C})})$. Similarly, since each observation sequence has a its own lag, we define the associated state sequences rewritten $S_{1:\tau}^{(2)} = (S_{1:\tau}^{(2,1)}, \ldots, S_{1:\tau}^{(2,\mathscr{C})})$. Hence, the relationship between hidden states becomes

$$S_t^{(2,c)} = S^{(1)}_{N_t - \varepsilon_{N_t}^{(c)}} \tag{4.8}$$

where $\varepsilon_{N_t}^{(c)}$ is the random variable modeling the lag for factor c at time N_t. The distribution of $\varepsilon_{1:N_\tau}^{(c)}$ may also share some common assumptions presented above. Therefore, the number of parameters to model the lag is multiplied by a factor $\mathscr{C}$. It is also required to rewrite the emission distributions, previously given by equation (4.2). Thus, $\forall c \in [\![1, \mathscr{C}]\!]$ we have

$$P(O_t^{(2,c)} | S_t^{(2,c)} = k) = \mathscr{N}(\mu_k, \Sigma_k), \tag{4.9}$$

which conveys the idea that even though the lags, and therefore the states change over time, are different in the hidden states $S_{1:\tau}^{(2)}$, the emission probabilities are shared between all variables.

Figure 4.1 represents the sampling process of an Asynchronous Heterogeneous Hidden semi-Markov Model with a general setting of non-constant non-iid per state lag with lag sampled at a low-rate sampling scale (fixation scale). In the next sections, we develop inference, learning and state restoration procedures within this setup, which is the most generic (we omit per variable lags, which make notations tedious).

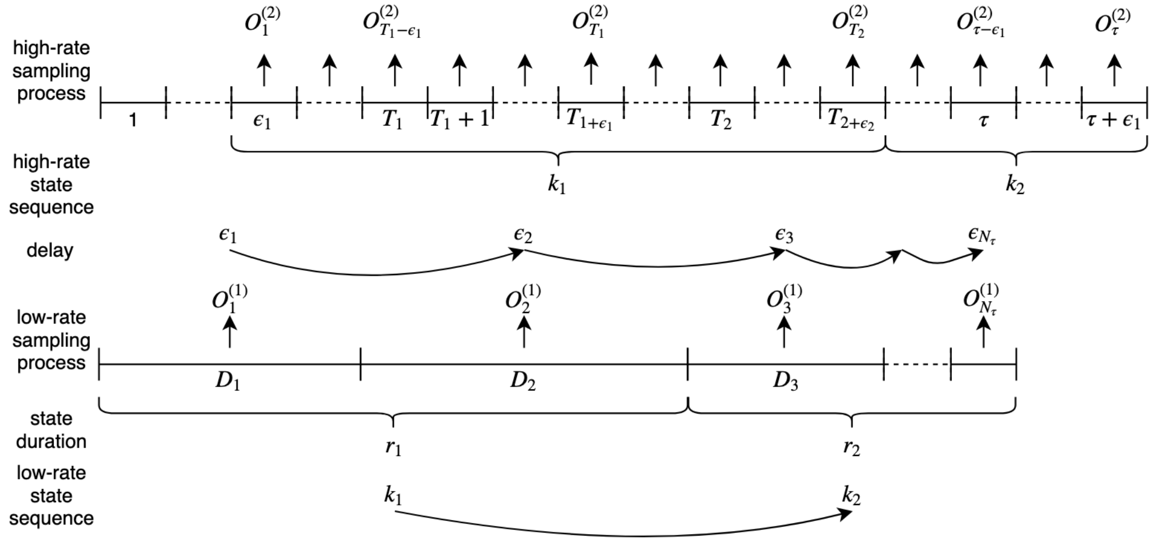


Figure 4.1: AHHSMM sampling process. The first state $S_1^{(1)} = k_1$ is selected using an initial probability π_{k_1}. Then, given k_1, we draw a sojourn duration R_1 with a probability $p_{k_1}(r_1)$ which lasts for two ($r_1 = 2$) low-rate time steps of fixed duration $D_1 + D_2$. A first low-rate observation is sampled from the emission distribution $b_{k_1}(O_1^{(1)})$. This low-rate sampled observation is associated with lag ε_1, intended to map the low-rate to the high-rate sampling processes. Its distribution possibly depends on state k_1. The high-rate sampling process from $O_{\varepsilon_1}^{(2)}$ to $O_{T_1+\varepsilon_1}^{(2)}$, corresponding to the low-rate observation $O_1^{(1)}$, is then sampled at each high-rate time step, from a distribution depending on state k_1, where T_1 is the beginning time of the second fixation. After that, still given k_1, the second low-rate output $O_2^{(1)}$ is emitted at time T_2, as well as the corresponding high-rate outputs $O_{T_1+\varepsilon_1+1:T_2+\varepsilon_2}^{(2)}$ and the associated lag ε_2, whose distribution may depend on the previous lag ε_1, and state k_1. The duration in state k_1 then expires and $S^{(1)}$ transits to a new state $k_2 \neq k_1$ using the transition matrix with a probability $A_{k1,k2}$. A duration R_2 is sampled for state k_2 with a probability $p_{k_2}(r_2)$, and the sampling process goes on again until the end of the sequence.

2 Inference, learning and state restoration

2.1 Parameter learning with heterogeneous data

So far, the proposed model answers the first of the two initial specifications, that is the delay. It can handle two processes sampled at different rates and captures the delay between them in order to synchronize the output processes as well as the latent semi-Markov chain. Consistent estimation of the parameter is expected to be obtained from maximizing the joint likelihood $P(O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau})$. However, this is only holds if the data actually are generated from the model. In analyzing real data sets, observations are expected to deviate from this assumption. If we suppose $\tau \gg N_\tau$, there are much more high-rate sampling outputs than low-rate sampling outputs. Any discrepancy between the assumptions regarding the true distribution of $(O^{(2)}_t)_{t\geq 1}$ and our model could lead to a dramatic perturbations in parameter and states estimation, including those related to the marginal distribution of $O^{(1)}_{j\geq 1}$. This is why we develop some specific estimation procedure that tends to give equal contributions of both processes.

To overcome this issue, let us first define the AHHSMM model parameters $\lambda = (\lambda_1, \lambda_2)$, where $\lambda_1 = \{\pi_k, A_{kk'}, p_k(d), b_k(v_g)\}$ are the model parameters of the traditional (Ferguson, 1980) explicit duration hidden Markov model (section 3.2) and $\lambda_2 = (\rho, \mu, \Sigma)$ are the parameters related to the second output process, more precisely ρ is the set of delay parameters and μ and Σ are the emission distribution parameters of the second output process. We also denote $O^{(1)} = O^{(1)}_{1:N_\tau}$ and $O^{(2)} = O^{(2)}_{1:\tau}$ and we write the joint likelihood of the observed data

$$\begin{aligned} P_{\lambda_1,\lambda_2}(O^{(1)}, O^{(2)}) &= P_{\lambda_1}(O^{(1)}) P_{\lambda_1,\lambda_2}(O^{(2)}|O^{(1)}) \\ &= \sum_{S^{(1)}} P_{\lambda_1}(O^{(1)}, S^{(1)}) \sum_{S^{(2)}} P_{\lambda_2}(O^{(2)}, S^{(2)}) P_{\lambda_2}(S^{(2)}|S^{(1)}), \end{aligned} \tag{4.10}$$

which we show to be decomposing into two parts, each depending on either λ_1 or λ_2. In order to perform the EM algorithm, we also write a decomposition of the conditional expectation of the complete data

$$\begin{aligned} &\mathbb{E}_{\Lambda_1^{(m)},\Lambda_2^{(m)}}[\log P_{\lambda_1,\lambda_2}(O^{(1)}, O^{(2)}, S^{(1)}, S^{(2)})|O^{(1)}, O^{(2)}] \\ &= \mathbb{E}_{\Lambda_1^{(m)},\Lambda_2^{(m)}}[\log P_{\lambda_1}(O^{(1)}, S^{(1)}) + \log P_{\lambda_2}(O^{(2)}|S^{(2)}) + \log P_{\lambda_2}(S^{(2)}|S^{(1)})|O^{(1)}, O^{(2)}], \end{aligned} \tag{4.11}$$

which is to be maximized using EM. Here, $\lambda_1^{(m)}$ and $\lambda_2^{(m)}$ denote the parameters at iteration m of EM. Furthermore, we make the following assumption on the MLE:

Assumption 10. *In an AHHSMM, the maximum likelihood estimator $\hat{\lambda}$ of λ is consistent. Moreover, the sequence of estimates yielded by the EM algorithm tends to $\hat{\lambda}$ when the number of iterations tends to infinity.*

Then, we propose the following decomposition for the maximization of equation (4.11):

Proposition 3. *Let $(\Lambda_1^{(m)}, \Lambda_2^{(m)})_{m\geq 1}$ denote the sequence of iterates of the following modified EM algorithm and $(\tilde{\lambda}_1, \tilde{\lambda}_2)$ denote the true parameters.*

$$(\Lambda_1^{(m+1)}, \Lambda_2^{(m+1)}) = \Bigg(\arg\max_{\lambda_1} \mathbb{E}_{\Lambda_1^{(m)}}[\log P_{\lambda_1}(O^{(1)}, S^{(1)})|O^{(1)}], \\ \arg\max_{\lambda_1,\lambda_2} \mathbb{E}_{\Lambda_1^{(m)},\Lambda_2^{(m)}}[\log P_{\lambda_2}(O^{(2)}|S^{(2)}) + \log P_{\lambda_2}(S^{(2)}|S^{(1)})|O^{(1)}, O^{(2)}] \Bigg), \tag{4.12}$$

Then $\lim_{m\to\infty} (\Lambda_1^{(m)}, \Lambda_2^{(m)}) = (\tilde{\lambda}_1, \tilde{\lambda}_2)$.

Proof. Under assumption 10, the left term in the expectation of equation (4.11) can be taken out and optimized independently since asymptotically:

$$\lim_{m\to\infty} \arg\max_{\lambda_1} \mathbb{E}_{\Lambda_1^{(m)},\Lambda_2^{(m)}}[\log P_{\lambda_1,\lambda_2}(O^{(1)}, S^{(1)})|O^{(1)}, O^{(2)}] = \lim_{m\to\infty} \max_{\lambda_1} \mathbb{E}_{\Lambda_1^{(m)}}[\log P_{\lambda_1}(O^{(1)}, S^{(1)})|O^{(1)}],$$

and both the quantities are equal to the real parameter $\tilde{\lambda}_1$. As a consequence, the three terms in the expectation of equation (4.11) can be optimized independently. The first term, $\mathbb{E}_{\Lambda_1^{(m)}}[\log P_{\lambda_1}(O^{(1)}, S^{(1)})|O^{(1)}]$, corresponds to the low-rate sampling process and is computed through the general EM algorithm for HSMM presented in Chapter 1 section 3.4. It shall be noticed from equation (4.12) that the high-rate sampling process are not influencing the parameter estimation related to the low-rate sampling process whereas the low-rate sampling process is influencing estimation of the parameters related to the high-rate sampling process. □

2.2 Inference and learning

By combining all the CPDs previously defined with the most general delay given by equation (4.7), i.e. the non-constant non-iid per state lag, we write the joint probability distribution as:

$$
\begin{aligned}
&P(\{O_j^{(1)}, S_j^{(1)}, R_j, F_j, \varepsilon_j\}_{j=1}^{N_\tau}, \{S_t^{(2)}, O_t^{(2)}\}_{t=1}^{\tau}) \\
&= P(\{O_j^{(1)}\}, \{O_t^{(2)}\} | \{S_j^{(1)}, R_j, F_j\}, \{S_t^{(2)}\}) P(\{S_j^{(1)}, R_j, F_j, \varepsilon_j\}, \{S_t^{(2)}\}) \\
&= P(\{O_j^{(1)}\} | \{S_j^{(1)}\}) P(\{O_t^{(2)}\} | \{S_t^{(2)}\}) P(\{S_t^{(2)}\} | \{S_j^{(1)}\}, \{\varepsilon_j\}) P(\{S_j^{(1)}, R_j, F_j\}) P(\{\varepsilon_j | S_j^{(1)}\}) \\
&= P(S_1^{(1)}) P(R_1 | S_1^{(1)}) P(\varepsilon_1 | S_1^{(1)}) \prod_{j=1}^{N_\tau} P(O_j^{(1)} | S_j^{(1)}) \\
&\quad \prod_{j=2}^{N_\tau} \left(P(R_j | R_{j-1}, S_j^{(1)}, F_j) P(S_j^{(1)} | S_{j-1}^{(1)}, F_j) P(F_j | R_{j-1}) P(\varepsilon_j | \varepsilon_{j-1}, F_{j-1}, S_j^{(1)}) \right) \\
&\quad P(\{S_t^{(2)} | \varepsilon_{N_t}, S_{N_t}^{(1)}\}) \prod_{t=1}^{\tau} P(O_t^{(2)} | S_t^{(2)}) \\
&= \prod_{k=1}^{K} \left(P(S_1^{(1)} = k)^{\mathbb{1}\{s_1^{(1)} = k\}} \prod_{d=1}^{\mathscr{D}} P(R_1 = d | S_1^{(1)} = k)^{\mathbb{1}\{r_1 = d, s_1^{(1)} = k\}} \prod_{l=1}^{\mathscr{L}} P(\varepsilon_1 = l | S_1 = k)^{\mathbb{1}\{\varepsilon_1 = l, s_1 = k\}} \right) \\
&\quad \prod_{j=1}^{N_\tau} \prod_{k=1}^{K} \Bigg[\prod_{v_g \in \mathscr{O}} P(O_j^{(1)} = v_g | S_j^{(1)} = k)^{\mathbb{1}\{o_j^{(1)} = v_g, s_j^{(1)} = k\}} \prod_{l=1}^{\mathscr{L}} \prod_{l'=1}^{\mathscr{L}} P(\varepsilon_j = l | \varepsilon_{j-1} = l', S_j^{(1)} = k)^{\mathbb{1}\{\varepsilon_j = l, \varepsilon_{j-1} = l', s_j^{(1)} = k\}} \\
&\quad \prod_{k'=1}^{K} \prod_{d=1}^{\mathscr{D}} \prod_{d'=1}^{\mathscr{D}} \prod_{f=0}^{1} \Bigg(P(R_j = d | R_{j-1} = d', S_j^{(1)} = k, F_{j-1} = f)^{\mathbb{1}\{r_j = d, r_{j-1} = d', s_j^{(1)} = k, f_{j-1} = f\}} \\
&\qquad\qquad P(S_j^{(1)} = k | S_{j-1}^{(1)} = k', F_{j-1} = f)^{\mathbb{1}\{s_j^{(1)} = k, s_{j-1}^{(1)} = k', f_{j-1} = f\}} \\
&\qquad\qquad P(F_j = f | R_j = d')^{\mathbb{1}\{f_j = f, r_j = d'\}} \Bigg) \Bigg] \\
&\quad \prod_{l=1}^{\mathscr{L}} \prod_{t=l}^{\tau} \prod_{k=1}^{K} \prod_{k=1}^{K} \mathbb{1}\{k = k'\}^{\mathbb{1}\{\varepsilon_{N_t} = l, s_t^{(2)} = k, s_{N_{t-l}}^{(1)} = k\}} \prod_{t=1}^{\tau} \prod_{k=1}^{K} P(O_t^{(2)} | S_t^{(2)} = k)^{\mathbb{1}\{s_t^{(2)} = k\}} \\
&= \prod_{k=1}^{K} \left(\pi_k^{\mathbb{1}\{s_1^{(1)} = k\}} \prod_{d=1}^{\mathscr{D}} p_j(d)^{\mathbb{1}\{r_1 = d, s_1^{(1)} = k\}} \prod_{l=1}^{\mathscr{L}} \upsilon_{kl}^{\mathbb{1}\{\varepsilon_1 = l, s_1^{(1)} = k\}} \right) \\
&\quad \prod_{j=1}^{N_\tau} \prod_{k=1}^{K} \Bigg[\prod_{v_g \in \mathscr{O}} b_j(v_g)^{\mathbb{1}\{o_j^{(1)} = v_g, s_j^{(1)} = k\}} \prod_{l=1}^{\mathscr{L}} \prod_{l'=1}^{\mathscr{L}} \rho_{kll'}^{\mathbb{1}\{\varepsilon_j = l, \varepsilon_{j-1} = l', s_j^{(1)} = k\}} \\
&\quad \prod_{k'=1}^{K} \prod_{d=1}^{\mathscr{D}} \prod_{d'=1}^{\mathscr{D}} \Bigg(\mathbb{1}\{d = d' - 1\}^{\mathbb{1}\{r_j = d, r_{j-1} = d', s_j^{(1)} = k, f_{j-1} = 0\}} p_j(d)^{\mathbb{1}\{r_j = d, r_{j-1} = d', s_j^{(1)} = k, f_{j-1} = 1\}} \\
&\qquad\qquad \mathbb{1}\{k = k'\}^{\mathbb{1}\{s_j^{(1)} = k, s_{j-1}^{(1)} = k', f_{j-1} = 0\}} A_{kk'}^{\mathbb{1}\{s_j^{(1)} = k, s_{j-1}^{(1)} = k', f_{j-1} = 1\}} \\
&\qquad\qquad \mathbb{1}\{d' > 1\}^{\mathbb{1}\{f_j = 0, r_j = d'\}} \mathbb{1}\{d' = 1\}^{\mathbb{1}\{f_j = 1, r_j = d'\}} \Bigg) \\
&\quad \prod_{l=l}^{\mathscr{L}} \prod_{t=1}^{\tau} \prod_{k=1}^{K} \prod_{k'=1}^{K} \mathbb{1}\{k = k'\}^{\mathbb{1}\{\varepsilon_{N_t} = l, s_t^{(2)} = k, s_{N_{t-l}}^{(1)} = k'\}} \prod_{t=1}^{\tau} \prod_{k=1}^{K} f_{\mathscr{N}(\mu_k, \Sigma_k)}(o_t^{(2)})^{\mathbb{1}\{s_t^{(2)} = k\}}.
\end{aligned}
$$

In order to apply EM, we use proposition 3 and compute the $Q(\theta,\theta^{old})$ function the following way:

$$\begin{aligned} Q(\theta^{old},\theta) =& \mathbb{E}[\log P(\{O_j^{(1)},S_j^{(1)},R_j,F_j,\varepsilon_j\}_{j=1}^{\tau},\{S_t^{(2)},O_t^{(2)}\}_{t=1}^{\tau};\theta)|\{O_j^{(1)}\}_{j=1}^{N_\tau},\{O_t^{(2)}\}_{t=1}^{\tau};\theta^{old}] \\ =& \mathbb{E}[\log P_{\lambda_1}(\{S_j^{(1)},O_j^{(1)},R_j,F_j\})|\{O_j^{(1)}\},\theta^{old}] \\ &+\mathbb{E}[\log P_{\lambda_2}(\{O_t^{(2)}|S_t^{(2)}\})+\log P_{\lambda_2}(\{S_t^{(2)}\},\{\varepsilon_j\}|\{S_j^{(1)}\})|\{O_j^{(1)}\},\{O_t^{(2)}\},\theta^{old}], \end{aligned} \tag{4.13}$$

with $\lambda_1 = (\pi,A,p_\theta,b_\theta)$ and $\lambda_2 = (\upsilon,\rho,\mu,\Sigma)$. The left term, i.e. the first expectation, corresponds exactly to the Q-function of a EDHMM given by equation (1.29) computed via three expected sufficient statistics given in Chapter 1, equations (1.40), (1.38) and (1.39). The novelty arise from the right term of equation (4.13) which we decompose:

$$\begin{aligned} &\mathbb{E}[\log P_{\lambda_2}(\{O_t^{(2)}|S_t^{(2)}\})+\log P_{\lambda_2}(\{S_t^{(2)}\},\{\varepsilon_j\}|S_j^{(1)})|\{O_j^{(1)}\},\{O^{2)}\},\theta^{old}] \\ &=\mathbb{E}[\log P_{\lambda_2}(\{O_t^{(2)}|S_t^{(2)}\})|\{O_j^{(1)}\},\{O^{2)}\},\theta^{old}]+\mathbb{E}[\log P_{\lambda_2}(\{S_t^{(2)}\},\{\varepsilon_j\}|S_j^{(1)})|\{O_j^{(1)}\},\{O^{2)}\},\theta^{old}]^{.} \end{aligned} \tag{4.14}$$

We first compute the left term:

$$\begin{aligned} &\mathbb{E}[\log P_{\lambda_2}(\{O_t^{(2)}|S_t^{(2)}\})+\log P_{\lambda_2}(\{S_t^{(2)}\},\{\varepsilon_j\}|S_j^{(1)})|\{O_j^{(1)}\},\{O^{2)}\},\theta^{old}] \\ &=\mathbb{E}[\sum_{t=1}^{\tau}\sum_{k=1}^{K}\mathbb{1}\{s_t^{(2)}=k\}\log f_{\mathcal{N}(\mu_k,\Sigma_k)}(O_t^{(2)})|\{O_j^{(1)}\},\{O_t^{(2)}\},\theta^{old}] \\ &=\sum_{t=1}^{\tau}\sum_{k=1}^{K}P_{\theta^{old}}(S_t^{(2)}=k|\{O_j^{(1)}\},\{O_t^{(2)}\})\log f_{\mathcal{N}(\mu_k,\Sigma_k)}(o_t^{(2)}) \end{aligned} \tag{4.15}$$

noting that the expectation of an indicator is simply the probability that it takes the value 1. We then compute the right term of equation (4.14):

$$\begin{aligned}
&\mathbb{E}[\log P_{\lambda_2}(\{S_t^{(2)}\},\{\varepsilon_j\}|S_j^{(1)})|\{O_j^{(1)}\},\{O^{2)}\},\theta^{old}] \\
&= \mathbb{E}[\sum_{k=1}^{K}\sum_{l=1}^{\mathscr{L}}\Bigg(\mathbb{1}\{\varepsilon_1 = l, s_1^{(1)} = k\}\log \upsilon_{k,l} + \sum_{j=2}^{N_\tau}\sum_{l'=1}^{\mathscr{L}}\mathbb{1}\{\varepsilon_j = l, \varepsilon_{j-1} = l', s_j^{(1)} = k\}\log \rho_{kl'l} \\
&+ \sum_{t=l}^{\tau}\sum_{k'=1}^{K}\mathbb{1}\{\varepsilon_{N_t} = l, s_t^{(2)} = k, s_{N_{t-l}}^{(1)} = k'\}\log \mathbb{1}\{k = k'\}\Bigg)|\{O_j^{(1)}\},\{O^{2)}\},\theta^{old}] \\
&= \sum_{k=1}^{K}\sum_{l=1}^{\mathscr{L}}\Bigg(P_{\theta^{old}}(\varepsilon_1 = l, S_1^{(1)} = k|\{O_j^{(1)}\},\{O^{2)}\})\log_{\upsilon_{k,l}} \qquad . \\
&+ \sum_{j=2}^{N_\tau}\sum_{l'=1}^{\mathscr{L}}P_{\theta^{old}}(\varepsilon_j = l, \varepsilon_{j-1} = l', S_j^{(1)} = k|\{O_j^{(1)}\},\{O^{2)}\})\log \rho_{kl'l} \\
&+ \sum_{t=l}^{\tau}\sum_{k'=1}^{K}P_{\theta^{old}}(\varepsilon_{N_t} = l, s_t^{(2)} = k, S_{N_{t-l}}^{(1)} = k'|\{O_j^{(1)}\},\{O^{2)}\})\log \mathbb{1}\{k = k'\}\Bigg)
\end{aligned} \tag{4.16}$$

Both equations (4.15) and (4.16) highlight new expected sufficient statistics, i.e. the terms multiplying parameters, to be computed in the E-step:

$$P_{\theta^{old}}(\varepsilon_j = l, \varepsilon_{j-1} = l', S_j^{(1)} = k|\{O_j^{(1)}\},\{O_t^{(2)}\}), \tag{4.17}$$

$$P_{\theta^{old}}(\varepsilon_1 = l, S_1^{(1)} = k|\{O_j^{(1)}\},\{O_t^{(2)}\}), \tag{4.18}$$

$$P_{\theta^{old}}(S_t^{(2)} = k|\{O_j^{(1)}\},\{O_t^{(2)}\}). \tag{4.19}$$

E-step. As in inference in HSMM, we start by defining forward and backward variables starting with the forward variables,

$$\begin{aligned}
\alpha_j(k,l) &= P(O^{(1)}_{1:j}, O^{(2)}_{1:T_j+l-1}, S^{(1)}_j = k, \varepsilon_j = l, F_j = 1) \\
&= \sum_{d=1}^{D} \sum_{l'} P(O^{(1)}_{1:j}, O^{(2)}_{1:T_j+l-1}, F_j = 1, \varepsilon_j = l, F_{j-d} = 1, R_{j-d+1} = d, S^{(1)}_{j-d+1:j} = k, \varepsilon_{j-d} = l') \\
&= \sum_{d=1}^{D} \sum_{l'} P(F_j = 1 | R_{j-d+1} = d) \\
&\qquad P(R_{j-d+1} = d | S^{(1)}_{j-d+1} = k, F_{j-d} = 1) \\
&\qquad P(\varepsilon_j = l | \varepsilon_{j-d+1} = l', S^{(1)}_{j-d+1:j} = k) \\
&\qquad P(O^{(1)}_{j-d+1:j} | S^{(1)}_{j-d+1:j} = k) \\
&\qquad P(O^{(2)}_{T_{j-d+1}+l':T_j+l-1} | S^{(1)}_{j-d+1:j} = k, \varepsilon_{j-d+1} = l', \varepsilon_j = l) \\
&\qquad P(O^{(1)}_{1:j-d}, O^{(2)}_{1:T_{j-d+1}+l'-1}, \varepsilon_{j-d+1} = l', S^{(1)}_{j-d+1} = k, F_{j-d} = 1) \\
&= \sum_{d=1}^{D} \sum_{l'} p_k(d) (\rho_k)^{d-1}_{l'l} \alpha^*_{j-d}(k,l') \prod_{j'=j-d+1}^{j} b_k(O^{(1)}_{j'}) \prod_{t'=T_{j-d+1}+l'}^{T_j+l-1} f_{\mathcal{N}(\mu_k,\Sigma_k)(O^{(2)}_{t'})}
\end{aligned} \tag{4.20}$$

with,

$$\begin{aligned}
\alpha^*_j(k,l) &= P(O^{(1)}_{1:j}, O^{(2)}_{1:T_{j+1}+l-1}, S^{(1)}_{j+1} = k, \varepsilon_{j+1} = l, F_j = 1) \\
&= \sum_{k'=1}^{K} \sum_{l'=0}^{\mathscr{L}} P(\varepsilon_{j+1} = l | \varepsilon_j = l', S^{(1)}_{j+1} = k) P(S^{(1)}_{j+1} = k | S^{(1)}_j = k', F_j = 1) \\
&\qquad P(O^{(2)}_{T_j+l':T_{j+1}+l-1} | S^{(1)}_j = k', \varepsilon_{j+1} = l, \varepsilon_j = l') P(O^{(1)}_{1:j}, O^{(2)}_{1:T_j+l-1}, \varepsilon_j = l, S^{(1)}_j = k', F_j = 1) \\
&= \sum_{k'=1}^{K} \sum_{l'=0}^{\mathscr{L}} \rho_{kll'} A_{k'k} \alpha_j(k',l') \prod_{t'=T_j+l'}^{T_{j+1}+l-1} f_{\mathcal{N}(\mu_{k'},\Sigma_{k'})}(O^{(2)}_{t'})
\end{aligned} \tag{4.21}$$

and the first term,

$$\begin{aligned}
\alpha^*_0(k,l) &= P(S^{(1)}_1 = k, F_0 = 1, \varepsilon_1 = l) \\
&= P(\varepsilon_1 = l | S^{(1)}_1 = k) P(S^{(1)}_1 = k) \\
&= \upsilon_{kl} \pi_k,
\end{aligned}$$

where we remind that T_j is the transition time of fixation j. First, considering the computation of $\alpha_j(k,l)$, the integration over the sojourn duration set $\mathscr{D}$ aims at computing all the possible durations through $F_{j-d} = 1$ which implies $R_{j-d+1} = d$ plus constant state k from time $j-d+1$ to j, i.e $S^{(1)}_{j-d+1:j} = k$. Then, the goal of the integration over ε_{j-d} is to compute probabilities related to different lag values for different durations that were all computed conditionally to state k. The second step is simply an application of the conditional independences. It should be noted that $S^{(1)}_{j-d+1:j} = k$ is a shortcut for $S^{(1)}_j = k, F_j = 1, F_{j-d} = 1, R_{j-d+1} = d$ and is equivalent. Also note that, $\forall d \in [\![1, \mathscr{D}]\!], P(F_j = 1|R_{j-d} = d) = 1$ and is therefore omitted in the last development of the equation. Finally, $P(\varepsilon_j = l|\varepsilon_{j-d+1} = l', S^{(1)}_{j-d+1:j} = k, R_{j-d+1} = d, F_{j-d} = 1) = (\rho_k)^{d-1}_{l'l}$ is directly computed at fixed state k in order to use the Chapman-Kolmogorov equation, see equation (1.11). The probability of the next state-conditional delay is then computed in the equation of $\alpha^*_j(k,l)$ as well as the preceding state. $\alpha_j(k,l)$ describes the forward behavior for state k which is going to be exited at time $j+1$ while $\alpha^*_j(k,l)$ is the forward behavior for a state k which was just entered at time $j+1$. This trick is HSMM-specific and true interest of the decomposition is be shown in the computation of the next equations. Nonetheless, the modeling of the delay induces a change in the computation of the emission distribution since it is split between $\alpha^*_j(k,l)$ and $\alpha_j(k,l)$ which was not the case before. In order to compute the emission distribution of $O^{(2)}_{T_{j-d+1}+\varepsilon_{j-d+1}:T_{j+1}+\varepsilon_{j+1}-1}$, it is indeed necessary to split the sequence such as $O^{(2)}_{T_{j-d+1}+\varepsilon_{j-d+1}:T_j+\varepsilon_j-1}$ and $O^{(2)}_{T_j+\varepsilon_j:T_{j+1}+\varepsilon_{j+1}-1}$ since the delay of the second term ε_{j+1} is conditional to $S^{(1)}_{j+1}$ while the rest is all conditional to a fixed state $S^{(1)}_{j-d+1:j}$. This trick is a performance improvement. From equations (4.20) and (4.21), it can be seen that forward pass has complexity $O(\tau\mathscr{L}^2\mathscr{K}^2\mathscr{D})$, an increase by a factor $\mathscr{L}^2$ compared to the forward pass EDHMM but which can be controlled by using inferior and/or superior bounds on the delay.

A similar schema is applied for the backward variables,

$$\begin{aligned}
\beta_j(k,l) &= P(O^{(1)}_{j+1:N_\tau}, O^{(2)}_{T_j+l:\tau} | S^{(1)}_j = k, \varepsilon_j = l, F_j = 1) \\
&= \sum_{k'=1}^{K} \sum_{l'=0}^{\mathscr{L}} P(O^{(1)}_{j+1:N_\tau}, O^{(2)}_{T_{j+1}+l':\tau} | S^{(1)}_{j+1} = k', \varepsilon_{j+1} = l', F_j = 1) \\
&\qquad\qquad P(\varepsilon_{j+1} = l' | \varepsilon_j = l, S^{(1)}_{j+1} = k') P(S^{(1)}_{j+1} = k' | S^{(1)}_j = k, F_j = 1) \\
&\qquad\qquad P(O^{(2)}_{T_j+l:T_{j+1}+l'-1} | S^{(1)}_j = k, \varepsilon_j = l, \varepsilon_{j+1} = l', F_j = 1) \\
&= \sum_{k'=1}^{K} \sum_{l'=0}^{\mathscr{L}} \rho_{k'll'} A_{kk'} \beta^*_j(k',l') \prod_{T_j+l}^{T_{j+1}+l'-1} f_{\mathcal{N}(\mu_k,\Sigma_k)}
\end{aligned} \tag{4.22}$$

with,

$$\begin{aligned}
\beta^*_j(k,l) &= P(O^{(1)}_{j+1:N_\tau}, O^{(2)}_{T_{j+1}+l:\tau} | S^{(1)}_{j+1} = k, \varepsilon_{j+1} = l, F_j = 1) \\
&= \sum_{d=1}^{D} \sum_{l'=0}^{\mathscr{L}} P(O^{(1)}_{j+1:N_\tau}, O^{(2)}_{T_{j+1}+l:\tau}, F_{j+d} = 1, R_{j+1} = d, \varepsilon_{j+d} = l' | F_j = 1, \varepsilon_{j+1} = l, S^{(1)}_{j+1:j+d} = k) \\
&= \sum_{d=1}^{D} \sum_{l'=0}^{\mathscr{L}} P(F_{j+d} = 1 | R_{t+1} = d) \\
&\qquad P(R_{j+1} = d | S^{(1)}_{j+1} = k, F_j = 1) \\
&\qquad P(\varepsilon_{j+d} = l' | \varepsilon_{j+1} = l, S^{(1)}_{j+1:j+d} = k) \\
&\qquad P(O^{(1)}_{j+1:j+d} | S^{(1)}_{j+1:j+d} = k) \\
&\qquad P(O^{(2)}_{T_{j+1}+l:T_{j+d}+l'-1} | S^{(1)}_{j+1:j+d} = k, \varepsilon_{j+1} = l, \varepsilon_{j+d} = l') \\
&\qquad P(O^{(1)}_{j+d+1:N_\tau}, O^{(2)}_{T_{j+d}+l':\tau}, S^{(1)}_{j+d} = k, \varepsilon_{j+d} = l', F_{t+d} = 1) \\
&= \sum_{d=1}^{D} \sum_{l'=0}^{\mathscr{L}} p_k(d) (\rho_k)^{d-1}_{ll'} \beta_{j+d}(k,l') \prod_{j'=j+1}^{j+d} b_k(O^{(1)}_{j'}) \prod_{t'=T_{j+1}+l}^{T_{j+d}+l'-1} f_{\mathcal{N}(\mu_k,\Sigma_k)(O^{(2)}_{t'})},
\end{aligned} \tag{4.23}$$

with the termination terms $\forall k \in [\![1,K]\!], l \in [\![0,\mathscr{L}]\!]$:

$$\beta_{N_\tau}(k,l) = 1.$$

The way of computing backward variables is very similar to the forward variables, the integration over d computes all possible durations through $F_{j+d} = 1$ which implies

$R_{j+1} = d$ as well as constant state k from $j+1$ to $j+d$. $\forall d, P(F_{j+d} = 1 | R_{t+1} = d) = 1$ and is then omitted.

Forward and backward variables are computed recursively and are used to compute the expected sufficient statistics. Starting with ESS (4.17), we first compute intermediate quantities:

$$\begin{aligned}
\zeta_j(l,l',k) &= P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_{j-1} = k, F_{j-1} = 1 | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) \\
&\propto P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_{j-1} = k, F_{j-1} = 1, O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) \\
&= \sum_{k'=1}^{K} P(O^{(1)}_{j:N_\tau}, O^{(2)}_{T_j+l:\tau} | \varepsilon_j = l, S^{(1)}_j = k', F_{j-1} = 1) \\
&\qquad P(\varepsilon_j = l | \varepsilon_{j-1} = l', S^{(1)}_j = k') P(S^{(1)}_j = k' | S^{(1)}_{j-1} = k, F_{j-1} = 1) \\
&\qquad P(O^{(2)}_{T_{j-1}+l':T_j+l-1} | S^{(1)}_{j-1} = k, \varepsilon_{j-1} = l', \varepsilon_j = l) \\
&\qquad P(O^{(1)}_{1:j-1}, O^{(2)}_{1:T_{j-1}+l'-1}, S^{(1)}_{j-1} = k, \varepsilon_{j-1} = l', F_{j-1} = 1) \\
&= \sum_{k'=1}^{K} \beta^*_{j-1}(k',l) \rho_{k'l'l} A_{kk'} \alpha_{j-1}(k,l') \prod_{t'=T_j+l'}^{T_{j+1}+l-1} f_{\mathcal{N}(\mu_k,\Sigma_k)(O^{(2)}_{t'})},
\end{aligned} \tag{4.24}$$

and

$$\begin{aligned}
\zeta^*_j(l,l',k) &= P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_j = k, F_{j-1} = 1 | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) \\
&\propto P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_j = k, F_{j-1} = 1, O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) \\
&= \sum_{k'=1}^{K} P(O^{(1)}_{j:N_\tau}, O^{(2)}_{T_j+l:\tau} | \varepsilon_j = l, S^{(1)}_j = k, F_{j-1} = 1) \\
&\qquad P(\varepsilon_j = l | \varepsilon_{j-1} = l', S^{(1)}_j = k) P(S^{(1)}_j = k | S^{(1)}_{j-1} = k', F_{j-1} = 1) \\
&\qquad P(O^{(2)}_{T_{j-1}+l':T_j+l-1} | S^{(1)}_{j-1} = k', \varepsilon_{j-1} = l', \varepsilon_j = l) \\
&\qquad P(O^{(1)}_{1:j-1}, O^{(2)}_{1:T_{j-1}+l'-1}, S^{(1)}_{j-1} = k', \varepsilon_{j-1} = l', F_{j-1} = 1) \\
&= \sum_{k'=1}^{K} \beta_{j-1}(k,l) \rho_{kl'l} A_{k'k} \alpha_{j-1}(k',l') \prod_{t'=T_j+l'}^{T_{j+1}+l-1} f_{\mathcal{N}(\mu_{k'},\Sigma_{k'})(O^{(2)}_{t'})},
\end{aligned} \tag{4.25}$$

which we use to compute ESS (4.17):

$$\begin{aligned}
&P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_{j-1} = k | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) \\
&= \sum_{S^{(1)}_j} P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_{j-1} = k, S^{(1)}_j = k' | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) \\
&= P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_{j-1} = k, S^{(1)}_j \neq k | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) + P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_{j-1} = k, S^{(1)}_j = k | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) \\
&= P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_{j-1} = k, S^{(1)}_j \neq k | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) + P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_j = k | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) \\
&\quad - P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_j = k, S^{(1)}_{j-1} \neq k [O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) \\
&= P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_{j-1} = k, F_{j-1} = 1 | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) + P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_j = k | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) \\
&\quad - P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_j = k, F_{j-1} = 1 | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) \\
&= P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_j = k | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) + \zeta^*_j(l, l', k) - \zeta_j(l, l', k) \\
&= \sum_{j'=1}^{N_\tau} \zeta_{j'}(l, l', k) - \zeta^*_j(l, l', k).
\end{aligned} \tag{4.26}$$

The computation of ESS (4.17) in equation (4.26) is a bit tricky and similar to (1.40) in HSMM. The first key is to notice that summing over all values of k for $S^{(1)}_j$ is equal to the sum over k plus the values different than k since different than k includes all values but k. For example it is clear that $P(X) = P(X, Y = k) + P(X, Y \neq k)$. The same trick is applied on line 3 to re-decompose the right term of line 2, but this time we decomposed the term as $P(X, Y = k) = P(X) - P(X, Y \neq k)$. The fourth line simply rewrites the probability s.t. $S^{(1)}_j \neq k$ and $S^{(1)}_{j-1} = k$ is equal to $S^{(1)}_{j-1} = k$ and $F_{j-1} = k$, in other words, both notations give the information of the current state plus a transition in the next step to an unknown state. The fifth lines simply rewrites the equation in terms of previously computed quantities $\zeta_j(k, l, l')$ and $\zeta^*_j(k, l, l')$. Finally, the last line rewrites the ESS by noticing the induction procedure in the equality. Since we have $P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_{j-1} = k | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) = P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_j = k | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) + \zeta^*_j(l, l', k) - \zeta_j(l, l', k)$ then $P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_j = k | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) = P(\varepsilon_j = l, \varepsilon_{j-1} = l', S^{(1)}_{j-1} = k | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau}) - \zeta^*_j(l, l', k) + \zeta_j(l, l', k)$ which gives us the induction step, with the base case $P(\varepsilon_1 = l, S^{(1)}_1 = k | O^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau})$ which is also the next expectation sufficient statistics, equation

(4.18), computed hereafter:

$$
\begin{aligned}
P(\varepsilon_1 = l, S_1^{(1)} = k | O_{1:N_\tau}^{(1)}, o_{1:\tau}^{(2)}) \propto P(\varepsilon_1 = l, S_1^{(1)} = k, O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}) \\
= P(O_{1:N_\tau}^{(1)}, O_{l:\tau}^{(2)} | S_1^{(1)} = k, \varepsilon_1 = l, F_0 = 1) \\
P(\varepsilon_1 = l | S_1^{(1)} = k) P(S_1^{(1)} = k) \\
= \beta_0^*(k,l) \pi_k \upsilon_{kl}
\end{aligned}
\tag{4.27}
$$

Finally, to deal with ESS (4.19), we first make the following assumption:

Assumption 11. *In an asynchronous hidden semi-Markov model, $\forall j \in [\![1, N_\tau]\!], \varepsilon_j < T_{j+1} - T_j$. In other words, at each time step, the delay must be upper-bounded by the current low-rate sampling process step duration.*

Then, we redefine equation (4.19) it in terms of the low-rate sampling process:

$$
\begin{aligned}
P(S_t^{(2)} = k | O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}) &= \sum_{k'=1}^{K} \sum_{l=0}^{\mathscr{L}} P(S_t^{(2)} = k, S_{N_{t-l}}^{(1)} = k', \varepsilon_{N_t} = l | O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}) \\
&= \sum_{k'=1}^{K} \sum_{l=0}^{\mathscr{L}} P(S_t^{(2)} = k | S_{N_{t-l}}^{(1)} = k', \varepsilon_{N_t} = l, O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}) \\
&\qquad P(S_{N_{t-l}}^{(1)} = k', \varepsilon_{N_t} = l | O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}) \\
&= \sum_{l=0}^{\mathscr{L}} P(S_{N_{t-l}}^{(1)} = k, \varepsilon_{N_t} = l | O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}) \\
&= \sum_{l=0}^{\mathscr{L}} \Bigg(\mathbb{1}\{N_{t-l} = N_t\} P(S_j^{(1)} = k, \varepsilon_j = l | O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}) \\
&\qquad + \mathbb{1}\{N_{t-l} < N_t\} P(S_{j-1}^{(1)} = k, \varepsilon_j = l | O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}). \Bigg)
\end{aligned}
$$

The first step integrates over state $S_{N_{t-l}}^{(1)}$ as well as the associated delay ε_{N_t} in order to bring the conditional of $S_t^{(2)}$ out which is equal to one if $k = k'$. The last step decomposition relies on assumption 11, $S_t^{(2)}$ may only be equal to the current state $S_{N_t}^{(1)}$ or the preceding one $S_{N_t-1}^{(1)}$ since the delay is upper-bounded by the low-rate. The first term of equation (2.2) is simply computed reusing the result of equation (4.26)

$$
P(S_j^{(1)} = k, \varepsilon_j = l | O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}) = \sum_{l=0}^{\mathscr{L}} P(\varepsilon_j = l, \varepsilon_{j-1} = l', S_{j-1}^{(1)} = k | O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}),
$$

then we compute the second term of equation (2.2) by applying a similar strategy as equation (4.26). Starting with

$$\begin{aligned}\gamma_j(k,l) &= P(S^{(1)}_{j-1}=k,\varepsilon_j=l,F_{j-1}=1|O^{(1)}_{1:N_\tau},O^{(2)}_{1:\tau})\\ &= P(S^{(1)}_{j}=k,\varepsilon_{j+1}=l,F_{j}=1|O^{(1)}_{1:N_\tau},O^{(2)}_{1:\tau})\\ &= \sum_{l'=0}^{\mathscr{L}} \zeta_j(l,l',k)\end{aligned}$$

and,

$$\begin{aligned}\gamma_j(k,l) &= P(S^{(1)}_{j}=k,\varepsilon_j=l,F_{j-1}=1|O^{(1)}_{1:N_\tau},O^{(2)}_{1:\tau})\\ &= P(S^{(1)}_{j+1}=k,\varepsilon_{j+1}=l,F_{j}=1|O^{(1)}_{1:N_\tau},O^{(2)}_{1:\tau})\\ &= \sum_{l'=0}^{\mathscr{L}} \zeta^*_j(l,l',k)\end{aligned}$$

which are simply summing out ε_j and applying the homogeneity definition, see Chapter 1 section 1.5. We may now rewrite the desired quantity:

$$P(S^{(1)}_{j-1}=k,\varepsilon_j=l|O^{(1)}_{1:N_\tau},O^{(2)}_{1:\tau}) = \sum_{j'=1}^{N_\tau} \gamma_{j'}(k)-\gamma^*_{j'}(k).$$

M-step. Denoting θ, the whole set of parameters, O, the whole set of observed variables, Z, the whole set of latent variables, we recall that since after E-step, $KL(q||p)=0$, we have:

$$\theta^{new} = \arg\max_{\theta} \mathscr{Q}(\theta,\theta^{old}) = \arg\max_{\theta} \sum_{Z\in\mathscr{Z}} P(Z|O;\theta^{old})\log P(O,Z;\theta),$$

which leads to maximizing the log-likelihood $\mathscr{L}(\theta)$. By taking into account the natural probabilistic contraints on the parameters, the optimization problem is:

$$
\begin{aligned}
\text{max} \quad & \mathscr{L}(\theta) \\
\text{subject to} \quad & \sum_k \pi_k = 1, \forall k, \pi_k \geq 0, \\
& \sum_i A_{ki} = 1, \forall i, k, A_{ki} \geq 0, \\
& (\sum_g b_k(v_g) = 1), \forall k, g, b_k(v_g) \geq 0, \\
& (\sum_d p_k(d) = 1), \forall k, d, p_k(d) \geq 0, \\
& (\sum_l \upsilon_{kl} = 1), \forall k, l, \upsilon_{kl}, \\
& (\sum_l \sum_{l'} \rho_{kll'} = 1), \forall k, l, l', \rho_{kll'} \geq 0,
\end{aligned}
$$

which can be equivalently written using its Lagrangian:

$$
\begin{aligned}
\text{max} \quad & \mathscr{L}(\theta) + \delta(1 - \sum_k \pi_k) + \sum_k \zeta_k(1 - \sum_{k'} A_{kk'}) + \sum_k \eta_k(1 - \sum_g b_k(v_g)) + \sum_k \lambda_k(1 - \sum_d p_k(d)) \\
& + \sum_k \nu_k(1 - \sum_l \upsilon_{kl}) + \sum_k \mu_k(1 - \sum_l \sum_{l'} \rho_{kll'})
\end{aligned}
$$

where the constraints that the parameters are greater or equal to zero have been dropped because it will naturally be handled by the constraint that probabilities sum to one.

Then, in order to find the parameters that maximize the log-likelihood, we compute the partial derivatives of the log-likelihood for each parameter, and cancel the gradient to find the maximum. Let us first compute the update formula of π_k, we have:

$$
\frac{\partial \mathscr{L}(\theta) + \dots}{\partial \pi_k} = -\delta + \sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \frac{\mathbb{1}\{s_1^{(1)} = k\}}{\pi_k}.
$$

Hence,

$$
\pi_k = \frac{1}{\delta} \sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{s_1^{(1)} = k\}. \tag{4.28}
$$

Then,

$$
\frac{\partial \mathscr{L}(\theta) + \dots}{\partial \delta} = 1 - \sum_k \pi_k \tag{4.29}
$$

substituting (4.28) into (4.29), δ is maximal when:

$$\delta = \sum_k \sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{s_1^{(1)} = k\}, \tag{4.30}$$

now substituting back (4.30) into (4.28), we obtain:

$$\begin{aligned} \pi_k &= \frac{\sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{s_1^{(1)} = k\}}{\sum_{k'} \sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{s_1^{(1)} = k'\}} \\ &= \frac{\sum_{Z \in \mathscr{Z}} P(Z|O;\theta^{old}) \mathbb{1}\{s_1^{(1)} = k\}}{\sum_k' \sum_{Z \in \mathscr{Z}} P(Z|O;\theta^{old}) \mathbb{1}\{s_1^{(1)} = k'\}} \\ &= \frac{\sum_{Z \in \mathscr{Z}} P(Z,O;\theta^{old}) \mathbb{1}\{s_1^{(1)} = k\}}{\sum_k' \sum_{Z \in \mathscr{Z}} P(Z,O;\theta^{old}) \mathbb{1}\{s_1^{(1)} = k'\}} \\ &= \frac{P(O_{1:N_\tau}^{(1)} S_1^{(1)} = k)}{\sum_{k'} P(O_{1:N_\tau}^{(1)}, S_1^{(1)} = k')} \\ &= P(S_1^{(1)} = k | O_{1:N_\tau}^{(1)}) \end{aligned} \tag{4.31}$$

which corresponds to the value computed in the E-step of EDHMM in Chapter 1, equation (1.30) for time $j = 1$ and can be seen as the expected number of transition in state k at time $j = 1$.

We now focus on the update formulas of $A_{kk'}$:

$$\frac{\partial \mathscr{L}(\theta) + \dots}{\partial A_{kk'}} = -\zeta_k + \sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \frac{\mathbb{1}\{s_j^{(1)} = k', s_{j-1}^{(1)} = k, f_{j-1} = 1\}}{A_{kk'}},$$

hence $A_{kk'}$ is maximal for,

$$A_{kk'} = \frac{1}{\zeta_k} \sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{s_j^{(1)} = k', s_{j-1}^{(1)} = k, f_{j-1} = 1\}. \tag{4.32}$$

Then we have,

$$\frac{\partial \mathscr{L}(\theta) + \dots}{\partial \zeta_k} = 1 - \sum_{k'} A_{kk'} \tag{4.33}$$

substituting (4.32) into (4.33), ζ_k is maximal when:

$$\zeta_k = \sum_{k'} \sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{s_j^{(1)} = k', s_{j-1}^{(1)} = k, f_{j-1} = 1\}, \tag{4.34}$$

now substituting back (4.34) into (4.32), we obtain:

$$\begin{aligned} A_{kk'} &= \frac{\sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{s_j^{(1)} = k', s_{j-1}^{(1)} = k, f_{j-1} = 1\}}{\sum_{k''} \sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{s_j^{(1)} = k'', s_{j-1}^{(1)} = k, f_{j-1} = 1\}} \\ &= \frac{\sum_{j=1}^{N_\tau} \sum_{Z \in \mathscr{Z}} P(Z, O; \theta^{old}) \mathbb{1}\{s_j^{(1)} = k', s_{j-1}^{(1)} = k, f_{j-1} = 1\}}{\sum_{j=1}^{N_\tau} \sum_{k''} \sum_{Z \in \mathscr{Z}} P(Z, O; \theta^{old}) \mathbb{1}\{s_j^{(1)} = k'', s_{j-1}^{(1)} = k, f_{j-1} = 1\}} \\ &= \frac{\sum_{j=1}^{N_\tau} P(O_{1:N_\tau}^{(1)}, S_j^{(1)} = k', S_{j-1}^{(1)} = k, F_{j-1} = 1)}{\sum_{j=1}^{N_\tau} \sum_{k''} P(O_{1:N_\tau}^{(1)}, S_j^{(1)} = k'', S_{j-1}^{(1)} = k, F_{j-1} = 1)} \\ &= \frac{\sum_{j=1}^{N_\tau} P(S_j^{(1)} = k', S_{j-1}^{(1)} = k, F_{j-1} = 1 | O_{1:N_\tau}^{(1)})}{\sum_{j=1}^{N_\tau} \sum_{k''} P(S_j^{(1)} = k'', S_{j-1}^{(1)} = k, F_{j-1} = 1 | O_{1:N_\tau}^{(1)})} \end{aligned} \tag{4.35}$$

where both the numerator and the denominator are given by ESS (1.30) in EDHMM and can be interpreted as the expected number of transitions from state k to state k' regardingless of time.

Then, the update formulas of $b_k(v_g)$:

$$\frac{\partial \mathscr{L}(\theta) + \ldots}{\partial b_k(v_g)} = -\eta_k + \sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \frac{\mathbb{1}\{o_j^{(1)} = v_g, s_j^{(1)} = k\}}{b_k(v_g)},$$

hence $b_k(v_g)$ is maximal for,

$$b_k(v_g) = \frac{1}{\eta_k} \sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{o_j^{(1)} = v_g, s_j^{(1)} = k\}. \tag{4.36}$$

Then we have,

$$\frac{\partial \mathscr{L}(\theta) + \ldots}{\partial \eta_k} = 1 - \sum_g b_k(v_g), \tag{4.37}$$

substituting (4.36) into (4.37), η_k is maximal when:

$$\eta_k = \sum_g \sum_{Z \in \mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{o_j^{(1)} = v_g, s_j^{(1)} = k\}, \tag{4.38}$$

now substituting back (4.38) into (4.36), we obtain:

$$\begin{aligned} b_k(v_g) &= \frac{\sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{o_j^{(1)} = v_g, s_j^{(1)} = k\}}{\sum_{g'} \sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{o_j^{(1)} = v_{g'}, s_j^{(1)} = k\}} \\ &= \frac{\sum_{j=1}^{N_\tau} \sum_{Z\in\mathscr{Z}} P(Z,O;\theta^{old}) \mathbb{1}\{o_j^{(1)} = v_g, s_j^{(1)} = k\}}{\sum_{j=1}^{N_\tau} \sum_{g'} \sum_{Z\in\mathscr{Z}} P(Z,O;\theta^{old}) \mathbb{1}\{o_j^{(1)} = v_{g'}, s_j^{(1)} = k\}} \\ &= \frac{\sum_{j=1}^{N_\tau} P(O_{1:N_\tau}^{(1)}, S_j^{(1)} = k) \mathbb{1}\{o_j^{(1)} = v_g\}}{\sum_{j=1}^{N_\tau} \sum_{g'} P(O_{1:N_\tau}^{(1)}, S_j^{(1)} = k) \mathbb{1}\{o_j^{(1)} = v_{g'}\}} \\ &= \frac{\sum_{j=1}^{N_\tau} P(S_j^{(1)} = k | O_{1:N_\tau}^{(1)}) \mathbb{1}\{o_j^{(1)} = v_g\}}{\sum_{j=1}^{N_\tau} P(S_j^{(1)} = k | O_{1:N_\tau}^{(1)})} \end{aligned} \tag{4.39}$$

where both the numerator and the denominator are given by ESS (1.30) and which can be interpreted as the expected number of observations v_g while being in state k.

Then, the update formulas of $p_j(d)$:

$$\frac{\partial \mathscr{L}(\theta) + \ldots}{\partial p_j(d)} = -\lambda_k + \sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \frac{\mathbb{1}\{r_j = d, s_j^{(1)} = k, f_{j-1} = 1\}}{p_j(d)}.$$

Hence $p_j(d)$ is maximal for,

$$p_j(d) = \frac{1}{\lambda_k} \sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{r_j = d, s_j^{(1)} = k, f_{j-1} = 1\}. \tag{4.40}$$

Then we have,

$$\frac{\partial \mathscr{L}(\theta) + \ldots}{\partial \lambda_k} = 1 - \sum_d p_j(d) \tag{4.41}$$

substituting (4.40) into (4.41), λ_k is maximal when:

$$\lambda_k = \sum_d \sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{r_j = d, s_j^{(1)} = k, f_{j-1} = 1\}, \tag{4.42}$$

now substituting back (4.42) into (4.40), we obtain:

$$
\begin{aligned}
p_j(d) &= \frac{\sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{r_j = d, s_j^{(1)} = k, f_{j-1} = 1\}}{\sum_{d'} \sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{r_j = d', s_j^{(1)} = k, f_{j-1} = 1\}} \\
&= \frac{\sum_{j=1}^{N_\tau} \sum_{Z\in\mathscr{Z}} P(Z,O;\theta^{old}) \mathbb{1}\{r_j = d, s_j^{(1)} = k, f_{j-1} = 1\}}{\sum_{j=1}^{N_\tau} \sum_{d'} \sum_{Z\in\mathscr{Z}} P(Z,O;\theta^{old}) \mathbb{1}\{r_j = d', s_j^{(1)} = k, f_{j-1} = 1\}} \\
&= \frac{\sum_{j=1}^{N_\tau} P(S_j^{(1)} = k, R_j = d, F_{j-1} = f | O_{1:N_\tau}^{(1)})}{\sum_{d'} \sum_{j=1}^{N_\tau} P(S_j^{(1)} = k, R_j = d', F_{j-1} = f | O_{1:N_\tau}^{(1)})}
\end{aligned}
\tag{4.43}
$$

where both the numerator and denominator are given by ESS (1.32). As we discussed previously, this update formula fits a tabular distribution with $\mathscr{D}$ parameters but which can be reduced bit fitting discrete distributions on top of the frequency table, see Chapter 1, section 1.3.3.

Then, the update formulas of υ_{kl}:

$$
\frac{\partial \mathscr{L}(\theta) + \dots}{\partial \upsilon_{kl}} = -\nu_k + \sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \frac{\mathbb{1}\{\varepsilon_1 = l, s_1^{(1)} = k\}}{\upsilon_{kl}}.
$$

Hence υ_{kl} is maximal for,

$$
\upsilon_{kl} = \frac{1}{\nu_k} \sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{\varepsilon_1 = l, s_1^{(1)} = k\}. \tag{4.44}
$$

Then we have,

$$
\frac{\partial \mathscr{L}(\theta) + \dots}{\partial \nu_k} = 1 - \sum_l \upsilon_{kl} \tag{4.45}
$$

substituting (4.44) into (4.45), ν_k is maximal when:

$$
\nu_k = \sum_l \sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{\varepsilon_1 = l, s_1^{(1)} = k\}, \tag{4.46}
$$

now substituting back (4.46) into (4.44), we obtain:

$$\begin{aligned}
\upsilon_{kl} &= \frac{\sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{\varepsilon_1 = l, s_1^{(1)} = k\}}{\sum_{l'} \sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{\varepsilon_1 = l', s_1^{(1)} = k\}} \\
&= \frac{\sum_{j=1}^{N_\tau} \sum_{Z\in\mathscr{Z}} P(Z,O;\theta^{old}) \mathbb{1}\{\varepsilon_1 = l, s_1^{(1)} = k\}}{\sum_{j=1}^{N_\tau} \sum_{l'} \sum_{Z\in\mathscr{Z}} P(Z,O;\theta^{old}) \mathbb{1}\{\varepsilon_1 = l', s_1^{(1)} = k\}} \\
&= \frac{P(O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}, S_1^{(1)} = k, \varepsilon_1 = l)}{\sum_{l'} P(O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}, S_1^{(1)} = k, \varepsilon_1 = l')} \\
&= P(S_1^{(1)} = k, \varepsilon_1 = l | O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)})
\end{aligned} \tag{4.47}$$

which is given by ESS (4.17) and can be interpreted as the expected number of times the process started (at time $j = 1$) with a delay l from state k.

Then, the update formulas of $\rho_{kll'}$:

$$\frac{\partial \mathscr{L}(\theta) + \dots}{\partial \rho_{kll'}} = -\mu_k + \sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \frac{\mathbb{1}\{\varepsilon_j = l, \varepsilon_{j-1} = l', s_j^{(1)} = k\}}{\rho_{kll'}}$$

Hence $\rho_{kll'}$ is maximal for,

$$\rho_{kll'} = \frac{1}{\mu_k} \sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{\varepsilon_j = l, \varepsilon_{j-1} = l', s_j^{(1)} = k\}. \tag{4.48}$$

Then we have,

$$\frac{\partial \mathscr{L}(\theta) + \dots}{\partial \mu_k} = 1 - \sum_{l} \sum_{l'} \rho_{kll'} \tag{4.49}$$

substituting (4.48) into (4.49), μ_{kl} is maximal when:

$$\mu_k = \sum_{l} \sum_{l'} \sum_{Z\in\mathscr{Z}} Q(Z) \sum_{j=1}^{N_\tau} \mathbb{1}\{\varepsilon_j = l, \varepsilon_{j-1} = l', s_j^{(1)} = k\}, \tag{4.50}$$

now substituting back (4.50) into (4.48), we obtain:

$$\begin{aligned}\rho_{kll'} &= \frac{\sum_{Z\in\mathscr{Z}} Q(Z)\sum_{j=1}^{N_\tau} \mathbb{1}\{\varepsilon_j = l, \varepsilon_{j-1} = l', s_j^{(1)} = k\}}{\sum_{l''}\sum_{l'''}\sum_{Z\in\mathscr{Z}} Q(Z)\sum_{j=1}^{N_\tau} \mathbb{1}\{\varepsilon_j = l'', \varepsilon_{j-1} = l''', s_j^{(1)} = k\}}\\ &= \frac{P(S_j^{(1)} = k, \varepsilon_j = l, \varepsilon_{j-1} = l', |O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)})}{\sum_{l''}\sum_{l'''} P(S_j^{(1)} = k, \varepsilon_j = l'', \varepsilon_{j-1} = l'''|O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)})}\end{aligned} \tag{4.51}$$

which is given by ESS (4.17) and can be interpreted as the expected number of transitions from state k and delay l to delay l' regardingless of time. Similarly to sojourn distribution, it is possible to fit distributions on top of the lag-transition matrix in order to reduce the complexity, see section 1.3 non exhaustive list of possibilities.

Finally, we give the update formula for the covariance matrix Σ_k:

$$\Sigma_k = \frac{\sum_{t=1}^{\tau} P(S_{N_t}^{(1)} = k|O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)}) O_t^{(2)} (O_t^{(2)})^T}{\sum_{t=1}^{\tau} P(S_{N_t}^{(1)} = k|O_{1:N_\tau}^{(1)}, O_{1:\tau}^{(2)})} \tag{4.52}$$

where both the numerator and the denominator are given by ESS (4.19). In our application, $O_{1:\tau}^{(2)}$ corresponds to features (wavelet coefficients) of a multi-channel EEGs leading to a covariance matrix of size $\mathscr{C}\lambda_j \times \mathscr{C}\lambda_j$, where $\mathscr{C}$ is the number of channels and λ_j, the number scales for the wavelet transform (Chapter 3 section 3.2). On the one hand, this results into a high dimension problem. On the other hand, the matrix is probably sparse since it does not only model interactions between channels at the same scale but also between channels at different scales. For an overview of sparse matrix estimation methods such as graphical lasso Friedman et al. (2008), we refer to the review of Fan et al. (2016). It is also interesting to note that the covariance matrix encodes marginal correlations between channels/scales while it might also be of interest to estimate the precision matrix which encodes conditional correlations between pairs of variables given the remaining variables.

2.3 State restoration

The state sequence restoration consists in finding the best state sequence given an observed sequence. In the case of an AHHSMM, we consider that the computation of the state sequence $S_{1:N_\tau}^{(1)}$ remains unchanged compared to EDHMM and is achieved using the general Viterbi HSMM Algorithm, see equation (1.41). This can be justified by the fact that in our EM procedure, see proposition 3, $O^{(2)}_{1:\tau}$ does not interfere in the parameter estimation of the first state sequence, so does it in state sequence restoration.

The novelty arise from the computation of the second state sequence $S^{(2)}_{1:\tau}$ along with the most likely delay. Hence, we define the recursive max product equation, which is the probability to end up in state k for at time t for $S^{(2)}_t$ given the most likely path was previously taken:

$$\delta^{(2)}(k) = \max_{s^{(2)}_{1:t-1}} P(S^{(2)}_{1:t-1} = s^{(2)}_{1:t-1}, S^{(2)}_t = k, O^{(1)}_{1:N_\tau} = o^{(1)}_{1:N_\tau}, O^{(2)}_{1:\tau} = o^{(2)}_{1:\tau}),$$

then, the optimal sequence is computed using the traceback of the two backpointers: one storing the optimal previous state, the other storing the optimal previous lag.

3 AHHSMM in practice

3.1 Implementation issues

In this section, we present implementation issues that are generally encountered in practice. We chose not to develop the related mathematical frameworks in this thesis since it leads to even heavier notations and disturbs the understanding of the model and algorithms. Nonetheless, we provide references that tackle the problem in similar models.

Numerical underflow. So far, we have provided quantities such as forward probabilities, equation (4.20), as a joint distribution of possibly growing number of random variables controlled by the sequences length τ and N_τ respectively. Such quantities rapidly underflow as τ resp. N_τ gets large. To face this practical issue, there exists two possibilities. The first one consists in decomposing quantities using logs and the LogSumExp trick, which is presented in Murphy (2002) in the EDHMM case. The second one consists in computing filtered probabilities, hence $\alpha_j(k,l) = P(O^{(1)}_{1:j}, O^{(2)}_{1:T_j+l-1}, S^{(1)}_j, \varepsilon_j = l, F_j = 1)$ becomes $\alpha(k,l) = P(S^{(1)}_j, \varepsilon_j = l, F_j = 1 | O^{(1)}_{1:j}, O^{(2)}_{1:T_j+l-1})$ which doesn't lead to numerical underflows. Refer to Guédon (2003) for this solution in the EDHMM case.

Saving memory. Devijver (1985) proposed a Forward-only algorithm for inference in HMM. This algorithm relies on a decomposition of the expected sufficient statistics using only forward variables which may be interesting in a memory saving perspective. Guedon and Cocozza-Thivent (1990) then applied it to EDHMM.

Multisequence framework. Similarly, we have considered unique sequences of observations until now in order to keep it simple. Rabiner (1989) shows the updated formulas for HMM in a multisequence framework. The corresponding changes consists in defining forward and backward variables for each sequence in the E-step. Changes in the M-step resides in summing out each quantity per sequence multiplied by a prior that is the likelihood of the sequence.

3.2 Assessing performance

Performance measure. Performance assessment of AHHSMM simply relies on model selection. Model selection has already been introduced in section 3.1.2. We again propose to assess AHHSMM based on information theory criterion such as BIC and ICL.

Alternative models. AHHSMM proposes two new aspects, that is, the asynchronous and the heterogeneity of the data. The asynchronous relationships between two signals has already been explored in the work of Bengio (2003), see Chapter 1 section 2.6, and has shown improvements regarding a simple HMM with synchronized multiple output processes. However, to the best of our knowledge, there has been no DBN proposed to model heterogeneous data. We proposed to assess these facets by comparing four different combinations of models:

1. asynchronous heterogeneous hidden semi-Markov model (AHHSMM),

2. asynchronous homogeneous Hidden semi-Markov model, which difference with AHHSMM is simply to use a standard EM procedure rather than the one proposed in proposition 3. This leads to different expected sufficient statistics, with different forward-backward variables as well as a M-Step which also takes into account $O_1^{(2)} : \tau$ when updating parameters related to the first chain $S_1^{(1)} : N_\tau$,

3. heterogeneous hidden semi-Markov model for which output processes are synchronized. This is a special case of AHHSMM by simply setting the upper limit of the lag to 0, $\mathscr{L} = 0$,

4. hidden semi-Markov model.

3.3 Discussion

Coupled models. AHHSMM may come within the scope of coupled models (Chapter 1 section 2.6) since each signal may be conditioned by a corresponding hidden chain (shared or not). However, coupled models have mainly been used in classification contexts, e.g. to separate two distinct tasks by building a Coupled HMM for each Zhong and Ghosh (2002). This highlights the main drawback of the CHMM: they were not designed for signal segmentation and interpretation, therefore, each channel has its own segmentation and it is difficult to characterize a segment. Obermaier et al. (2001a) who used HMM for EEG segmentation simply assumed that changes of states were due to physiological changes in the patterns of output processes. In AHHSM, we force the EEG segments to be tied up to eye-movement segments which segments EEGs into reading strategies using delayed changes in patterns of eye-movements.

Lag distribution. In section 1.3, we proposed a wide variety of distributions for the lag. Sometimes leading to more parameters, sometimes leading to a much higher complexity in parameter inference. Most of the distributions propose to re-synchronize the EEG signal at every fixation step. However, if the goal is only to perform segmentation, a very simple distribution which is is invariant during a state, i.e. change of strategy, may be adopted. This is explained by the fact that segmentation only takes into account the lag before and after a state transition. Nonetheless, more complicated distributions may be used to explore the behavior of the eye-movement / EEG delay.

Range of influence of output processes. In our application, the low-rate sampling process corresponds to eye-movement features indexed at fixation time steps. The time elapsed between two fixations corresponds to the time of the current fixation plus the time of the outgoing saccade. In the current model, we suppose that the eye-movement output at time $j-1$, $O_{j-1}^{(1)}$, influences EEGs from time $T_{j-1}+\varepsilon_{j-1}$ to $T_j+\varepsilon_j$ that is the beginning of the next fixation plus the delay. However we could suppose that the most interesting part of the signal EEG signal caused by the fixation at time $j-1$ only ranges till time T_j, beginning of the next fixation. Several examples of range of influences are shown in figure 4.2.

Missing values. There are plenty of reasons to take into consideration missing values in the EEG or eye-movement signal. It may be caused by a simple acquisition problem. But it could also be interesting to remove an undesired part of the signal if we focus on

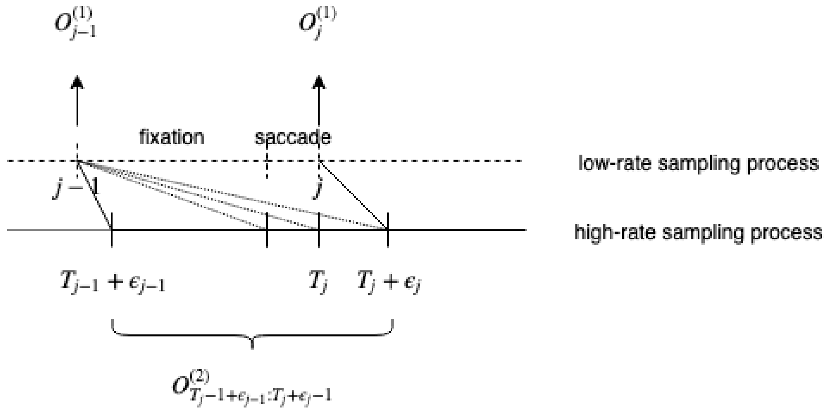

$O^{(1)}_{j-1}$
$O^{(1)}_{j}$
fixation
saccade
low-rate sampling process
$j-1$
j
high-rate sampling process
$T_{j-1}+\epsilon_{j-1}$
T_j
$T_j+\epsilon_j$
$O^{(2)}_{T_j-1+\epsilon_{j-1}:T_j+\epsilon_j-1}$

Figure 4.2: Influence range of low-rate sampling process on high-rate sampling process from a coupled eye-tracking and EEG perspective. The top line represent the eye-movement signal. At time $j-1$, a readmode observation $O^{(1)}_{j-1}$ is sampled and the next one is sampled at time j. The current model fails at taking into account the associated fixation and saccade durations. Indeed the time step of this low-rate sampling process is the fixation. However, it is not necessary for this eye-movement related process, this information could be use to determine the range of influence of observation $O^{(1)}_{j-1}$. In the current model, the influence starts at time $T_{j-1}+\varepsilon_{j-1}$ until $T_j+\varepsilon_j$, i.e. until the beginning of the influence of the next fixation plus a delay, but it could be interesting to stop the influence before. For example, at the beginning of the saccade associated with time $j-1$ or simply the next fixation at time j. These hypothesis are represented by the dotted lines going from one process to another.

a specific range of influence. Moreover, when dealing with wavelets, the deeper we go in the scales, the more the wavelets overlap, which can become a problem around state transitions. This signal could also be deleted around state transition times, to point out the state-specific part of the signal. A possible solution is to considering parameter learning with missing values similarly to the work of Celeux and Durand (2008) with HMMs.

Scaling and technical issues. In practise, we have 10 seconds of acquisition of a 32 multichannel-EEG sampled at 1000Hz on 7 wavelet scales for 15 subjects, 3 text types and 60 texts. Considering that this information is stored on a double of 8 bytes size, the storage requirement is $\approx 4.8384e10$ bytes or $\approx 45Gb$. The standard Expectation-Maximization algorithm requires all the data to be stored at the same time and such amount of RAM is usually not available on computers. This leads us to turn our attention to online learning methods which optimize learn parameter by taking one data point at a time. Cappé (2011) has developed an online EM algorithm for HMM and Bietti et al. (2015) has done a similar work on HSMM. Alas, online algorithms are usually slow since they use one data point at a time but might be solved with mini-batch versions of EM which is currently a hot topic of research in statistical learning, see Nguyen et al. (2019) for mini-batch learning in mixture models with exponential family distributions.

Ongoing experiments. Some ongoing experiments aim at properly evaluating the characteristics of the proposed model (asynchrony and heterogeneity), and evaluating it with its alternatives.

Conclusion

Summary of contributions.

In Chapter 1, we addressed how dynamic Bayesian networks could help to better model, understand and interpret temporal data. A global framework was presented and models were all presented accordingly. We refreshed HSMM's representation, inference, learning and restoration algorithms from a dynamic Bayesian network point of view. We also pointed out the need of a random restart strategy for HSMM.

Hence, in Chapter 2 section 2, we proposed and compared two new strategies to search for a good local maximum likelihood for HSMM with multiple categorical sequences. In accordance with some previous investigations (Biernacki et al., 2003), we showed empirically that a local maximizer with a large attraction domain might sometimes be preferable rather than a spurious local maximizer with a small attraction area. Similarly, we showed that information-theory-based criteria such as BIC and ICL should be used with caution. There is not an absolute better criterion, the choice mainly depends on the aim of the analysis.

In the sequel of Chapter 2, we proposed to identify and characterize reading strategies using HSMMs. This process was rigorously tied together with a methodology proposing data selection, output process selection and model selection. Along the study, two models learned with different strategy were opposed, showing high similarities and encouraging results in terms of interpretation. However, we also presented a drawback of the model on the data: there is a high uncertainty in the restoration which could be reduced by notably incorporating EEGs into a same model.

To this end, in Chapter 3, we first made sure that the model was making sense, not only through his parameters but also thanks to thoroughly chosen covariates. We resorted to covariates of eye movements to demonstrate that segmentation was discriminant enough and managed to relate our strategies with those previously observed in the literature. We also showed that readers could almost be clustered into two distinct groups: careful readers and efficient readers. Then, we measured semantic information gathered all along the trial by readers to show that strategy changes are, at least in part, triggered by target words regarding the given task. Finally, we related reading strategies to contrasted EEG features (wavelet coefficients) and to correlation patterns. We interpreted well-correlated areas as information diffusion and showed that strategies that require deeper sentential integration seemed to involve more connections of temporal areas with parietal, occipital and frontal areas, especially in the theta and alpha bands.

Lastly, in Chapter 4, we proposed to integrate both eye movements and EEGs into a same model to decrease uncertainty regarding segmentation. The originality of the proposition lied on the two characteristics of the signals: they were asynchronous and heterogeneous. EEGs were continuously observed at a fixed (high) sampling rate over say 30 channels. EEG patterns were characterized by a delayed semantic integration with respect to the eye movements, which were univariate discrete measures sampled at a low rate, and non constant (but known) time. To this end, we proposed to exploit asymptotic properties of the estimators to propose an alternative EM procedure. We also proposed an appropriate inference algorithm.

Perspectives

Short-term perspectives

Our very next work will be focused on the finalization of the implementation and experiments of AHHSMMs. As we discussed in Chapter 4, we are facing scalability issues but first experiments can be done on synthetic data to validate: (i) the accuracy of the inference algorithm, (ii) the identifiability of model parameters, (iii) the behavior of our new EM procedure. Afterwards, we will work on subsamples of the real data, at the wavelet scale (i.e., alpha band) that showed the most salient correlations. Electrodes will be clustered into regions of interests and some random subsampling of subjects and text types will be performed.

The next short-term perspective will be to evaluate quantitatively individual variability on EEG's wavelet variance using linear models with random effects. The goal is to compute a random effect per subject to then subtract it to each trial's variance, to be able to better observe activity through the variance with respect to reading strategies and wavelet scales.

Thirdly, we plan on providing more results on more datasets concerning the sequence breaking framework proposed to search high local likelihood values for HSMM. Its assessment on supervised tasks might also help us to better validate the assumptions emitted on the search of spurious local maxima.

Finally, regarding Chapter 3 and EEG a posteriori analysis, we would also like to better assess our graph properties with more small-world properties such as clustering ratio, path length ratio and many more graph indicators.

Long-term perspectives

The most important long-term perspective for this PhD will surely be to ensure the scalability of the proposed AHHSMM. Developing mini-batch versions of EM is a current topic of research and just started to emerge on much simpler models such as Gaussian mixture models (Nguyen et al., 2019).

Afterwards, it will be possible to better characterize the link and the true nature of the eye-movement EEG response delay. The assessment of the lag distribution and the range of influence will be of interest for the reading community.

A final long-term perspective could be to directly integrate random effects such as text and subjects directly into the AHHSMM model to better characterize and quantify their contributions to the variability of the data.

Bibliography

Achard, S., Salvador, R., Whitcher, B., Suckling, J., and Bullmore, E. (2006). A resilient, low-frequency, small-world human brain functional network with highly connected association cortical hubs. *Journal of Neuroscience*, 26(1):63–72.

Akaike, H. (1987). Factor analysis and aic. In *Selected Papers of Hirotugu Akaike*, pages 371–386. Springer.

Allman, E. S., Matias, C., Rhodes, J. A., et al. (2009). Identifiability of parameters in latent structure models with many observed variables. *The Annals of Statistics*, 37(6A):3099–3132.

Asavathiratham, C. (2001). *The influence model: A tractable representation for the dynamics of networked markov chains*. PhD thesis, Massachusetts Institute of Technology.

Barabâsi, A.-L., Jeong, H., Néda, Z., Ravasz, E., Schubert, A., and Vicsek, T. (2002). Evolution of the social network of scientific collaborations. *Physica A: Statistical mechanics and its applications*, 311(3-4):590–614.

Barbu, A. and Zhu, S.-C. (2005). Generalizing swendsen-wang to sampling arbitrary posterior probabilities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1239–1253.

Barbu, V. and Limnios, N. (2006). Maximum likelihood estimation for hidden semi-markov models. *Comptes Rendus Mathematique*, 342(3):201–205.

Barbu, V. S. and Limnios, N. (2009). *Semi-Markov chains and hidden semi-Markov models toward applications: their use in reliability and DNA analysis*, volume 191. Springer Science & Business Media.

Bashashati, A., Fatourechi, M., Ward, R. K., and Birch, G. E. (2007). A survey of signal processing algorithms in brain–computer interfaces based on electrical brain signals. *Journal of Neural engineering*, 4(2):R32.

Bassett, D. S. and Bullmore, E. (2006). Small-world brain networks. *The neuroscientist*, 12(6):512–523.

Basu, S., Choudhury, T., Clarkson, B., and sandy Pentland, A. (2001). Learning human interactions with the influence model. In *Advances in neural information processing systems*.

Batra, D., Yadollahpour, P., Guzman-Rivera, A., and Shakhnarovich, G. (2012). Diverse m-best solutions in markov random fields. In *European Conference on Computer Vision*, pages 1–16. Springer.

Baum, L. E. and Petrie, T. (1966). Statistical inference for probabilistic functions of finite state markov chains. *The annals of mathematical statistics*, 37(6):1554–1563.

Beal, M. J. et al. (2003). *Variational algorithms for approximate Bayesian inference*. university of London London.

Bengio, S. (2003). An asynchronous hidden markov model for audio-visual speech recognition. In *Advances in Neural Information Processing Systems*, pages 1237–1244.

Bengio, S. (2004). Multimodal speech processing using asynchronous hidden markov models. *Information Fusion*, 5(2):81–89.

Bengio, S. and Bengio, Y. (1996). An em algorithm for asynchronous input/output hidden markov models. In *International Conference On Neural Information Processing*, volume 78, pages 328–334. Hong-Kong.

Bengio, Y. (1999). Markovian models for sequential data. *Neural computing surveys*, 2(199):129–162.

Bengio, Y. and Frasconi, P. (1995). An input output hmm architecture. In *Advances in neural information processing systems*, pages 427–434.

Bengio, Y. and Frasconi, P. (1996). Input-output hmms for sequence processing. *IEEE Transactions on Neural Networks*, 7(5):1231–1249.

Bengio, Y., Lauzon, V.-P., and Ducharme, R. (2001). Experiments on the application of iohmms to model financial returns series. *IEEE Transactions on Neural Networks*, 12(1):113–123.

Benzécri, J.-P. et al. (1973). *L'analyse des données*, volume 2. Dunod Paris.

Biernacki, C., Celeux, G., and Govaert, G. (2000). Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE transactions on pattern analysis and machine intelligence*, 22(7):719–725.

Biernacki, C., Celeux, G., and Govaert, G. (2003). Choosing starting values for the em algorithm for getting the highest likelihood in multivariate gaussian mixture models. *Computational Statistics & Data Analysis*, 41(3-4):561–575.

Bietti, A., Bach, F., and Cont, A. (2015). An online em algorithm in hidden (semi-) markov models for audio segmentation and clustering. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 1881–1885. IEEE.

Blanchard, H. E. and Iran-Nejad, A. (1987). Comprehension processes and eye movement patterns in the reading of surprise-ending stories. *Discourse Processes*, 10(1):127–138.

Bliss, C. I. and Fisher, R. A. (1953). Fitting the negative binomial distribution to biological data. *Biometrics*, 9(2):176–200.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Boyen, X. and Koller, D. (1998). Tractable inference for complex stochastic processes. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, pages 33–42. Morgan Kaufmann Publishers Inc.

Brand, M. (1997). Coupled hidden markov models for modeling interacting processes.

Brand, M. (1999). Structure learning in conditional probability models via an entropic prior and parameter extinction. *Neural Computation*, 11(5):1155–1182.

Brand, M., Oliver, N., and Pentland, A. (1997). Coupled hidden markov models for complex action recognition. In *Computer vision and pattern recognition, 1997. proceedings., 1997 ieee computer society conference on*, pages 994–999. IEEE.

Brandt, P. T. and Williams, J. T. (2001). A linear poisson autoregressive model: The poisson ar (p) model. *Political Analysis*, 9(2):164–184.

Brandt, P. T., Williams, J. T., Fordham, B. O., and Pollins, B. (2000). Dynamic modeling for persistent event-count time series. *American Journal of Political Science*, 44(4):823–843.

Brushe, G. D., Mahony, R. E., and Moore, J. B. (1998). A soft output hybrid algorithm for ml/map sequence estimation. *IEEE Transactions on Information Theory*, 44(7):3129–3134.

Burnham, K. P. and Anderson, D. R. (1998). Practical use of the information-theoretic approach. In *Model Selection and Inference*, pages 75–117. Springer.

Burshtein, D. (1996). Robust parametric modeling of durations in hidden markov models. *IEEE Transactions on Speech and Audio Processing*, 4(3):240–242.

Cappé, O. (2011). Online em algorithm for hidden markov models. *Journal of Computational and Graphical Statistics*, 20(3):728–749.

Cappé, O., Moulines, E., and Ryden, T. (2006). *Inference in Hidden Markov Models*. Springer Science & Business Media.

Cappé, O., Moulines, E., and Rydén, T. (2009). Inference in hidden markov models. In *Proceedings of EUSFLAT Conference*, pages 14–16.

Carver, R. P. (1990). *Reading rate: A review of research and theory.* Academic Press.

Carver, R. P. (1992). Reading rate: Theory, research, and practical implications. *Journal of Reading*, 36(2):84–95.

Carver, R. P. (2000). *The causes of high and low reading achievement*. Routledge.

Celeux, G. and Diebolt, J. (1987). A probabilistic teacher algorithm for iterative maximum likelihood estimation. In *1. Conference of the International Federation of Classification Societies*, pages 617–624.

Celeux, G. and Durand, J.-B. (2008). Selecting hidden markov model state number with cross-validated likelihood. *Computational Statistics*, 23(4):541–564.

Celeux, G. and Govaert, G. (1992). A classification em algorithm for clustering and two stochastic versions. *Computational statistics & Data analysis*, 14(3):315–332.

Chaubert-Pereira, F. (2008). *Combinaisons markoviennes et semi-markoviennes de modèles de régression. Application à la croissance d'arbres forestiers.* PhD thesis, Université Montpellier II-Sciences et Techniques du Languedoc.

Chaubert-Pereira, F., Guédon, Y., Lavergne, C., and Trottier, C. (2008). Estimating markov and semi-markov switching linear mixed models with individual-wise random effects. In *Computational Statistics, COMPSTAT'2008, 18th Symposium of IASC*, volume 2, pages 11–18. Physica-Verlag.

Chaubert-Pereira, F., Guédon, Y., Lavergne, C., and Trottier, C. (2010). Markov and semi-markov switching linear mixed models used to identify forest tree growth components. *Biometrics*, 66(3):753–762.

Chen, M.-Y., Kundu, A., and Srihari, S. N. (1993). Variable duration hidden markov model and morphological segmentation for handwritten word recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 600–601. IEEE.

Chiappa, S. and Bengio, S. (2003). Hmm and iohmm modeling of eeg rhythms for asynchronous bci systems. Technical report, IDIAP.

Chuk, T., Chan, A. B., and Hsiao, J. H. (2014). Understanding eye movements in face recognition using hidden markov models. *Journal of vision*, 14(11):8–8.

Cincotti, F., Scipione, A., Timperi, A., Mattia, D., Marciani, A., Millan, J., Salinari, S., Bianchi, L., and Bablioni, F. (2003). Comparison of different feature classifiers for brain computer interfaces. In *First International IEEE EMBS Conference on Neural Engineering, 2003. Conference Proceedings.*, pages 645–647. IEEE.

Clark, S. J. and Perry, J. N. (1989). Estimation of the negative binomial parameter κ by maximum quasi-likelihood. *Biometrics*, pages 309–316.

Cooper, G. F. (1990). The computational complexity of probabilistic inference using bayesian belief networks. *Artificial intelligence*, 42(2-3):393–405.

Coutrot, A., Hsiao, J. H., and Chan, A. B. (2018). Scanpath modeling and classification with hidden markov models. *Behavior research methods*, 50(1):362–379.

Dean, T. and Kanazawa, K. (1989). A model for reasoning about persistence and causation. *Computational intelligence*, 5(2):142–150.

Dechter, R. (1999). Bucket elimination: A unifying framework for reasoning. *Artificial Intelligence*, 113(1-2):41–85.

Dechter, R. and Pearl, J. (1988). Network-based heuristics for constraint-satisfaction problems. In *Search in artificial intelligence*, pages 370–425. Springer.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38.

Devijver, P. A. (1985). Baum's forward-backward algorithm revisited. *Pattern Recognition Letters*, 3(6):369–373.

Dimigen, O., Sommer, W., Hohlfeld, A., Jacobs, A. M., and Kliegl, R. (2011). Coregistration of eye movements and eeg in natural reading: analyses and review. *Journal of experimental psychology: General*, 140(4):552.

Douc, R., Moulines, E., Olsson, J., Van Handel, R., et al. (2011). Consistency of the maximum likelihood estimator for general hidden markov models. *the Annals of Statistics*, 39(1):474–513.

Doucet, A., De Freitas, N., Murphy, K., and Russell, S. (2000). Rao-blackwellised particle filtering for dynamic bayesian networks. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, pages 176–183. Morgan Kaufmann Publishers Inc.

Durand, J.-B. and Guédon, Y. (2014). Quantifying and localizing state uncertainty in hidden markov models using conditional entropy profiles. In *COMPSTAT 2014-21st International Conference on Computational Statistics*, pages 213–221. Université de Genève.

Durand, J.-B. and Guédon, Y. (2016). Localizing the latent structure canonical uncertainty: entropy profiles for hidden markov models. *Statistics and Computing*, 26(1-2):549–567.

Durbin, R., Eddy, S. R., Krogh, A., and Mitchison, G. (1998). *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge university press.

Eddy, S. R. (1998). Profile hidden markov models. *Bioinformatics (Oxford, England)*, 14(9):755–763.

Ehrlich, K. and Rayner, K. (1983). Pronoun assignment and semantic integration during reading: Eye movements and immediacy of processing. *Journal of verbal learning and verbal behavior*, 22(1):75–87.

Engbert, R., Nuthmann, A., Richter, E. M., and Kliegl, R. (2005). Swift: a dynamical model of saccade generation during reading. *Psychological review*, 112(4):777.

Ephraim, Y. and Merhav, N. (2002). Hidden markov processes. *IEEE Transactions on information theory*, 48(6):1518–1569.

Fan, J., Liao, Y., and Liu, H. (2016). An overview of the estimation of large covariance and precision matrices. *The Econometrics Journal*, 19(1):C1–C32.

Farid, M. and Grainger, J. (1996). How initial fixation position influences visual word recognition: A comparison of french and arabic. *Brain and Language*, 53(3):351–368.

Feng, Y., Cheung, G., Tan, W.-t., and Ji, Y. (2011). Hidden markov model for eye gaze prediction in networked video streaming. In *2011 IEEE International Conference on Multimedia and Expo*, pages 1–6. IEEE.

Ferguson, J. (1980). pp. 143–179, variable duration models for speech. In *Proc. of the Symposium on the applications of hidden Markov models to text and speech, JD Ferguson, Ed. Princeton: IDA-CRD*.

Ferri, R., Rundo, F., Bruni, O., Terzano, M. G., and Stam, C. J. (2007). Small-world network organization of functional connectivity of eeg slow-wave activity during sleep. *Clinical neurophysiology*, 118(2):449–456.

Fine, S., Singer, Y., and Tishby, N. (1998). The hierarchical hidden markov model: Analysis and applications. *Machine learning*, 32(1):41–62.

Finesso, L. (1992). Estimation of the order of a finite markov chain. In *Recent Advances in the Mathematical Theory of Systems, Control, and Network Signals, Proc. MTNS-91, H. Kimura and S. Kodama, Eds., Mita Press*, pages 643–645.

Fisher, R. A. (1941). The negative binomial distribution. *Annals of Eugenics*, 11(1):182–187.

Fokianos, K., Rahbek, A., and Tjøstheim, D. (2009). Poisson autoregression. *Journal of the American Statistical Association*, 104(488):1430–1439.

Foreman, L. A. (1992). Generalisation of the viterbi algorithm. *IMA Journal of Management Mathematics*, 4(4):351–367.

Fraser, A. M. (2008). *Hidden Markov models and dynamical systems*, volume 107. Siam.

Frazier, L. and Rayner, K. (1982). Making and correcting errors during sentence comprehension: Eye movements in the analysis of structurally ambiguous sentences. *Cognitive psychology*, 14(2):178–210.

Freese, A. R. (1997). Reading rate and comprehension: Implications for designing computer technology to facilitate reading comprehension. *Computer Assisted Language Learning*, 10(4):311–319.

Frey, A., Ionescu, G., Lemaire, B., López-Orozco, F., Baccino, T., and Guérin-Dugué, A. (2013). Decision-making in information seeking on texts: an eye-fixation-related potentials investigation. *Frontiers in systems neuroscience*, 7:39.

Frey, A., Lemaire, B., Vercueil, L., and Guérin-Dugué, A. (2018). An eye fixation-related potential study in two reading tasks: reading to memorize and reading to make a decision. *Brain topography*, 31(4):640–660.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

Fung, R. and Chang, K.-C. (1990). Weighing and integrating evidence for stochastic simulation in bayesian networks. In *Machine Intelligence and Pattern Recognition*, volume 10, pages 209–219. Elsevier.

Ghahramani, Z. (2001). An introduction to hidden markov models and bayesian networks. *International journal of pattern recognition and artificial intelligence*, 15(01):9–42.

Ghahramani, Z. and Hinton, G. E. (2000). Variational learning for switching state-space models. *Neural computation*, 12(4):831–864.

Ghahramani, Z. and Jordan, M. I. (1996). Factorial hidden markov models. In *Advances in Neural Information Processing Systems*, pages 472–478.

Gilks, W. R., Richardson, S., and Spiegelhalter, D. (1995). *Markov chain Monte Carlo in practice*. Chapman and Hall/CRC.

Giudici, P., Ryden, T., and Vandekerkhove, P. (2000). Likelihood-ratio tests for hidden markov models. *Biometrics*, 56(3):742–747.

Gramfort, A., Luessi, M., Larson, E., Engemann, D. A., Strohmeier, D., Brodbeck, C., Goj, R., Jas, M., Brooks, T., Parkkonen, L., et al. (2013). Meg and eeg data analysis with mne-python. *Frontiers in neuroscience*, 7:267.

Gramfort, A., Luessi, M., Larson, E., Engemann, D. A., Strohmeier, D., Brodbeck, C., Parkkonen, L., and Hämäläinen, M. S. (2014). Mne software for processing meg and eeg data. *Neuroimage*, 86:446–460.

Grandchamp, R. and Delorme, A. (2011). Single-trial normalization for event-related spectral decomposition reduces sensitivity to noisy trials. *Frontiers in psychology*, 2:236.

Grave, E., Bojanowski, P., Gupta, P., Joulin, A., and Mikolov, T. (2018). Learning word vectors for 157 languages. *arXiv preprint arXiv:1802.06893*.

Guédon, Y. (1999). Computational methods for discrete hidden semi-markov chains. *Applied Stochastic Models in Business and Industry*, 15(3):195–224.

Guédon, Y. (2003). Estimating hidden semi-markov chains from discrete sequences. *Journal of Computational and Graphical Statistics*, 12(3):604–639.

Guédon, Y. (2007). Exploring the state sequence space for hidden markov and semi-markov chains. *Computational Statistics & Data Analysis*, 51(5):2379–2409.

Guedon, Y. and Cocozza-Thivent, C. (1990). Explicit state occupancy modelling by hidden semi-markov models: application of derin's scheme. *Computer Speech & Language*, 4(2):167–192.

Hanslmayr, S., Gross, J., Klimesch, W., and Shapiro, K. L. (2011). The role of alpha oscillations in temporal attention. *Brain research reviews*, 67(1-2):331–343.

Hayashi, M. (2003). Hidden markov models to identify pilot instrument scanning and attention patterns. In *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme-System Security and Assurance (Cat. No. 03CH37483)*, volume 3, pages 2889–2896. IEEE.

Hyönä, J., Lorch Jr, R. F., and Kaakinen, J. K. (2002). Individual differences in reading to summarize expository text: Evidence from eye fixation patterns. *Journal of Educational Psychology*, 94(1):44.

Jas, M., Engemann, D., Raimondo, F., Bekhti, Y., and Gramfort, A. (2016). Automated rejection and repair of bad trials in meg/eeg. In *6th International Workshop on Pattern Recognition in Neuroimaging (PRNI)*.

Jas, M., Engemann, D. A., Bekhti, Y., Raimondo, F., and Gramfort, A. (2017). Autoreject: Automated artifact rejection for meg and eeg data. *NeuroImage*, 159:417–429.

Johnson, M. J. and Willsky, A. (2012). The hierarchical dirichlet process hidden semi-markov model. *arXiv preprint arXiv:1203.3485*.

Johnson, M. J. and Willsky, A. S. (2013). Bayesian nonparametric hidden semi-markov models. *Journal of Machine Learning Research*, 14(Feb):673–701.

Johnson, N. L., Kemp, A. W., and Kotz, S. (2005). *Univariate discrete distributions*, volume 444. John Wiley & Sons.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233.

Joulin, A., Grave, E., Bojanowski, P., Douze, M., Jégou, H., and Mikolov, T. (2016a). Fasttext. zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016b). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Juan, A., García-Hernández, J., and Vidal, E. (2004). Em initialisation for bernoulli mixture learning. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 635–643. Springer.

Juang, B.-H. and Rabiner, L. R. (1985). A probabilistic distance measure for hidden markov models. *AT&T technical journal*, 64(2):391–408.

Jurafsky, D. and Martin, J. H. (2009). *Speech and Language Processing (2Nd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Karlis, D. and Xekalaki, E. (2003). Choosing initial values for the em algorithm for finite mixtures. *Computational Statistics & Data Analysis*, 41(3-4):577–590.

Katz, R. W. (1981). On some criteria for estimating the order of a markov chain. *Technometrics*, 23(3):243–249.

Kim, J. and Pearl, J. (1983). A computational model for causal and diagnostic reasoning in inference systems. In *International Joint Conference on Artificial Intelligence*, pages 0–0.

Klimesch, W. (1996). Memory processes, brain oscillations and eeg synchronization. *International journal of psychophysiology*, 24(1-2):61–100.

Koller, D., Friedman, N., and Bach, F. (2009). Probabilistic graphical models: principles and techniques.

Kolouri, S., Park, S. R., Thorpe, M., Slepcev, D., and Rohde, G. K. (2017). Optimal mass transport: Signal processing and machine-learning applications. *IEEE Signal Processing Magazine*, 34(4):43–59.

Kolouri, S., Rohde, G. K., and Hoffmann, H. (2018). Sliced wasserstein distance for learning gaussian mixture models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3427–3436.

Kulesza, A., Taskar, B., et al. (2012). Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning*, 5(2–3):123–286.

Kupiec, J. (1992). Robust part-of-speech tagging using a hidden markov model. *Computer Speech & Language*, 6(3):225–242.

Kwon, J. and Murphy, K. (2000). Modeling freeway traffic with coupled hmms. Technical report, Technical report, Univ. California, Berkeley.

Lee, H. and Choi, S. (2003). Pca+ hmm+ svm for eeg pattern classification. In *Seventh International Symposium on Signal Processing and Its Applications, 2003. Proceedings.*, volume 1, pages 541–544. IEEE.

Lemaire, B., Guérin-Dugué, A., Baccino, T., Chanceaux, M., and Pasqualotti, L. (2011). A cognitive computational model of eye movements investigating visual strategies on textual material. In *33rd annual meeting of the Cognitive Science Society (CogSci 2011)*, pages 1146–1151. Cognitive Science Society.

Leroux, B. G. (1992). Maximum-likelihood estimation for hidden markov models. *Stochastic processes and their applications*, 40(1):127–143.

Leu, D. J., Forzani, E., Rhoads, C., Maykel, C., Kennedy, C., and Timbrell, N. (2015). The new literacies of online research and comprehension: Rethinking the reading achievement gap. *Reading Research Quarterly*, 50(1):37–59.

Liechty, J., Pieters, R., and Wedel, M. (2003). Global and local covert visual attention: Evidence from a bayesian hidden markov model. *Psychometrika*, 68(4):519–541.

Ljolje, A. and Levinson, S. E. (1991). Development of an acoustic-phonetic hidden markov model for continuous speech recognition. *IEEE Transactions on signal processing*, 39(1):29–39.

Luck, S. J. (2014). *An introduction to the event-related potential technique*. MIT press.

MacKay, D. J. (1997). Ensemble learning for hidden markov models. Technical report, Citeseer.

MacKinnon, J. G. (2009). Bootstrap hypothesis testing. *Handbook of computational econometrics*, 183:213.

Mallat, S. (1999). *A wavelet tour of signal processing*. Elsevier.

Marhasev, E., Hadad, M., and Kaminka, G. A. (2006). Non-stationary hidden semi markov models in activity recognition. In *Proceedings of the AAAI Workshop on Modeling Others from Observations (MOO-06)*.

Marroquin, J., Mitter, S., and Poggio, T. (1987). Probabilistic solution of ill-posed problems in computational vision. *Journal of the american statistical association*, 82(397):76–89.

McLachlan, G. and Krishnan, T. (2007). *The EM algorithm and extensions*, volume 382. John Wiley & Sons.

Meilă, M. and Heckerman, D. (2001). An experimental comparison of model-based clustering methods. *Machine learning*, 42(1-2):9–29.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., and Joulin, A. (2018). Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Mitchell, C., Harper, M., and Jamieson, L. (1995). On the complexity of explicit duration hmm's. *IEEE transactions on speech and audio processing*, 3(3):213–217.

Mitchell, C. D. and Jamieson, L. H. (1993). Modeling duration in a hidden markov model with the exponential family. In *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on*, volume 2, pages 331–334. IEEE.

Mochihashi, D. and Sumita, E. (2008). The infinite markov model. In *Advances in neural information processing systems*, pages 1017–1024.

Murphy, K. and Weiss, Y. (2001). The factored frontier algorithm for approximate inference in dbns. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 378–385. Morgan Kaufmann Publishers Inc.

Murphy, K. P. (2002). Hidden semi-markov models (hsmms). *unpublished notes*, 2.

Murphy, K. P. and Russell, S. (2002). *Dynamic bayesian networks: representation, inference and learning*. PhD thesis, University of California, Berkeley.

Murphy, K. P., Weiss, Y., and Jordan, M. I. (1999). Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 467–475. Morgan Kaufmann Publishers Inc.

Nagarajan, R., Scutari, M., and Lèbre, S. (2013). Bayesian networks in r. *Springer*, 122:125–127.

Nagel, B. J., Herting, M. M., Maxwell, E. C., Bruno, R., and Fair, D. (2013). Hemispheric lateralization of verbal and spatial working memory during adolescence. *Brain and cognition*, 82(1):58–68.

Natarajan, P. and Nevatia, R. (2007). Coupled hidden semi markov models for activity recognition. In *null*, page 10. IEEE.

Nefian, A. V., Liang, L., Pi, X., Liu, X., and Murphy, K. (2002). Dynamic bayesian networks for audio-visual speech recognition. *EURASIP Journal on Advances in Signal Processing*, 2002(11):783042.

Neuper, C. and Klimesch, W. (2006). *Event-related dynamics of brain oscillations*, volume 159. Elsevier.

Nguyen, H. D., Forbes, F., and McLachlan, G. J. (2019). Mini-batch learning of exponential family finite mixture models. *arXiv preprint arXiv:1902.03335*.

Nilsson, D. and Goldberger, J. (2001). Sequentially finding the n-best list in hidden markov models. In *Proceedings of the 17th international joint conference on Artificial intelligence-Volume 2*, pages 1280–1285. Morgan Kaufmann Publishers Inc.

Obermaier, B., Guger, C., Neuper, C., and Pfurtscheller, G. (2001a). Hidden markov models for online classification of single trial eeg data. *Pattern recognition letters*, 22(12):1299–1309.

Obermaier, B., Munteanu, C., Rosa, A., and Pfurtscheller, G. (2001b). Asymmetric hemisphere modeling in an offline brain-computer interface. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 31(4):536–540.

Obermaier, B., Neuper, C., Guger, C., and Pfurtscheller, G. (2001c). Information transfer rate in a five-classes brain-computer interface. *IEEE Transactions on neural systems and rehabilitation engineering*, 9(3):283–288.

O'regan, J., Lévy-Schoen, A., Pynte, J., and Brugaillère, B. é. (1984). Convenient fixation location within isolated words of different length and structure. *Journal of Experimental Psychology: Human Perception and Performance*, 10(2):250.

O'Connell, J., Højsgaard, S., et al. (2011). Hidden semi markov models for multiple observation sequences: The mhsmm package for r. *Journal of Statistical Software*, 39(4):1–22.

Park, T., Eckley, I. A., and Ombao, H. C. (2014). Estimating time-evolving partial coherence between signals via multivariate locally stationary wavelet processes. *IEEE Transactions on Signal Processing*, 62(20):5240–5250.

Pearl, J. (2009). *Causality*. Cambridge university press.

Percival, D. B. and Walden, A. T. (2006). *Wavelet methods for time series analysis*, volume 4. Cambridge university press.

Peyhardi, J., Trottier, C., and Guédon, Y. (2016). Partitioned conditional generalized linear models for categorical responses. *Statistical Modelling*, 16(4):297–321.

Porway, J. and Zhu, S.-C. (2011). Cˆ 4: Exploring multiple solutions in graphical models by cluster sampling. *IEEE transactions on pattern analysis and machine intelligence*, 33(9):1713–1727.

Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Ramesh, P. and Wilpon, J. G. (1992). Modeling state durations in hidden markov models for automatic speech recognition. In *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, volume 1, pages 381–384. IEEE.

Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological bulletin*, 124(3):372.

Rayner, K., Pollatsek, A., Ashby, J., and Clifton Jr, C. (2012). *Psychology of reading*. Psychology Press.

Rayner, K. and Well, A. D. (1996). Effects of contextual constraint on eye movements in reading: A further examination. *Psychonomic Bulletin & Review*, 3(4):504–509.

Reichle, E. D., Pollatsek, A., Fisher, D. L., and Rayner, K. (1998). Toward a model of eye movement control in reading. *Psychological review*, 105(1):125.

Reichle, E. D., Pollatsek, A., and Rayner, K. (2012). Using ez reader to simulate eye movements in nonreading tasks: A unified framework for understanding the eye–mind link. *Psychological review*, 119(1):155.

Reichle, E. D., Rayner, K., and Pollatsek, A. (2003). The ez reader model of eye-movement control in reading: Comparisons to other models. *Behavioral and brain sciences*, 26(4):445–476.

Reichle, E. D. and Reingold, E. M. (2013). Neurophysiological constraints on the eye-mind link. *Frontiers in Human Neuroscience*, 7:361.

Rezek, I., Gibbs, M., and Roberts, S. J. (2002). Maximum a posteriori estimation of coupled hidden markov models. *Journal of VLSI signal processing systems for signal, image and video technology*, 32(1-2):55–66.

Rezek, I. and Roberts, S. (2000a). A comparison of bayesian and maximum likelihood learning of coupled hidden markov models. *IEE Proc. Sci. Technol. Measur*, 147(6):345–350.

Rezek, I. and Roberts, S. J. (2000b). Estimation of coupled hidden markov models with application to biosignal interaction modelling. In *Neural Networks for Signal Processing X. Proceedings of the 2000 IEEE Signal Processing Society Workshop (Cat. No. 00TH8501)*, volume 2, pages 804–813. IEEE.

Rezek, L., Sykacek, P., and Roberts, S. J. (2000). Coupled hidden markov models for biosignal interaction modelling. In *2000 First International Conference Advances in Medical Signal and Information Processing (IEE Conf. Publ. No. 476)*, pages 54–59. IET.

Rimey, R. D. and Brown, C. M. (1991). Controlling eye movements with hidden markov models. *International Journal of Computer Vision*, 7(1):47–65.

Ross, G., Jones, R., Kempton, R., Laukner, F., Payne, R., Hawkins, D., and White, R. (1980). *MLP: maximum likelihood program*. Rothamsted Experimental Station.

Ross, G. and Preece, D. (1985). The negative binomial distribution. *The Statistician*, pages 323–335.

Salojärvi, J., Puolamäki, K., and Kaski, S. (2005). Implicit relevance feedback from eye movements. In *International Conference on Artificial Neural Networks*, pages 513–518. Springer.

Salvucci, D. D. and Goldberg, J. H. (2000). Identifying fixations and saccades in eye-tracking protocols. In *Proceedings of the 2000 symposium on Eye tracking research & applications*, pages 71–78. ACM.

Sauseng, P., Klimesch, W., Schabus, M., and Doppelmayr, M. (2005). Fronto-parietal eeg coherence in theta and upper alpha reflect central executive functions of working memory. *International journal of Psychophysiology*, 57(2):97–103.

Schwarz, G. et al. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464.

Schwarz, R. and Chow, Y.-L. (1990). The n-best algorithm: An efficient and exact procedure for finding the n most likely hypotheses. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 81–84.

Seidkhani, H., Nikolaev, A. R., Meghanathan, R. N., Pezeshk, H., Masoudi-Nejad, A., and van Leeuwen, C. (2017). Task modulates functional connectivity networks in free viewing behavior. *NeuroImage*, 159:289–301.

Sereno, S. C. and Rayner, K. (1992). Fast priming during eye fixations in reading. *Journal of Experimental Psychology: Human Perception and Performance*, 18(1):173.

Shachter, R. D. and Peot, M. A. (1990). Simulation approaches to general probabilistic inference on belief networks. In *Machine Intelligence and Pattern Recognition*, volume 10, pages 221–231. Elsevier.

Shimojo, S., Simion, C., Shimojo, E., and Scheier, C. (2003). Gaze bias both reflects and influences preference. *Nature neuroscience*, 6(12):1317.

Simola, J., Salojärvi, J., and Kojo, I. (2008). Using hidden markov model to uncover processing states from eye movements in information search tasks. *Cognitive systems research*, 9(4):237–251.

Sin, B. and Kim, J. H. (1995). Nonstationary hidden markov model. *Signal Processing*, 46(1):31–46.

Smit, D. J., Stam, C. J., Posthuma, D., Boomsma, D. I., and De Geus, E. J. (2008). Heritability of "small-world" networks in the brain: A graph theoretical analysis of resting-state eeg functional connectivity. *Human brain mapping*, 29(12):1368–1378.

Smyth, P. (1997). Clustering sequences with hidden markov models. In *Advances in neural information processing systems*, pages 648–654.

Smyth, P., Heckerman, D., and Jordan, M. I. (1997). Probabilistic independence networks for hidden markov probability models. *Neural computation*, 9(2):227–269.

Soheily-Khah, S., Douzal-Chouakria, A., and Gaussier, E. (2016). Generalized k-means-based clustering for temporal data under weighted and kernel time warp. *Pattern Recognition Letters*, 75:63–69.

Spiegelhalter, D. J. and Lauritzen, S. L. (1990). Sequential updating of conditional probabilities on directed graphical structures. *Networks*, 20(5):579–605.

Stolcke, A. and Omohundro, S. (1993). Hidden markov model induction by bayesian model merging. In *Advances in neural information processing systems*, pages 11–18.

Strogatz, S. H. (2001). Exploring complex networks. *nature*, 410(6825):268.

Tu, Z. and Zhu, S.-C. (2002). Image segmentation by data-driven markov chain monte carlo. *IEEE Transactions on pattern analysis and machine intelligence*, 24(5):657–673.

van Handel, R. (2011). On the minimal penalty for markov order estimation. *Probability theory and related fields*, 150(3-4):709–738.

Varga, A. and Moore, R. (1990). Hidden markov model decomposition of speech and noise. In *Acoustics, Speech, and Signal Processing, 1990. ICASSP-90., 1990 International Conference on*, pages 845–848. IEEE.

Vaseghi, S. (1991). Hidden markov models with duration-dependent state transition probabilities (speech recognition). *Electronics letters*, 27(8):625–626.

Vaseghi, S. (1995). State duration modelling in hidden markov models. *Signal processing*, 41(1):31–41.

Verma, T. and Pearl, J. (1990). Causal networks: Semantics and expressiveness. In *Machine Intelligence and Pattern Recognition*, volume 9, pages 69–76. Elsevier.

Viterbi, A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE transactions on Information Theory*, 13(2):260–269.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world'networks. *nature*, 393(6684):440.

Whitcher, B., Guttorp, P., and Percival, D. B. (2000). Wavelet analysis of covariance with application to atmospheric time series. *Journal of Geophysical Research: Atmospheres*, 105(D11):14941–14962.

Wise, M. (1946). The use of the negative binomial distribution in an industrial sampling problem. *Supplement to the Journal of the Royal Statistical Society*, 8(2):202–211.

Woodman, G. F. (2010). A brief introduction to the use of event-related potentials in studies of perception and attention. *Attention, Perception, & Psychophysics*, 72(8):2031–2046.

Wutz, A., Melcher, D., and Samaha, J. (2018). Frequency modulation of neural oscillations according to visual task demands. *Proceedings of the National Academy of Sciences*, 115(6):1346–1351.

Yang, S. and McConkie, G. W. (2005). New directions in theories of eyemovement control during reading. *Cognitive processes in eye guidance*, pages 105–130.

Yu, S.-Z. (2010). Hidden semi-markov models. *Artificial intelligence*, 174(2):215–243.

Yu, S.-Z. (2015). *Hidden Semi-Markov Models: Theory, Algorithms and Applications*. Morgan Kaufmann.

Yu, S.-Z. and Kobayashi, H. (2003). An efficient forward-backward algorithm for an explicit-duration hidden markov model. *IEEE signal processing letters*, 10(1):11–14.

Yu, S.-Z. and Kobayashi, H. (2006). Practical implementation of an efficient forward-backward algorithm for an explicit-duration hidden markov model. *IEEE Transactions on Signal Processing*, 54(5):1947–1951.

Zhang, N. L. and Poole, D. (1994). A simple approach to bayesian network computations. In *Proc. of the Tenth Canadian Conference on Artificial Intelligence*.

Zhong, S. and Ghosh, J. (2001). A new formulation of coupled hidden markov models. *Dept. Elect. Comput. Eng., Univ. Austin, Austin, TX, USA*.

Zhong, S. and Ghosh, J. (2002). Hmms and coupled hmms for multi-channel eeg classification. In *proceedings of the IEEE international joint conference on neural networks*, volume 2, pages 1254–1159.

Zweig, G. (1996). A forward-backward algorithm for inference in bayesian networks and an empirical comparison with hmms. *Master Thesis, UC Berkeley*.

Appendix A

Descriptive statistics on eye-movement dataset

Subject	Fixation Duration	Saccade amplitude	#Fixations per trial
s01	192.6 ± 70.6	132.1 ± 98.1	16.2 ± 6.3
s02	155.3 ± 47.4	116.2 ± 103.9	15.5 ± 7.3
s04	189.4 ± 68.1	143.6 ± 95.3	31 ± 9.3
s05	176.4 ± 54.3	114 ± 96.3	19.6 ± 7.8
s06	176.9 ± 67.7	120.9 ± 112.7	18.8 ± 7.9
s07	175 ± 48.3	172.8 ± 98.4	10.8 ± 4.5
s08	163.4 ± 53.6	136.6 ± 92.3	16.4 ± 7
s10	154 ± 39.4	136.3 ± 91.6	16.2 ± 7.3
s13	209.6 ± 78.6	118.1 ± 115.2	17.5 ± 5.9
s14	243 ± 92.6	116 ± 94	15.9 ± 6.5
s17	177.8 ± 51.7	145.4 ± 103.5	20.9 ± 10.8
s18	200.3 ± 66.9	140 ± 87.3	12.1 ± 6
s19	157.2 ± 40.3	169.3 ± 99.8	13.4 ± 6.4
s20	217.5 ± 73.4	140.3 ± 95.9	11.5 ± 6.3
s21	183.4 ± 50.7	148.4 ± 84.3	11.7 ± 4.4
Grand Total	184 ± 66	135 ± 100	17 ± 9

Table A.1: Per subject average (mean ± std) fixation durations, saccade amplitudes and number of fixations.

Subject	Readmode				
	long regression	regression	refixation	progression	long progression
s01	0.03	0.02	0.28	0.21	0.45
s02	0.01	0	0.37	0.25	0.37
s04	0.15	0.05	0.2	0.18	0.42
s05	0.03	0.02	0.36	0.22	0.36
s06	0.02	0.01	0.38	0.26	0.33
s07	0.06	0.02	0.1	0.23	0.58
s08	0.11	0.02	0.22	0.22	0.44
s10	0.08	0.02	0.23	0.23	0.44
s13	0.02	0	0.4	0.23	0.34
s14	0.09	0.03	0.31	0.19	0.38
s17	0.06	0.03	0.22	0.24	0.44
s18	0.08	0.04	0.15	0.22	0.5
s19	0.04	0.01	0.13	0.23	0.58
s20	0.09	0.04	0.26	0.18	0.44
s21	0.09	0.03	0.13	0.16	0.58
Grand Total	0.07	0.02	0.26	0.22	0.43

Table A.2: Per subject readmode frequencies. Long regression (Bwd++): more than one word skipped with a backward saccade, regression (Bwd+): one word skipped with a backward saccade, short progression (Fwd+): one word skipped with a forward saccade, long progression (Fwd++): more than one word skipped with a forward saccade.

Subject	Text Type		
	UR	HR	MR
s01	0.95	0.96	0.53
s02	1.00	0.98	0.38
s04	0.66	0.81	0.84
s05	0.97	0.88	0.48
s06	0.97	0.95	0.44
s07	1.00	0.95	0.34
s08	1.00	0.92	0.19
s10	1.00	1.00	0.36
s13	0.86	0.88	0.56
s14	0.90	0.96	0.71
s17	1.00	0.88	0.61
s18	1.00	0.93	0.41
s19	0.98	0.97	0.32
s20	1.00	0.78	0.39
s21	1.00	0.81	0.35
Total	0.93	0.91	0.49

Table A.3: Answer rate per subject and per text. Note that there is no good answer for texts MR as it is ambiguous. UR: Unrelated texts, HR: Highly related texts, MR: Moderately related texts.

Text Type	Mean Fixation Duration	Mean Saccade Amplitude	Mean no. of Fixations per trial
UR	181.9 ± 65.6	132.2 ± 97.9	14.3 ± 7.9
HR	185.8 ± 67	135 ± 100.5	15.3 ± 7.9
MR	184.2 ± 66.7	137.3 ± 102	20.1 ± 8.6
Grand Total	184 ± 66.5	135.1 ± 100.4	16.6 ± 8.5

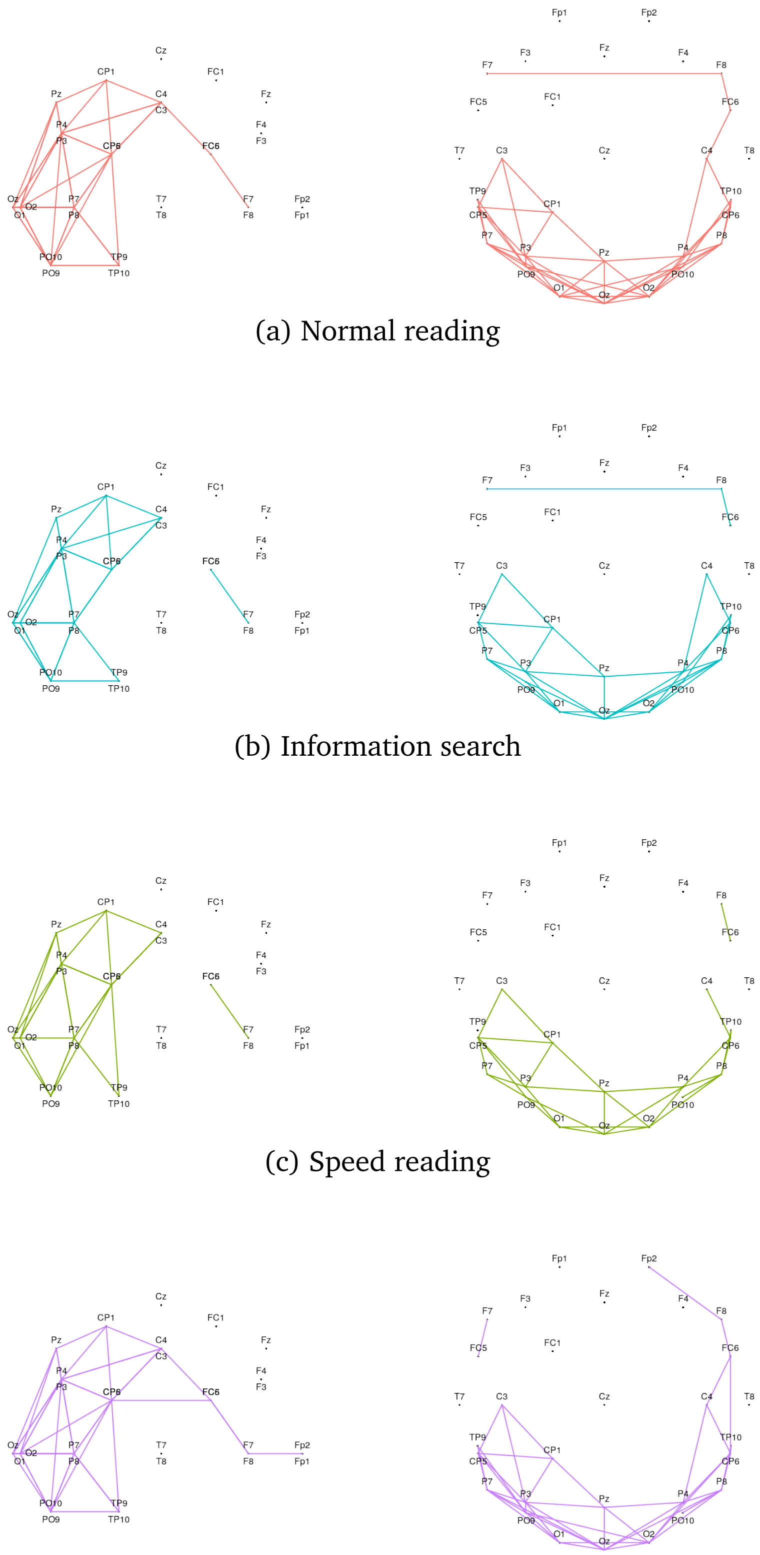
Table A.4: Per text type average (mean ± std) fixation durations, saccade amplitudes and number of fixations. UR: Unrelated texts, HR: Highly related texts, MR: Moderately related texts.

Text Type	Readmode				
	long regression	regression	refixation	progression	long progression
UR	0.06	0.02	0.25	0.22	0.45
HR	0.06	0.02	0.26	0.23	0.42
MR	0.07	0.02	0.26	0.21	0.43
Grand Total	0.07	0.02	0.26	0.22	0.43

Table A.5: Readmode frequencies per text type. UR: Unrelated texts, HR: Highly related texts, MR: Moderately related texts.

Appendix B

Anatomical maps for scale 5 (beta band)



(a) Normal reading

(b) Information search

(c) Speed reading

(d) Slow confirmation

Figure B.1: Anatomical maps (left: sagittal view, right: top view) per reading strategy for wavelet scale 5 (β band) with thresholded covariance at 0.54. Left map is a sagittal view, right map is a top view.