République Algérienne Démocratique et Populaire.
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique.

Université A-Mira de Béjaïa

Faculté des Sciences Exactes

Département d'Informatique

ÉCOLE DOCTORALE RÉSEAUX ET SYSTÈMES DISTRIBUÉS

Mémoire de Magistère

En Informatique

OPTION : : RÉSEAUX ET SYSTÈMES DISTRIBUÉS

Présentée par

GHIDOUCHE Kahina

Etude et analyse des données temporelles par un processus à base d' HMMs

JURY

Président	Mr. AIT SAIDI Ahmed	Professeur. Université de Bejaïa.
Rapporteur	Mr. KECHADI Mohand-Tahar	Professeur. University College Dublin.
Co-rapporteur	Mr. TARI Abdelkamel	M.C.A. Université de Bejaïa.
Examinatrice	Mme. BENATCHABA Karima	Professeur. ESI, Alger.
Examinatrice	Mme. Nader Fahima	M.C.A. ESI, Alger.
Invité	Mr. DAHAMNI Foudil	M.A.A. ESI, Alger.

Promotion 2008-2009

La recherche procède par des moments distincts et durables, intuition, aveuglement, exaltation et fièvre. Elle aboutit un jour à cette joie,et connaît cette joie celui qui a vécu des moments singuliers.

par *Albert Einstein*

Remerciement

Merci à tous ! Merci

à tous ! Merci à tous !

Table des matières

Liste des acronymes

AR	**A**uto **R**egressive
ARMA	**A**uto **R**egressive **M**oving **A**verage
ARIMA	**A**uto **R**egressive **I**ntegration **M**oving Average
DTW	**D**ynamic **T**ime **W**rapping
EM	**E**xpectation **M**aximization
FDBA	**F**eature **D**ata **B**ased **A**pproach
MBA	**M**odel **B**ased **A**pproach
MMC	**M**odèle **M**arkov **C**aché
RDBA	**R**aw **D**ata **B**ased **A**pproach
SSS	**S**uccessive **S**tate **S**plitting

Table des figures

Liste des tableaux

Liste des algorithmes

Introduction générale

La plupart des systèmes et des phénomènes du monde réels sont de nature dynamiques. À cet effet, une étude scientifique importante a porté sur les moyens de les caractériser, les comprendre, les analyser et modéliser leurs comportements. Ces études ont généré plusieurs modèles qui sont utiles dans de nombreux domaines. Par exemple, pour les systèmes économiques, les modèles financiers construits à partir de données historiques peuvent être utilisés pour l'analyse de prévision et de tendance sur divers indices économiques. Pour les domaines médicaux, le développement de modèles pour différent comportements des patients infectés par une maladie chronique est crucial a fin de leur éviter d'éventuelle complication. Ces processus sont devenus plus complexes, et les domaines d'application s'étendent au-delà des sciences physiques et de l'ingénierie. De plus, nos connaissances sur les domaines et les théories qui régissent le comportement des processus sont incomplète. Les progrès de l'informatique et d'outils de stockage ont rendu possible le recueil et le stockage de grandes quantités de données collectée à travers le temps. La question est de savoir comment pouvons-nous apprendre d'avantage sur les processus et les phénomènes impliqués à travers des modèles utilisant des données recueillies.

La construction de modèles, ou la création des structures de données à l'aide de techniques d'analyse des données est une tâche complexe. Ce mémoire est une étape vers l'analyse et la modélisation de données temporelles à l'aide de techniques de clustering. Ces techniques tentent de créer des clusters homogènes et séparés. Le terme "homogène" signifie que les éléments d'une classe sont les plus proches possible les uns des autres. Le terme "séparé" veut dire qu'il y a un maximum d'écart entre les classes. La technique de clustering a été largement utilisée dans diveres domaines pour dextraire des connaissances ne sont pas évidentes, ou bien elles sont cachés ou incomplètes [Biswas et al, 95] [Cheeseman et al, 96].

Dans le passé, le concept de clustering de données est décrit par des caractéristiques statiques [Fisher,87],[Jain, 88],[Cheeseman et al, 96],[Biswas et al, 95], négligeant ainsi le caractère dynamique des données. Pour les systèmes dynamiques, leur comportement est mieux décrit par les caractéristiques dont les valeurs peuvent changer de façon significative au cours de la période d'observation. Le clustering des données temporelles est une tâche plus complexe que le clustering des données statiques car la dimensionnalité des données est beaucoup plus importante dans le cas temporel. La complexité de la description et l'interprétation des modèles temporels ou dynamiques d'un processus augmentent également par rapport aux modèles statiques. Dans la section suivante, nous présentons un exemple de motivation sur l'application de méthodes de clustering pour l'analyse des données temporelles.

Exemple de motivation

Dans cet exemple, notre domaine d'intérêt est le comportement dynamique des patients diabétiques aux cours du temps pour capturer l'état physiologique du patient. Nous utilisons les caractéristiques pertinentes et temporelles comme la variation du taux de glycémie dans le sang, la détection de protéines dans les urines, etc. Parfois, ces caractéristiques temporelles ne sont pas suffisantes pour détecter l'état du patient, car elles n'offrent pas assez de connaissances dans ce domaine. Pour cela, l'élaboration d'une méthode de clustering et de modélisation de données temporelles s'avère nécessaire.

L'objectif de l'étude du comportement des patients diabétiques est de trouver des outils d'analyse afin de prévenir et soigner le diabète et ses complications et d'améliorer durablement l'état médical d'un diabétique permettant ainsi d'assurer une bonne qualité de vie. L'application des méthodes de clustering impliquerait tout d'abord la modélisation de données temporelles pour ce problème, puis une étude de grandes quantités de données sur les patients pour générer des clusters de patients de différents comportements. De tels modèles peuvent aider les médecinsà définir le bon traitement pour chaque groupe de patients. En effet, l'étude de ces modèles permet d'identifier les caractéristiques critiques qui définissent les conditions des patients. Ces caractéristiques peuvent bénéficier d'une attention particulière lors de la surveillance et le diagnostic des patients. En outre, les clusters générés peuvent également servir à la prévision en identifiant la meilleure correspondance d'un modèle de cluster pour un nouveau patient.

Des problèmes analogues existent dans d'autres domaines où les techniques de clustering des données temporelles peuvent être utilisées. Par exemple, les banques peuvent créer des profils

client pour suivre le comportement de leurs transactions. Un changement de profil peut laisser croire à une utilisation frauduleuse du compte bancaire du client. De même, l'approche est également applicable aux systèmes d'ingénierie où les modèles construits à partir de mesures faites sur les systèmes peuvent être utilisés pour aider les diverses tâches, telles que la surveillance et la localisation des pannes.

Une analyse efficace de ces types de données comprend non seulement leur partitionnement en clusters homogènes, mais aussi leur modélisation au sein de chaque cluster offre une bonne interprétation et rend ainsi les modèles utiles pour les experts du domaine dans la résolution de problèmes complexes. Il est important que les modèles construits à partir de l'analyse décrivent au mieux les phénomènes associés à chacun des clusters.

La problématique étudiée

L'objectif de ce mémoire est de développer une méthodologie de clustering et de modélisation des données temporelles. Ceci est possible en classant les données en groupes homogènes. Puis construire des modèles interprétables pour les données dans chaque groupe. En particulier, nous étudions l'utilisation de modèles de Markov cachés (MMC) pour le clustering de données temporelles.

L'avantage principal de la méthodologie MMC réside dans le fait qu'elle est une approche à base des probabilités. Les états d'un MMC ne sont pas directement observables (d'où le terme "caché") mais les valeurs de ces états "cachés" peuvent être estimées à partir des données qui sont directement observables dans le système. Dans notre problème de modélisation, les états cachés d'un MMC correspondent aux états valides du système dynamique considéré et les transitions probabilistes entre ces états correspondent au comportement non déterministe du système.

Les travaux antérieurs sur le clustering utilisant la modélisation MMC ont principalement été développés pour des systèmes de reconnaissance vocale [Rabiner, 89]. Le but du clustering dans ce domaine est d'améliorer la précision de reconnaissance vocale en construisant des MMCs multiples pour le même le mot (les syllabes d'un mot compososite) pour tenir compte des variations de prononciation parmi les différents orateurs. Dans ces applications, les topologies MMC utilisées dans le processus de clustering ont une taille prédéfinie par des experts linguistiques avant d'appliquer les algorithmes de regroupement.

Ce mémoire propose une approche plus générale pour le clustering des données temporelles

à base des MMCs où nous ne précisons pas la topologie de MMC, notamment la taille d'un MMC (le nombre d'états dans un MMC). Notre approche fournit des modèles qui reflètent mieux les données en dérivant la taille optimale du modèle MMC de chaque cluster. La sélection d'un MMC favorise celui qui est le plus simple et qui explique les observations en offrant une représentation précise des données pour faciliter leur interprétation.

Contribution du mémoire

Dans la littérature existante, la méthodologie MMC a été adapté au clustering et la modélisation des données temporelles en s'appuyant sur l'expertise humaine pour spécifier la structure d'un modèle. Ceci représente un handicape majeur car le processus d'analyse n'est pas automatique et nécessite l'intervention des experts.

Ce mémoire apporte une contribution à ce problème en introduisant une procédure de sélection de la taille d'un modèle MMC dans le processus de clustering afin de calculer automatiquement la structure optimale des clusters. Cette étape de traitement fournit des meilleures modèles de clustering, facilitant ainsi la tâche d'interprétation du modèle et améliorant la qualité de la structure globale du clustering généré. Cette approche permet d'utiliser la modélisation MMC dans differents domaines, surtout ceux qui ont une connaissance imparfaite de prédéfinition de la structure des clusters.

Nous proposons une fonction objectif qui fait le lien entre une variable représentant la taille optimale du modèle avec la vraisemblance de l'ensemble de données. Ainsi, l'objectif est de trouver la taille du modèle qui maximise la vraisemblance de l'ensemble des données.

En outre, nous utilisons le clustering par mélange fini d'MMCs. Dans ce cas, le modèle est composé d'un ensemble de MMCs caractérisant les différents clusters d'objets de données homogènes. Nous utilisons la fonction objectif pour sélectionner la partition de clustering optimale.

Nous présentons notre algorithme de clustering à base de MMCs comme un processus de recherche décomposé de 4 étapes clés. Des procédures de recherche heuristique basées sur la fonction objectif sont employées pour traiter les 2 premières étapes clés de l'algorithme à savoir la sélection du modèle MMC pour chaque cluster dans la partition et la détermination du nombre optimal de clusters. Nous avons également initialisé et estimé des paramètres du MMC à l'aide de l'algorithme Baum Walch. Ainsi une procédure itérative de redistribution d'objets a été appliquée afin d'assurer la convergence du modèle. Ces quatre étapes ont été intégrées dans

une structure de recherche efficace qui exploite les caractéristiques de la fonction objectif. Les résultats expérimentaux montrent que ces heuristiques sont efficaces pour trouver la structure optimale.

Organisation du mémoire

Ce mémoire est organisé comme suit : Le chapitre 1 illustre les différentes caractéristiques des données temporelles ainsi que leurs domaines d'application. Nous avons mis l'accent sur la correspondance entre une donnée temporelle et un processus stochastique. À cet effet, nous présentons les MMCs qui constituent une famille d'outils mathématiques probabilistes parfaitement adaptés à la modélisation de séquences temporelles.

Le chapitre 2 aborde en premier lieu les différentes méthodes qui permettent de traiter les séquences temporelles, notamment celles basées sur les données (avec ou sans prétraitement), puis nous mettons l'accent sur les méthodes à base de modèles probabilistes à savoir les MMCs. Nous consacrons le reste de ce chapitre à présenter les travaux antérieurs menés sur le clustering des séquences temporelles où nous relatons pour chacune des méthodes son principe, ses avantages et ses inconvénients. Cet état de l'art permettra de présenter des approches alternatives de classification pour premettre de positionner notre contribution dans le chapitre suivant.

Dans le chapitre 3, nous proposons une méthodologie de clustering pour l'analyse des séquences temporelles à base de modèles de Markov cachés (MMC). Notre approche utilise le critère de fonction objectif dans le but de déterminer le nombre optimal de clusters ainsi que la cohérence de leurs structures. L'approche proposée est composée de quatre étapes : la recherche du nombre optimal de clusters, la recherche de la structure cohérente de chaque cluster, la distribution d'objets aux clusters, et la configuration des paramètres de chaque cluster. Cette approche sera validée par des résultats expérimentaux, qui sont présenté dans la chapitre suivant.

Le chapitre 4 est consacré à une étude de cas pour l'analyse de la variabilité de comportement des diabétiques au cours du temps dans le but d'évaluer et d'améliorer la qualité des soins du diabète. En appliquant notre méthodologie de clustering à base de MMCs cette étude va montrer l'efficacité de notre algorithme à travers les résultats experimentaux obtenus. Ainsi, une étude comparative est aussi présentée pour montrer l'apport de notre approche par apport aux travaux antérieurs.

Le dernier chapitre conclut ce travail en dégageant plusieurs pistes de recherche notamment en évoquant les hypothèses de travail qui restent à reconsidérer.

Chapitre I

Les données temporelles et les MMCs

1 Introduction

Les séquences temporelles sont des suites ordonnées de symboles (de lettres, de signaux, d'états, d'événements), et sont au cœur de divers domaines, en Biologie (les séquences d'ADN), en Ingénierie (les séquences de données collectées par des capteurs, tels que le contrôle des télécommunications), en Marketing (l'étude de comportements dans le temps des clients), en Médecine (l'étude de la variabilité au cours du temps des résultats des analyses), etc.

L'analyse de séquences temporelles permet de rechercher les relations cachées entre les séquences (les tendances de plusieurs variables, relations temporelles entre occurrences de certains types d'événements, etc.). Il s'agit d'exploiter les dépendances entre les occurrences de certains types d'événements ; ce qui est difficile quand la dimension temporelle est négligée ou simplement introduite comme un attribut numérique additionnel dans la description des données. Cela met clairement en évidence la nécessité de modéliser la dimension temporelle.

Les modèles probabilistes sont considérés comme un générateur de séquences temporelles, et l'apparition d'un événement dans ces séquences est considérée comme une variable aléatoire. L'une des variantes de ces modèles probabilistes est le modèle de Markov caché (MMC) qui suscite depuis plusieurs années un vif intérêt dans la modélisation des systèmes et s'est rapidement imposé comme une référence dans plusieurs disciplines comme la reconnaissance de la parole, la classification automatique, la reconnaissance des séquences d'ADN,etc. Les MMCs doivent leur sucés à l'existence des algorithmes élégants et efficaces pour l'apprentissage, et qui reposent sur des bases mathématiques rigoureuses

Dans ce chapitre, nous présenterons tout d'abord les séquences temporelles, leurs caractéristiques et leurs domaines d'application. Nous consacrerons le reste du chapitre à présenter les MMCs, leur concepts et les techniques pour la modélisation et la classification des séquences temporelles.

Definition Une séquence temporelle est une suite de valeurs qui sont ordonnées selon leur ordre chronologique. Une séquence temporelle est donc formée de valeurs discrètes et à chacune de ces valeurs est associée une information temporelle [Weigend, 94].

2 Exemple de données temporelles

De nombreux phénomènes physiques sont représentés sous forme de séquences temporelles et elles apparaissent dans différents domaines. On peut trouver quelques bons exemples de séquences temporelles dans [Lendasse, 03].

Les séquences financières, comme les indices boursiers : le Dow Jones, le Bel20, le CAC40, etc. Pour ces séquences la période d'échantillonnage peut être d'une journée ; dans ce cas les valeurs successive sont les valeurs des indices à l'ouverture ou à la fermeture des marchés. Cette période peut être également d'une heure ou de quelques minutes ; pour ce type de séquence, la fermeture des marchés pendant la nuit provoque des discontinuités temporelles comme on peut le voir dans la Figure A.1 illustrant l'indice Dow Jones de 1990 à 2010.

Figure A.1 – l'indice de Dow Jones de 1990 à 2010.

Les séquences environnementales : Par exemple, le débit d'une rivière. Dans ce cas, la période d'échantillonnage est d'une heure ou de quelques heures. Le débit d'une rivière belge (en m^3/s) est représenté dans la Figure A.2 entre 1993 et 1996. La période d'échantillonnage est d'une heure.

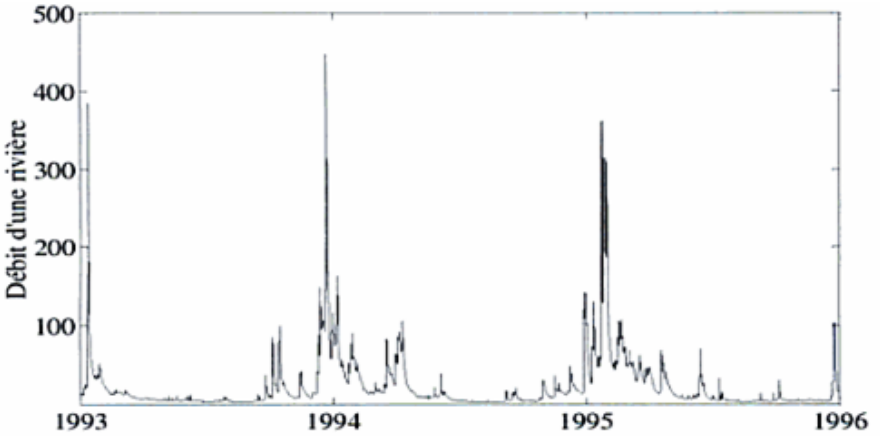

Figure A.2 – Le débit d'une rivière de Belge de 1993 à 1996

Les séquences industrielles Par exemple, la consommation électrique d'une entreprise, d'une région ou d'un pays. La période d'échantillonnage peut être dans ce cas de 15 minutes, d'une heure ou d'une journée. La Figure A.3 illustre la consommation d'électricité en Pologne entre 1987 à 1999. La période d'échantillonnage est d'une journée. Les valeurs utilisées sont les valeurs moyennes journalières.

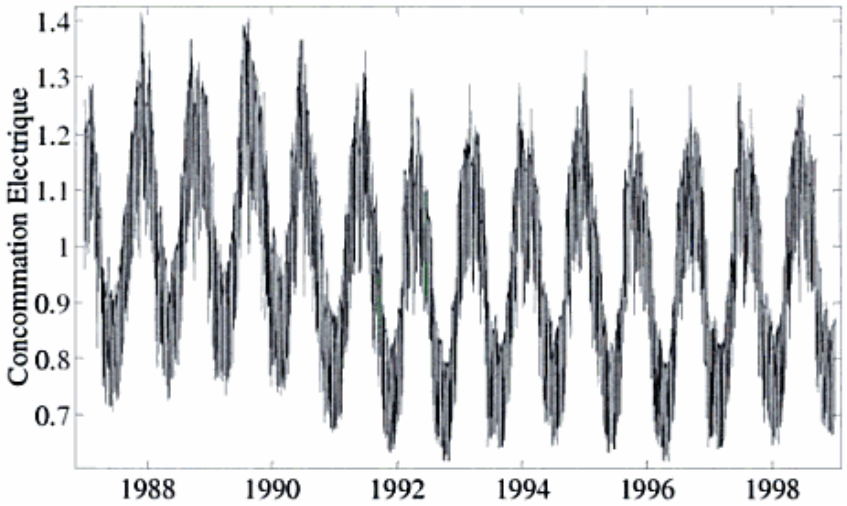

Figure A.3 – La consommation électrique normalisée de Pologne de 1987 à 1999

3 L'intérêt des séries temporelles

L'intérêt d'étudier *les séries temporelles* réside dans une large gamme de systèmes qui sont concernés par ce type de données. Pour chacun de ces systèmes, les enjeux sous-jacents sont présentés [Lendasse, 03] :

- Les séries financières comme les indices boursiers. Elles influencent la vie de tous les jours : quand on fait le plein de carburant ou de manière plus dramatique quand les grandes multinationales licencient des milliers de travailleurs afin d'augmenter leur rentabilité.
- L'étude des séries d'environnement est d'un intérêt primordial pour tout ceux qui vivent de l'agriculture ou pour tout ceux qui vivent près des cours d'eau.
- L'étude des séries de consommation d'électricité permet non seulement aux producteurs et aux consommateurs de faire des économies, mais permet aussi de sauvegarder les ressources naturelles utilisées.
- En astronomie, l'études des séries, comme le nombre de taches solaires est utile pour comprendre le fonctionnement du soleil, et permet également de mieux connaître les champs magnétiques qui en résultent et qui sont, par exemple, les causes du disfonctionnement dans les réseaux GSM.
- ETC.

L'utilité de l'étude de ces séries temporelles est donc indéniable. Il s'agit de comprendre la série et son comportement pour enfin faire une prédiction ou une classification. Pour cela, il est nécessaire de ramener la série à un environnement formel adéquat afin de bien maitriser sa dimension temporelle et ainsi comprendre son comportement.

4 Modélisation des séquences temporelles

Nous allons tout d'abord présenter une description formelle d'une séquence temporelle puis citer les différentes méthodes d'analyse de données temporelles.

4.1 Représentation d'une séquence temporelle

Une séquence temporelle est définie comme une suite ordonnée de paire (x_i, t_i) avec $x_i \in E$(E ensemble fini) la nature de l'événement et t_i un nombre réel positif représentant la date de l'occurrence de l'événement $(i = 1...n : t_i \leq t_{i+1})$. Une séquence d'événements sur E est le triplet (Seq, T_d, T_f) où [Mannila et al. ; 1997] :

$$Seq =< (x_1, t_1), (x_2, t_2), ..., (x_n, t_n) >$$

$$T_d = t_1, T_f = t_n$$

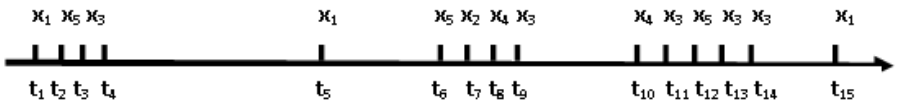

Figure A.4 – Une séquence temporelle

La figure A.4 ci-dessus représente une séquence temporelle (5 types d'événements où (x_j, t_i) i=1,5 et j=1,N représente respectivement la nature de l'événements et son instant d'apparition.

4.2 Modèle d'analyse des séquences temporelles

Les approches d'analyse de séquences temporelles sont multiples et variées, ces méthodes peuvent êtres classées en trois catégories [liao, 2005] :

- *Approche basée sur les données brutes (Raw-Data-Based-Approches (RDBA))* : ces méthodes utilisent directement les données temporelles initiales sans aucun traitement au préalable. Dans cette catégorie les méthodes sont basées sur l'utilisation de distances entre les géométries des séries temporelles considérées, comme la distance géométrique moyenne utilisée par [VanWijk 99], ou la distance DTW '*Dynamic Time Warping*l utilisée en reconnaissance de gestes [Gavrila 95] ou en reconnaissance de parole [Rabiner 93], etc.
- *Approche basée sur les caractéristiques des données (FDBA pour Feature-Data-Based-Approches* [Goutte et al 1999]) : une nouvelle approche se basant sur l'étude des caractéristiques des données, elle regroupe les caractéristiques principales des données brutes dans un vecteur de *petite dimension*. Ces réductions de dimension peuvent permettre de diminuer le temps de calcul, ainsi que la convergence de certains algorithmes de clustering (de type EM ou k-means) vers des minima locaux [Ding 02]. De plus, les données pouvant être fortement bruitées, un prétraitement peut s'avérer nécessaire et efficace pour l'analyse de séries temporelles.
- *Approche à base de modèles (Model-Based-Approch)* : cette classe d'approches considère que chaque séquence temporelle est générée par un modèle formel. Dans cette catégorie, nous trouvons principalement la famille des modèles de régression linéaire (AR, ARMA, ARIMA...) [Piccolo, 1990], les modèles probabilistes, et les réseaux de neurones. Contrairement aux FDBA, les MBA ne nécessitent pas une phase intermédiaire d'extraction de caractéristiques des données. Ils se basent directement sur l'estimation des paramètres du

modèle choisi. elles sont ainsi plus faciles et plus simples à mettre en œuvre. C'est sur ces types d'approches que porte notre réflexion dans ce projet.

a) Modèles de régression linéaire : Les modèles ARMA (Auto Regressive Moving Average), introduit par [Box et Jenkins, 1976], sont des modèles linéaires utilisés pour la prédiction des valeurs futures d'une séquence. Ils permettent de combiner deux types de processus temporels : les processus autorégressifs (*AR-AutoRegressive*), et les moyennes mobiles (*MA-Moving Average*). Pour un processus AR, chaque valeur de la série est une combinaison linéaire des valeurs précédentes. Si la valeur de la série à l'instant t, y_t , ne dépend que de la valeur précédente y_{t-1} à une perturbation aléatoire prés ϵ_t, le processus est dit autorégressif du premier ordre et noté AR(1). Dans un cadre général, un modèle AR d'ordre p est décrit par :

$$y_t = \delta + \phi_1 y_{t-1} + ... + \phi_p y_{t-p} + \epsilon_t \tag{I.1}$$

La valeur courante d'un processus de moyenne mobile (MA) est définie comme une combinaison linéaire de la perturbation courante avec une ou plusieurs perturbations précédentes. Un MA d'ordre q est alors décrite par :

$$y_t = \theta_1 \epsilon_{t-1} + ... + \theta_q \epsilon_{t-q} \tag{I.2}$$

La valeur courante d'un processus ARMA (p,q) est défini en combinant les équations précédentes

$$y_t = \delta + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{i=1}^{q} \theta_i \epsilon_{t-i} + \epsilon_t \tag{I.3}$$

Les modèles ARMA sont des outils utilisés pour l'analyse des séquences temporelles. Cependant, ils ne peuvent être efficaces que lorsque :

- Les séquences étudiées sont stationnaires.
- Les données d'apprentissage sont de très bonne qualité.
- Le modèle mathématique est correctement choisi ;ie, le choix de l'indice des ordres p et q.

b) Les réseaux de neurones : Les réseaux de neurones (NN pour Neural Network), et spécialement les réseaux de neurones récursifs [Dorffner, 1996] (RNN pour Recurrent Neural Network), sont des modèles non linéaires très utilisés pour l'analyse des séquences temporelles. Ils sont définis comme $I \times H \times O$, où I, H et O représentent respectivement, les neurones d'entrée (*input/units*), neurones cachés (*Hidden units*) et neurones de sortie (*Outputs units*). Les NN et les RNN ont été utilisés avec succés dans plusieurs domaines de prédiction [Aussem, 1995] se caractérisant par des phénomènes temporels. Toutefois, les NN admettent quelques inconvénients et notamment :

- La structure d'un NN, c'est-à-dire le nombre de couches cachées et des nœuds dans chaque couche, ainsi que les fonctions d'activation et de combinaison utilisées, a un impact très important sur la performance d'un NN. Cependant, la structure d'un NN est déterminée de façon relativement empirique puisqu'il n'existe pas de procédure systémique définissant ces paramètres.
- Les modèles ne sont pas interprétables. En effet, durant l'apprentissage d'un NN, l'ajustement des paramètres du modèle consiste à optimiser les sorties du réseau selon un objectif bien défini[Li, 2000]. Par conséquent, il est très difficile d'interpréter physiquement les neurones, et /ou les relations entres les neurones.

c) Les modèles probabilistes : Contrairement au ARMA et les NN, les modèles probabilistes, tels que les DMC (Discret Markov Chain : chaîne de Markov discrète), ou les MMC (Hidden Markov Model : Modèle de Markov cachés) l'objectif est non pas de déterminer directement la nature exacte de l'événement courant, mais plutôt sa meilleure conjecture. Il s'agit donc de déterminer le degré de confiance attribué à l'appartenance de cet événement à une classe prédéterminée. La mesure de ce degré se traduit formellement par le calcul d'une probabilité conditionnelle $P(x_{t+h} \quad x_{1:n})$.

$x_{1:n} = (x_1, x_2, ..., x_n)$.

h est l'horizon sur lequel nous voulons faire la prédiction $h > n$.

Dans l'analyse des séquences temporelles avec ARMA la prédiction joue un rôle important, alors que les modèles probabilistes c'est plutôt l'évolution qu'on essaie de modéliser. De cet effet les modèles probabilistes se sont rapidement imposés comme référence dans plusieurs disciplines comme la reconnaissance de la parole, la classification automatique, la reconnaissance des séquences d'ADN, etc. Pour cela notre choix est fixé sur ce type d'approche.

Il est d'usage d'aborder l'étude des séries temporelles en évoquant des outils de modélisation fondés, essentiellement, sur les modèles probabilistes. Parmi ces variantes les modèles de Markov cachés.

Les Modèles de Markov cachés (MMC) suscitent depuis plusieurs années un vif intérêt dans le domaine de la modélisation. À cet effet nous nous intéressons à la modélisation des séries temporelles à base des modèles de Markov cachés.

5 Modélisation avec les MMCs

D'une façon plus générale, la modélisation de séquences temporelles d'observations est un problème important en analyse de données multidimensionnelles, intelligence artificielle et en reconnaissance des formes. Il s'agit de classer des observations -on dit souvent aligner des observations sur des états qui jouent le rôle de classes - non seulement en fonction de leurs caractéristiques statiques, mais aussi en fonction de leur chronologie les unes par rapport aux autres. Ce travail de classification cherche à réduire la dimension de l'ensemble des observations, car une séquence d'observations se décrit comme une séquence d'états. Chaque état représente la variabilité du phénomène capturé par une variable aléatoire.

Les modèles de Markov cachés permettent un alignement entre un nombre quelconque d'observations temporelles et un ensemble d'états définis a priori. Ces caractéristiques sont des outils appropriés pour dégager des régularités temporelles ou spatiales comme le montrent les travaux en reconnaissance de la parole [Jelinek, 1976, J.-F. Mari, 1997] et comme cela a été démontré dans différents domaines : segmentation d'images [Benmiloud,1995], génétique [F. Mury, 1997, L. Bize 8, 1999], robotique [Aycard, 1997] et fouille de données [Berndt, 1996]. Les modèles de Markov cachés sont adaptés à la modélisation de séquences temporelles, pour pouvoir les présenter, il est nécessaire de commencer par présenter les modèles de Markov et les propriétés qui leur sont associées [Alan, 1988].

5.1 Les chaînes de Markov discrètes

En probabilités, on définit une variable aléatoire (v.a) réelle une fonction mesurable X : $\Omega \longrightarrow R$. Ω est appelé l'univers. Dans de nombreux cas de figure, Ω est l'ensemble des réels R ou l'ensemble des entiers positifs N ou un de leurs sous-ensembles. **Définition 1** : *Un Processus stochastique est une famille $X_{t,t\in T}$ de variable alétoire définit dans :*

$$\Omega(X_t : \Omega \longrightarrow R).$$

. L'ensemble T représente souvent la notion de temps, mais il peut également correspondre à la notion de position spatiale en dimension 2 ou a toute autre notion avec autant de dimensions

que nécessaire. Dans le cas où T représente la notion de temps et si T est discret, on parle de processus stochastique en temps discret, tandis que le processus est dit en temps continu, lorsque T est continu.

Définition 2 *les états d'un processus stochastique défini par les variables alétoires* $X_t : \Omega \longrightarrow R$ *pour tout* $t \in T$ *sont des valeurs prises par ces variables aléatoires lorsque t varie. On note S l'ensemble des "états" du processus.*

Andreï Markov[1] fut le premier à étudier et à poser les bases mathématiques permettant l'étude des chaînes qui portent son nom. La définition de ces chaînes est la suivante :

Définition 3 : Un $S_{t,t\in T}(S_t : \Omega \longrightarrow S)$ est une chaîne de Markov s'il vérifie les trois conditions suivantes :

1. *Test dénombrable ou fini. Dans ce cas et pour simplifier les notations ultérieures, il est toujours possible de prendre* $T \subseteq N^* = 1, 2, ...$ *cette condition signifie que le processus ne change de valeurs qu'à des instants déterminés à priori.*
2. *L'ensemble S des états du processus est dénombrable. Dans la suite, nous supposons également que S est fini. Nous pouvons alors définir* $S = \{s_1, s_2, ..., s_N\}$.
3. *Le processus est associé à une fonction de probabilité* $\boldsymbol{P}$ *vérifiant la propriété markovienne : " la probabilité que le processus soit dans un état particulier à un instant t ne dépend que de l'état dans lequel se trouve le processus au temps t-1 ". soit* $Q = (q_t), t \in T$ *une suite d'états du processus* $(q_t \in S)$. *La propriété de Markov vérifie la relation suivante, pour toute suites d'état Q et pour tout instant* $t \in T$ *:*

$$P(S_t = q_t/S_{t-1} = q_{t-1}, ..., S_1 = q_1) = P(S_t = q_t/S_{t-1} = q_{t-1})$$

La probabilité $P(S_t = q_t/S_{t-1} = q_{t-1})$ correspond à la probabilité de transition de l'état q_{t-1} à l'instant t-1 vers l'état q_t à l'instant t.

5.2 MMC

Un modèle de Markov caché discret correspond à la modélisation de deux processus stochastiques : un processus caché parfaitement modélisé par une chaîne discrète et un processus observable dépendant des états du processus caché.

1. Né en 1856 à Riazan, il étudia à l'Université d'État de Saint-Pétersbourg en 1874 sous la tutelle de Tchebychev et en 1886, il devint membre de l'Académie des Sciences de Saint-Pétersbourg. Ses travaux sur la théorie des probabilités l'ont amené à mettre au point les chaînes de Markov qui l'ont rendu célèbre. Ceux-ci peuvent représenter les prémices de la théorie du calcul stochastique.

Les MMCs permettent de modéliser par différents états chaque sous-partie statistiquement stable de la séquence d'observations. C'est-à-dire que toutes les observations modélisées par chaque état sont représentées par des vecteurs de primitives se regroupant dans une certaine région de l'espace des observations. Les transitions entre ces parties stables correspondent aux transitions entre les états de la chaîne de Markov cachée.

5.3 Les paramètres d'un MMC

Soit S=$\{s_1..., s_N\}$ l'ensemble des N états cachés du système, soit V=$\{v_1..., v_M\}$ l'ensemble des M symboles émissibles par le système.

Un modèle de Markov caché est alors défini par les paramètres suivants :

- **N, le nombre d'états** *du modèle* Nous désignons les états individuels par :
$$\text{S}=\{s_1..., s_N\}$$
et l'état au temps t par $q_t, t \in S$.
- **M, le nombre des symboles d'observation distincts**. Nous désignons ainsi l'ensemble des symboles d'observation par :
$$O_t = v_k, vk \in \text{V}=\{v_1, v_2, ..., v_M\}$$
- **A, les probabilités de transition** entre états cachés :
$$A = a_{ij}$$
$$a_{ij} = P(q_t = s_j / q_{t-1} = s_i), 1 \leq i, j \leq N$$
$$\sum_{j=1}^{N} a_{ij} = 1, 1 \leq i, j \leq N$$
- **B, les probabilités d'émission** des symboles dans chaque état caché :
$$B = b_j(O_t), j = 1, 2, ..., N$$
$$b_j(O_t = v_k) = P(O_t = v_k / q_t = s_j), 1 \leq j \leq N, 1 \leq k \leq M$$
$$\textstyle\sum_{k=1}^{M} b_j(O_t = v_k) = 1, 1 \leq j \leq N$$
- Π**, les probabilités d'initialisation** d'états cachés :
$$\Pi = \pi_i$$
$$\pi_i = P(q_1 = s_i), 1 \leq i \leq N$$
$$\sum_{j=1}^{N} \pi_i = 1, 1 \leq i \leq N$$

On peut conclure que la spécification complète d'un MMC nécessite :

- Deux paramètres (N et M pour un MMC discret) ;
- La définition des vecteurs d'observations
- Les distributions des probabilités A, B, Π.

Nous le désignons par :

$$\lambda = (A, B, \Pi)$$

Pour faciliter la compréhension de ces différentes notions, considérons le MMC de la figure A.5. Ce MMC montre un modèle de prévision météorologique qui possède N= 3 états (*Beau, Mauvais ou Variable*). Le vecteur de densités de probabilités initiales, Π=(0.5,0.2,0.3), indique que le MMC peut être initialement dans l'état "Beau" avec une probabilité 0.5, dans l'état "Mauvais" avec une probabilité 0.2 et dans l'état "Variable" avec une probabilité 0.3 ; $\sum_{i=1}^{R} \pi_i = 1$. À tout instant t, un des M symboles dans V= {Ensoleillé, Brumeux, Pluvieux} peut être choisi puis émis en analysant la matrice B de probabilités d'observation ; $B = b_j(s)|1 \leq i \leq N, 1 \leq s \leq M$.

A titre d'illustration, le modèle possède une forte probabilité 0.7 d'émettre le symbole "Ensoleillé" quand il se trouve dans l'état "Beau".

Une fois un symbole de sortie choisi et émis par le modèle, le MMC passe à un nouvel état, sélectionné à partir de la matrice de probabilités de transition entre les états du modèle $A = a_{(i,j)}|1 \leq i, j \leq N$. Par exemple, la probabilité de passage de l'état "Beau" à l'état "Mauvais" $a_{Beau,Mauvais} = 0.2$, alors que la probabilité de rester dans l'état "Beau" est $a_{Beau,Beau}$= 0.5, sachant que la somme des probabilités de transitions pour chaque doit être égale à $\sum_{i=1}^{R} a_{ij} = 1$.

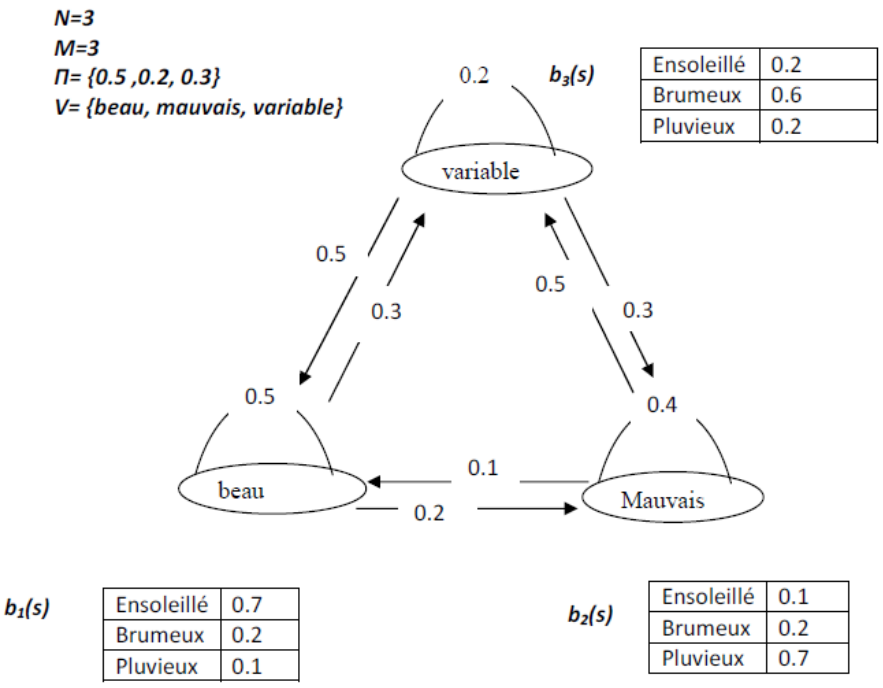

Figure A.5 – un exemple de modèle de Markov caché

5.4 Les algorithmes associés au MMC

Rabiner and Juang, 1986 identifie trois différents types de problèmes pour rendre ce formalisme des MMCs utile :

1. Problème 1 : **Evaluation du modèle**
   Étant donnée une séquence d'observations O et un modèle λ= (Π, A, B), comment calculer la vraisemblance de la séquence, c'est-à-dire la probabilité que ce modèle génère cette séquence ? Ce problème est résolu par la programmation dynamique avec l'algorithme Forward ou par son approximation Viterbi.
2. Problème 2 : **Estimation de la suite d'états cachés**
   Étant donnée une séquence d'observations O et un modèle λ= (Π, A, B), comment trouver la meilleure séquence d'états au sens de la probabilité de vraisemblance de la séquence ? L'algorithme "retour arrière" (backtracking) après décodage par l'algorithme de Viterbi permet de trouver cette séquence d'états.
3. Problème 3 : **Apprentissage**
   Comment ajuster les paramètres d'un modèle pour maximiser la probabilité d'observation des exemples ? L'apprentissage se fait selon l'algorithme de Baum-Welch, qui fait appel au principe de l'algorithme de EM, (Expectation Maximization [Dempster et al., 1977]). Ces algorithmes de programmation dynamique rendent les MMCs efficaces.
   Nous allons à présent décrire ces algorithmes qui permettent de résoudre les 3 problèmes cité ci-dessus.

5.4.1 L'algorithme Forward

L'algorithme Forward résout le problème d'évaluation de la vraisemblance d'une séquence d'observations. Pour présenter cet algorithme, il est nécessaire de définir les variables Forward pour tout (i=1,...,N et t=2,...,T) : cette variable $\alpha_t(i)$ représente la probabilité d'émettre la suite $O_1O_2, ..., O_t$ et d'aboutir à l'état q_i à l'instant t sachant le modèle λ.

$$\alpha_1(i) = P(V = O_1, S_1 = s_i/\lambda)$$
$$\alpha_t(i) = P(V_1 = O_1, ..., V_t = O_t, S_t = s_i/\lambda)$$

On remarque alors que la relation de récurrence suivante est vérifiée pour tout t=1,...,T-1 et j=1,...,N :

$$\alpha_{t+1}(j) = b_j(o_{t+1}) \sum_{i=1}^{N} \alpha_t(i) a_{ij}$$

De plus, on a $P(V = O/\lambda) = \sum_{i=1}^{N} \alpha_t(i)a_{ij}$, l'algorithme Forward est alors donné par l'algorithme 1. La complexité de cet algorithme est $O(N^2T)$.

Algorithm 1 l'algorithme Forward

1: **Pour** i=1 à N **faire**
2: $\alpha_1(i) = \pi_i b_i(o_1)$
3: **Fin pour**
4: **Pour** t=1 à T-1 **faire**
5: **Pour** j=1 à N **faire**
6: $\alpha_{t+1}(j) = \sum_{i=1}^{N}(\alpha_t(i)a_{ij})b_j(o_{t+1})$
7: **Fin pour**
8: **Fin pour**
9: $P(V = O/\lambda) = \sum_{i=1}^{N}(\alpha_T(i))$

5.4.2 L'algorithme de Viterbi

L'algorithme de Viterbi [Viterbi , 1967] est une approximation de la fonction Forward, qui calcule la probabilité du meilleur chemin à la place de la somme sur tous les chemins. Il permet de répondre à la deuxième question de Rabiner. L'optimisation globale de la recherche du meilleur chemin est décomposée en une succession d'optimisations locales, selon le "principe de Maupertuis" (rappelé dans [Lecolinet, 1990]), qui remarque que tous les sous-chemins du chemin optimal sont également optimaux localement, principe qui est utilisé dans tous les algorithmes de programmation dynamique. À chaque instant t, la probabilité $\delta_t(i)$ du meilleur chemin aboutissant à l'état i est optimisé en fonction des probabilités à l'instant précédentes et des probabilités de transition $a_{i,j}$. Un pointeur $\phi_t(i)$ est gardé sur le meilleur candidat de l'optimisation :

$$\delta_1(i) = \pi_i b_i(o_1)$$
$$\forall t \delta_t(i) = (max_j \delta_{t-1} a_{j,i}) b_i(o_t)$$
$$\psi_t(i) = argmax_j \delta_{t-1}(j) a_{j,i}$$

Dans un deuxième temps, la recherche arrière du meilleur chemin (backtracking) permet de segmenter le signal en régions stationnaires, c'est-à-dire d'aligner les observations o_t sur les états q_t du MMC :

$$q_t = \psi_{t+1}(q_{t+1})$$

L'algorithme de Viterbi est donné dans Algorithme 2. Sa complexité est $O(N^2T)$.

Algorithm 2 L'algorithme de Viterbi

1: **Pour** i=1 à N **Faire**
2: $delta_1(i) = \pi_i b_i(o_1)$
3: **Fin pour**
4: **Pour** t =2 à N **Faire**
5: **Pour** j =1 à N **Faire**
6: $\psi_t(j) = argmax\left\{\delta_{t-1}(i)a_{ij}\right\}$
7: $\delta_t(j) = max_{1\leq i\leq N}\left\{\delta_{t-1}(i)a_{ij}\right\} b_j(o_t) = \delta_{t-1}(\psi_t(j))a_{\psi_t(j),j}b_j(o_t)$
8: **Fin Pour**
9: **Fin Pour**
10: $q_T^* = argmax_{1\leq i\leq N}\left\{\delta_T(i)\right\}$
11: $P(V = O, S = Q^*) = max_{1\leq i\leq N}\left\{\delta_T(i)\right\} = \delta_T q_T^*$
12: **Pour** t=T-1 à 1 **Faire**
13: $q_t^* = \psi_{t+1}(q_{t+1}^*)$
14: **Fin Pour**

5.5 Apprentissage des MMCs

l'apprentissage d'un MMC consiste à ajuster les paramètres du modèle de manière à maximiser un certain critère. Différents critères sont décrits dans la littérature. Nous n'allons pas tous les recenser, mais nous allons présenter les plus importants et les plus couramment utilisés.

5.5.1 Maximisation de la vraisemblance

Le critère de *maximum de vraisemblance* consiste à trouver le modèle λ^* maximisant la probabilité $P(V = O/\lambda)$ [Rabiner, 1989]. En général, il n'est pas possible de trouver le modèle optimal. Néanmoins, pour tenter de résoudre ce problème, il existe principalement deux méthodes : utiliser l'algorithme Expectation-Maximization, ou descente de gradient.

a) Expectation-Maximization : L'algorithme *Expectation-Maximization* (EM) est une méthode générale d'optimisation en présence d'information incomplète [Dempster et al., 1977]. L'algorithme permet, à partir d'un modèle initial ***m'***, de trouver un modèle ***m*** qui augmente la vraisemblance. Cette algorithme repose sur deux hypothèses simples :

- Maximiser *P(X=x/M=m)* est équivalent à maximiser *ln P(X=x/M=m)*; tel que X est l'ensemble de séquences de données observées et M le modéle probabiliste.
- L'introduction de l'ensemble de variables non observés ou cachés Y dans l'expression de la vraisemblance permet d'effectuer les calculs plus facilement. Dans le cas de variables aléatoire discrète, on définit $\Gamma_x(m, m')$ la log-vraisemblance complétée (Amini, 2001)

comme la quantité :

$$\begin{aligned}\Gamma_x(m,m') &= \sum_{y\in\Upsilon} P(Y=y/X=x,M=m')lnP(X=x,Y=y/M=m)\\ &= E_{y\in Y}\left[lnP(X=x,Y=y/M=m)/X=x,M=m'\right]\end{aligned}$$

où $E_Y[f]$: est l'espérance mathématique de f sur l'ensemble Y

L'algorithme EM (Dempster et al., 1977) (Moon, 1966) consiste donc à construire, à partir d'un modèle initial m_1,une suite de modèle $(m_t)_{t\geq 0}$ vérifiant :

$$\Gamma_x(m_{t+1},m_t) \geq \Gamma_x(m_t,m_t)$$

Une condition suffisante est alors de rechercher le modèle m_{t+1} qui maximise la fonction $\Gamma_x(m_{t+1},m_t)$. La suite vérifie, pour tout t > 1 et $m_{t+1} \neq m_t$, relation

$$P(X=x/M=m_{t+1}) > p(X=x/M=m_t)$$

L'une des plus célèbres applications de l'algorithme EM est l'algorithme Baum-Welch qui est utilisé pour l'apprentissage des MMC (Bilmes, 1998).

• **Algorithme de Baum-Welch :** Dans le cas des MMCs, on cherche à maximiser $P(V = O/\lambda)$ où O désigne une séquence de T observations. En appliquant l'algorithme EM pour maximiser cette probabilité (Bilmes, 1998), on est amené à maximiser $\Gamma(\lambda,\lambda')$, avec $\lambda = (A,B,\Pi)$ le nouveau modèle et λ' le modèle connu (ou actuel) :

$$\Gamma_o(\lambda,\lambda') = \sum_{Q\in S^T} P(S=Q/V=O,\lambda')lnP(V=O,S=Q/\lambda)$$

En effectuant les différents calculs, on obtient :

$$\begin{aligned}\pi_i &= P(S_1=s_i/O,\lambda')\\ a_{ij} &= \frac{\sum_{t=1}^{T-1} P(S_t=s_i,S_{t+1}=s_j/V=O,\lambda')}{\sum_{t=1}^{T-1} P(S_t=s_i/V=O,\lambda')}\\ b_i(j) &= \frac{\sum_{t=1}^{T} P(S_t=s_i/V=O,\lambda')\delta(o_t=j)}{\sum_{t=1}^{T-1} P(S_t=s_i/V=O,\lambda')}\end{aligned}$$

Les formules de ré-estimation obtenues ci-dessus peuvent s'interpréter de la façon suivante :

$$\pi_i = \textit{la probabilité d'être dans l'état } s_i \textit{ à l'instant t=1.}$$

$$\frac{a_{ij}\text{= nombre de transitions de l'état } s_i \text{ vers l'état } s_j}{\text{nombre de fois où l'on quite l'état } s_i}$$

$$\frac{b_i(k)\text{= nombre d'apparitions simultanées de l'état } s_j \text{ et du symbole } v_k}{\text{nombre d'apparitions de l'état } s_j}$$

L'algorithme de Baum-Welch (Baum and Eagon, 1967) est donné ci-dessous. Sa complexité est $O(N^2T + NMT)$.

Algorithm 3 L'algorithme de Baum-Welch

1: Choisir un modèle initial λ_0
2: t=0
3: **Répéter**
4: t=t+1
5: Calculer les variables *Forward* et *Backward* pour le modèle λ_{t-1}
6: Calculer Π de λ_t
7: Calculer **A** de λ_t
8: Calculer **B** de λ_t
9: **Tant que** $(P(V = O/\lambda_t) > P(V = O/\lambda_{t-1}))$**et**$(t < t_{max})$
10:

5.5.2 Maximisation de l'information mutuelle

L'un des buts principaux de l'apprentissage de MMC est d'effectuer une classification. En effet, on cherche, à partir d'une observation O, à décider de manière automatique à quelle autre observation elle ressemble le plus et surtout à décider à quelle classe de séquence d'observations appartient elle réellement. Prenons un exemple, on considère un système d'identification biométrique basé sur une photographie d'un visage. Initialement, le système possède au moins une photographie de chacune des personnes à reconnaître, chaque photographie est modélisée par un MMC après qu'elle ait été transformée par un procédé quelconque en séquence d'observations. Si une personne se présente devant la caméra, le système va prendre une photographie, la transformer en séquence et comparer les différentes Vraisemblances avec les MMCs appris. Le

MMC qui permet d'obtenir la meilleure vraisemblance permet alors de dire que la personne est celle qui correspond à la photographie du MMC. En théorie, cette méthode fonctionne mais, en pratique, ce n'est pas toujours le cas. Si l'ensemble des photographies concerne des photographies de visages de personnes de même couleur de peau et de même couleur de cheveux, alors il ya de grandes chances pour que les modèles reconnaissent bien l'ensemble des visages, car la modélisation des visages sera quasi identique. Une solution consiste à effectuer l'apprentissage des MMCs avec un autre critère que la vraisemblance. Le critère de prédilection pour cette tâche est la maximisation de l'information mutuelle (MIM).

5.5.3 Le critère de segmental k-means

Parmi l'ensemble des critères utilisés pour l'apprentissage de MMC, le critère de *segmental k-means* se détache des autres. En effet, pour ce critère, on cherche à optimiser la probabilité $P(V = O, S = Q^*/\lambda)$ avec Q^* la séquence d'états cachés la plus probablement engendré par l'algorithme de Viterbi (cf.section 1.5.2)(Juang and Rabiner, 1990) (Dugal and Desai,1996). Cet algorithme, appelé segmental k-means repose sur deux algorithmes, décrits précédemment : l'algorithme de Viterbi et l'apprentissage étiqueté.
Son principe est simple :

- À partir d'un modèle initial et de la séquence d'observation O, on cherche la séquence d'états cachés qui a le plus probablement été suivie pour générer $\boldsymbol{O}$ à l'aide de l'algorithme de Viterbi. Cette recherche permet d'étiquter la séquence O et par conséquent de la segmenter ;
- Une fois étiquètée, la séquence $\boldsymbol{O}$ est apprise par comptage des transitions effectives entre les états et les émissions de symboles. Cette étape peut alors être considérée comme un *k-means* consistant à ré-estimer les "les centres des classes" ;
- Le nouveau modèle est alors utilisé comme modèle initial et deux opérations précédentes sont repétées tant que nécessaire.

(Juang and Rabiner,1990) ont montré que l'algorithme de *segmental k-means* (cf, algorithme 4) permet d'augmenter la probabilité $P(V = O, S = Q^*/\lambda)$ de manière itérative et qu'il converge vers un maximum local du critère considéré.

Algorithm 4 l'algorithme de segmental k-means

1: Choisir un MMC initial λ_0
2: t=0
3: **Répéter**
4: t=t+1
5: $Q^* = Viterbi(O, \lambda_{t-1})$
6: Estimer λ_t à partir de O et Q^*
7: **Tant que** $P(V = O, S = Q_t^*/\lambda_t) > P(V = O, S = Q_{t-1}^*/\lambda_{t-1})$

L'algorithme de *segmental k-means* peut également être utilisé avec plusieurs séquences d'observation. Pour cela, il suffit de considérer le critère d'apprentissage suivant généraliser pour K séquences de données :

$$\prod_{k=1}^{K} P(V = O^k, S = Q^{k*}/\lambda)$$

Cet algorithme est parfois utilisé en raison de sa rapidité de l'algorithme de Baum-welch, en considérant l'hypothèse suivante : les probabilités complétées $P(V = O, S = Q/\lambda)$ sont nulles ou négligeables pour toutes les séquences d'états, à l'exception de celle de la séquence Q^* de Viterbi associée. Par conséquent, maximiser $P(V = O/\lambda)$ est équivalent à maximiser $P(V = O, S = Q^*/\lambda)$.

5.5.4 Remarques sur les critères d'apprentissage

Comme nous venons de le voir, de nombreux critères peuvent être considérés pour l'apprentissage des MMCs. Les critères que nous avons décrits dans ce chapitre ne sont pas les seuls envisageables, mais ce sont la majorité des outils nécessaires à la conception des algorithmes d'apprentissage.

Tous les algorithmes d'apprentissage de ce chapitre n'ont pas la même complexité. Pour faciliter le choix à la fois du critère et de l'algorithme de résolution, [Aupetit,05] a construit le tableau A.1.

Critére	Algorithme	Complexité d'une itération
$P(V=o,S=Q/\lambda)$	apprentissage étiqueté	N^2+NM+T
$\prod_{k=1}^{K}P(V=O_k,S=Q_k/\lambda)$	apprentissage étiqueté	$N^2+NM+\sum_{k=1}^{K}T_k$
$P(V=O/\lambda)$	Baum Welch	N^2T+NMT
$\prod_{k=1}^{K}P(V=O_k/\lambda)$	Baum-Welch	$N^2\sum_{k=1}^{K}T_k+NM\sum_{k=1}^{K}T_k$
$P(V=O/\lambda)$	Gradient	N^2T+NMT
$\prod_{k=1}^{K}P(V=O_k/\lambda)$	Gradient	$N^2\sum_{k=1}^{K}T_k+NM\sum_{k=1}^{K}T_k$
$\frac{P(V=O/\lambda)}{\prod_{k=1}^{K}P(V=O_k/\lambda)}$	Information mutuelle	$N^2(T+\sum_{k=1}^{K}T_k)+NM(T+\sum_{k=1}^{K}T_k)$
$P(V=O/\lambda)$	simple MAP	N^2T+NMT
$P(V=O/\lambda)$	MAP complexe	Ptentiellement trés élevée
$P(V=O,S=Q^*/\lambda)$	Segmental k-means	N^2T+NM
$\prod_{k=1}^{K}P(V=O_k,S=Q_k^*/\lambda)$	Segmental k-means	$N^2\sum_{k=1}^{K}T_k+NM$

Tableau A.1 – Complexité associée aux algorithmes en fonction des critères optimisés. $O_1,...O_K$ est l'ensemble des séquences d'observation de longueur $T_1,...,T_k$, N est le nombre d'états caché du MMC.M est le nombre des symboles du MMC

Il est intéressant de remarquer que l'algorithme de segmental k-means peut être beaucoup plus rapide que l'algorithme de Baum-Welch. En effet, il est très courant que le nombre de symboles M soit beaucoup plus grand que le nombre d'états cachés. Dans ces conditions, lorsque la longueur T de la séquence d'observation augmente, le terme dominant dans la complexité de l'algorithme de Baum-Welch est NMT tandis que pour l'algorithme de segmental k-means, ce terme dominant est N^2T. par conséquent, pour M>N, l'algorithme de segmental k-means est plus rapide que l'algorithme de Baum-Welch lorsque la longueur de la séquence d'observation augmente.

6 Concepts d'analyses à base des MMCs

Cette partie va introduire le principe MMC pour la classification des séquences temporelles, à savoir la génération de séquences temporelles par un MMC, et les mesures de similarité généralement utilisées pour la classification.

6.1 MMC pour la génération de séquence

Un MMC peut être vu donc comme un processus permettant de générer une séquence d'observations $O_1 \to O_2 \to O_3 \to ... \to O_T$ comme suit :

1. choix de l'état de départ : t=1 , choisir l'état initial $q_1 = s_i$ avec π_i
2. choix d'observation dans l'état sélectionné : choisir l'observation $O_t = v_s$ avec $b_i(s)$.
3. sélection de l'état suivant : passer à l'état suivant $q_{t+1} = s_j$ à l'aide de $a_{i,j}$.
4. changement d'état : $t=t+1$; si $t<T$, alors retourner à l'étape 2 sinon STOP.

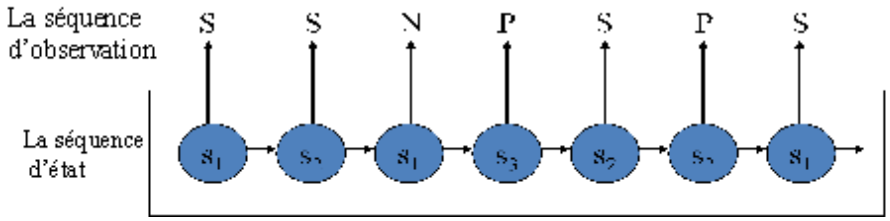

Figure A.6 – génération d'une séquence temporelle par un MMC

6.2 Mesures de similarité

Une question clé dans la classification est les mesures qui peuvent être utilisées pour déterminer la similitude ou de dissemblance entre les séquences, ou entre une séquence et un modèle MMC, ou entre deux modèles MMC.Les deux calculs de similarité utiles concernant MMC sont : (i) le calcul de la probabilité d'une séquence donnée à un MMC, et (ii) le calcul de la similarité entre deux MMCs.

6.2.1 Distance séquence-modèle

En utilisant le composant forward de la procédure forward-Backward. Ce dernier prend en compte toutes les séquences d'états possibles pour la séquence d'observation, la probabilité de la séquence est obtenue en additionnant les valeurs de $\alpha_L(i)$ pour les états :

$$L(O/\lambda) = logP(O/\lambda) = \sum_{i=1}^{M} \alpha_L(i).$$

6.2.2 Distance de similitude entre deux MMCs

Étant donné deux MMCs $\lambda_1 et \lambda_2$, la similitude entre ces deux MMCs,$D_2(\lambda_1; \lambda_2)$, est calculée en fonction de la somme normalisée de probabilités des séquences de données générées à partir des deux modèles [Rabiner,1989] :

$$D(\lambda_1;\lambda_2) = L(S_2/\lambda_1) - L(S_2/\lambda_2)$$
$$D(\lambda_2;\lambda_1) = L(S_1/\lambda_2) - L(S_1/\lambda_1)$$
$$D_s(\lambda_1;\lambda_2) = \frac{D(\lambda_1;\lambda_2)+D(\lambda_2;\lambda_1)}{2}$$

Où S_1 et S_2 représentent l'ensemble des séquences générées par λ_1 et λ_2, respectivement. Deux MMCs sont semblables les uns aux autres, lorsque les séquences observables générées par les deux modèles sont similaires, c'est-à-dire que les données de séquence générées à partir de l'un des deux modèles ont une probabilité élevée, compte tenu de l'autre modèle.

6.3 Classification des séquences à base des MMCs

Soient une population D, S_i une séquence associe à un individu $i \in D$ et soit K composants de modèle Markov caché λ=$\{\lambda_1, \lambda_2, ..., \lambda_k\}$ de paramètres (A_i, B_i, λ_i) pour chaque $\lambda_{i,1} \leq i \leq K.Z_i = \{z_{i,\lambda_1}, z_{i,\lambda_2}, ..., z_{i,\lambda_k}\}$ z_{i,λ_k} : est la probabilité que le i^{ieme} individu soit généré par le MMC λ_k

Le principe de la classification à base des MMCs est que pour chaque séquence $S_i \in D$ on

calcule le vecteur d'adhésion $Z_i = \{z_{i,\lambda_1}, z_{i,\lambda_2}, ..., z_{i,\lambda_k}\}$, où z_{i,λ_k} : la probabilité que le i^{ieme} individu soit généré par le MMC λ_i puis effectuer une distribution des séquences à base des vecteurs d'adhésion (une séquence est affecté au groupe qui maximise sa probabilité de génération $max_{1\leq j\leq k} P(S_i/z_{i,\lambda(j)})$). Enfin il faut ré-estimer les paramètres de chaque composant et refaire la distribution pour un nombre fixe d'itération.

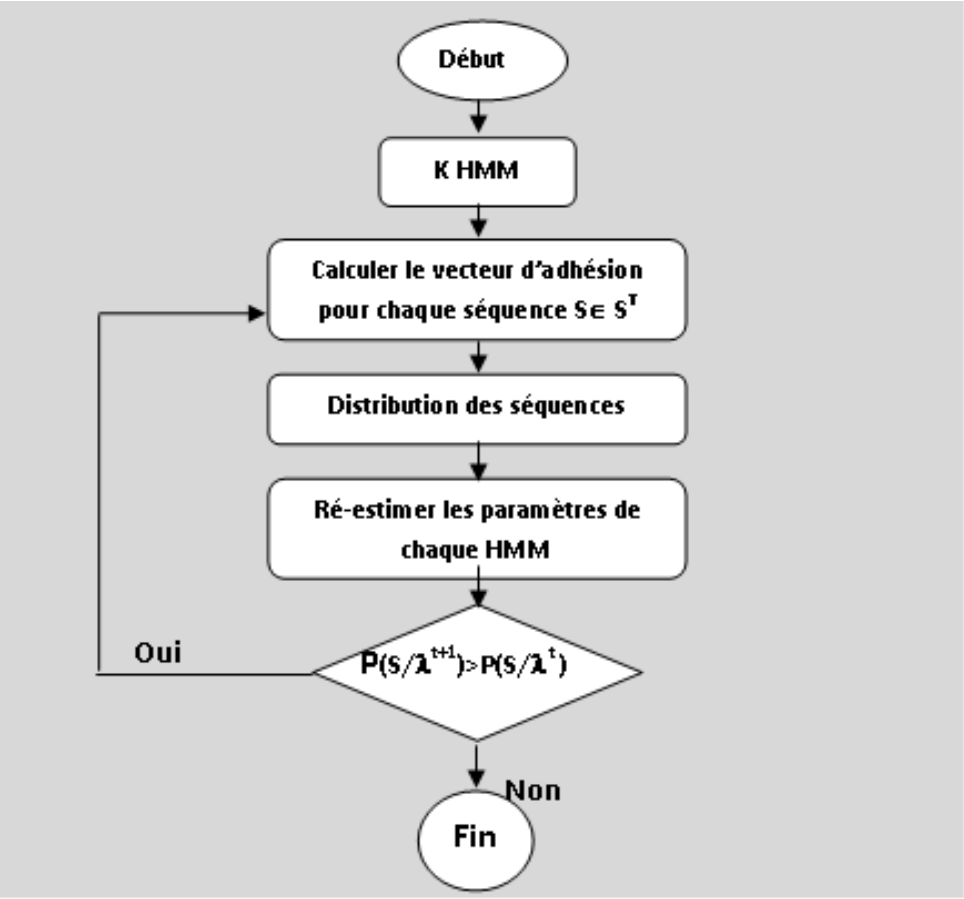

Figure A.7 – L'organigramme pour la classification par MMC

7 Conclusion

Durant ce chapitre nous avons présenté les différentes caractéristiques des données temporelles ainsi que leurs domaines d'applications. Nous avons mis le point sur la correspondance entre une donnée temporelle et un processus stochastique, une séquence d'observations se décrit comme une séquence d'états. Chaque état représente la variabilité du phénomène modélisé par une variable aléatoire. À cet effet, nous avons présenté le principe et technique des MMCs, une famille d'outils mathématiques probabilistes pour la modélisation de séquences temporelles. Ainsi, nous avons introduit les différents concepts utilisés pour la classification des séquences temporelle à base de MMCs, à savoir les mesures de similarité et le principe de classification avec les MMCs.

Le prochain chapitre est consacré à un état de l'art sur la classification des séquences temporelles à base des MMCs.

Chapitre II

État de l'art sur l'analyse des données temporelles à base d'un modèle de Markov caché

1 Introduction

Les MMCs constituent une des approches les mieux adaptées aux traitements des séquences temporelles, en raison de leur capacité à modéliser la dynamique d'un phénomène décrit par des séquences d'événements. Ces modèles se sont rapidement imposés comme référence dans plusieurs disciplines référence dans plusieurs disciplines. Dans ce chapitre nous nous intéressons à la classification automatique par MMC.

Ce chapitre aborde en premier lieu les différentes méthodes de traitement des séquences temporelles (avec et sans prétraitement), puis nous mettons l'accent sur les méthodes à bases de modèles probabilistes à savoir les modèles de Markov cachés. Nous consacrons le reste de ce chapitre aux travaux antérieurs menés sur le clustering des séquences temporelles et nous relatons pour chacune des méthodes *son principe, ses avantages et ses inconvénients.*

2 Analyse de séquences temporelles

les méthodes permettant de traiter des séquences temporelles sont de deux catégories. La première catégorie englobe les méthodes de traitement de séquences temporelles basées sur les données (avec et sans prétraitement des données), alors que la seconde catégorie est dédiée aux modéles probabilistes des séquences temporelles à l'aide des MMCs les modèles de Markov cachés. Ces approches seront détaillées dans la suite et notament le clustering de séquences temporelles [Liao 05] :

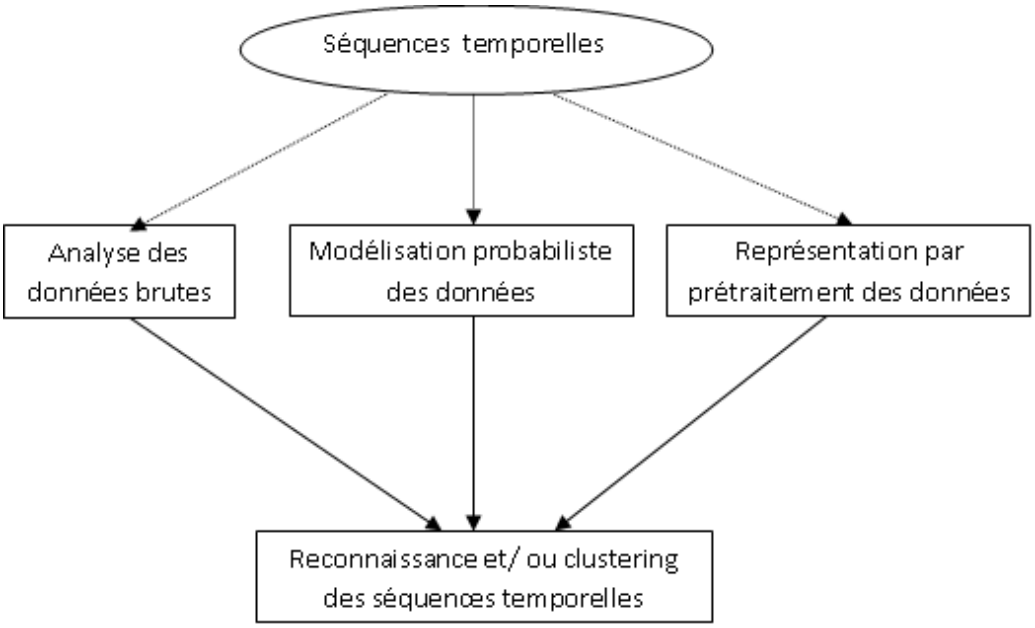

Figure A.1 – Présentation générale des méthodes de traitement des séquences temporelles.

3 Approches déterministes d'analyse des séquences temporelles

Cette partie s'intéresse aux approches qui traitent directement les séries de données pour le clustering de séquences temporelles basées sur l'utilisation de distances entre les géométries des séquences. Le but n'est pas de donner une liste exhaustive des travaux réalisés jusqu'à présent ; ce qui pourrait s'avérer fastidieux et n'aurait pas un intérêt décisif. Dans ce manuscrit, l'objectif est plutôt de décrire de façon logique les principales voies existantes et les travaux qui nous paraissent les plus intéressants.

Nous avons identifié deux types d'approches regroupant les méthodes développées pour l'analyse déterministe de séquences temporelles à savoir les méthodes de traitement de séquences temporelles sur les données brutes et les méthodes d'analyse de séquence temporelle par un prétraitement des données brutes.

3.1 Approches d'analyse des données brutes

Pour des raisons de simplicité de présentation les distances entre séquences temporelles sont décrites pour des données temporelles de dimension 1. Elles s'étendent toutes trivialement à des données temporelles de dimension supérieure.

∗***Distances de Minkowski*** Ces approches s'appuient essentiellement sur l'utilisation de distances entre séquences temporelles. La première approche, la plus simple, est de considérer la distance euclidienne entre séquences temporelles. Soient deux séquences temporelles V=$v_1, v_2, ..., v_N$ et W=$w_1, w_2, ..., w_N$ de même taille N. La distance euclidienne entre v et w est définie par :

$$d_{Eucl}(v, w) = \sqrt{\sum_{k=1}^{N}(v_i - w_i)^2}.$$

La distance géométrique moyenne, très proche de la distance euclidienne, est définie par :

$$d_{gm}(v, w) = \frac{d_{Eucl}(v,w)}{N}$$

La distance euclidienne étant elle-même un cas particulier de la distance de Minkowski définie par :

$$d_{Mink}(v, w) =^p \sqrt{\sum_{k=1}^{N} |v_i - w_i|^p}$$

Il est à noter que ces distances ne peuvent comparer que des séquences temporelles de même taille, ce qui s'avère très restrictif. Parmi les travaux utilisant la distance géométrique moyenne pour le traitement de séquences temporelles et plus particulièrement des tâches de clustering, on peut citer [VanWijk 99].

∗***Distance "Dynamic Time Warping" (DTW)*** La distance dite Dynamic Time Warping, proposée initialement par Myers et al. [Myers 81], est largement exploitée pour la comparaison de séquences temporelles. Soit deux séquences temporelles $v = v_1, ..., v_N$ et $w = w_1, ..., w_M$ de tailles N et M. On définit la fonction H(v) par H(v) = v_0 où $v_0 = v_1, ..., v_{N-1}$. La distance DTW entre v et w est définie de façon récursive par [Berndt 94] :

$$DTW(v, w) = d_{Eucl}(v_N, w_M) + min(DTW(H(v), H(w)), DTW(H(v), w), DTW(v, H(w))).$$

Le principal avantage de la DTW est qu'elle permet de comparer des séquences temporelles tout en tolérant une déformation temporelle des observations ; ce qui lui confère une souplesse et une flexibilité dans la comparaison de séquences temporelles par rapport aux distances usuelle comme la distance de Minkowski qui comparent les valeurs temporelles "point à point". La DTW permet donc de comparer des séquences temporelles de tailles différentes, en effectuant les comparaisons à l'aide d'un alignement temporel des données. La distance DTW a été utilisée avec succès notamment en recherche de données [Keogh 00, Yi 98] en reconnaissance de gestes [Gavrila 95] ou en reconnaissance de parole [Rabiner 93]. Une limite inférieure permettant de

diminuer fortement les temps de calcul pour l'indexation de séquences temporelles a été proposée dans [Keogh 02]. Un développement intéressant de la DTW appelé Derivative Dynamic Time Warping peut être trouvé dans [Keogh 01]. Récemment DTW a été utilisé avec les MMCs, cette approche tente d'éliminer les erreurs de classement générés par la méthode DTW en modélisant les classes obtenues par un MMC et déplace itérativement les séquences entre les clusters à base de leur vraisemblance obtenue sur les déférents MMCs.

3.2 Analyses des données brutes par prétraitement

L'intérêt d'effectuer un prétraitement réside a filtrer les données(valeur maquantes, bruit,etc.). Les méthodes présentées ci-dessus exploitent directement les données brutes et nécessitent de travailler dans des espaces potentiellement de grande dimension. Ces réductions de dimension permettent de diminuer le temps de calcul, ainsi que la convergence de certains algorithmes de clustering (de type EM ou k-means) [Ding 02]. De plus, les données pouvant être fortement bruitées, un prétraitement peut s'avérer nécessaire et efficace pour l'analyse de séquences temporelles. Des premières tentatives de prétraitement de données à l'aide de représentation par transformée de Fourier [Agrawal 93, Faloutsos 94] ou par décomposition en valeurs singulières [Korn 97] ont été proposées. Ces représentations ont ensuite été surpassées par les représentations à l'aide d'ondelettes [Wu 00].

Les transformées en ondelettes ont été exploitées pour l'indexation et le clustering de séquences temporelles [Popivanov 02] et ont été largement été utilisées pour représenter les signaux de façon compacte [Mallat 99]. Ces ondelettes forment un ensemble de fonctions de base de variation multi-échelles, ou multi-résolution, utilisées dans le but de l'approximation et/ou de la compression des données. Les transformées en ondelettes de Haar ont également été considérées [Chan 99, Vlachos 03b, Lin 04]. La décomposition en ondelettes de Haar s'opère niveau par niveau où chaque niveau de transformation permet de représenter plus précisément le signal. [Vlachos et al, 99] ont donc exploité ces différents niveaux de représentation et ont proposé un algorithme appelé I-k-means (pour iterative k-means) qui s'appuie sur l'algorithme de clustering k-means. A chaque niveau de représentation en transformée de Haar, l'algorithme effectue un traitement par k-means de partitionnement des données qui est est utilisé comme initialisation au niveau suivant et ainsi de suite jusqu'à ce qu'il n'y ait plus de changement dans le partitionnement de données obtenu.

3.3 Approches d'analyse à base des modèles probabilistes

Les approches décrites dans ce paragraphe font l'hypothèse que les séquences temporelles sont générées par des modèles ou par un ensemble de distributions de probabilité sous-jacentes. L'intérêt d'utiliser des modèles probabilistes pour le traitement de séquences temporelles est la prise en compte de l'incertitude associée aux observations ainsi que la possibilité, notamment avec les MMCs, de spécifier les modélisations adaptées aux différentes problématiques. Nous nous concentrerons dans la suite de ce mémoire sur les différentes méthodologies caractérisant l'analyse des séquences temporelles à l'aide de modèles de Markov cachés, plus exactement nous nous intéressons au clustering des séquences temporelles à base des MMCs.

4 La méthodologie et travaux antérieurs de MMC clustering

Dans cette section, nous présentons les travaux antérieurs sur le clustering à base de MMCs. Rabiner et al. [Rabiner, 89] ont été les premiers à utiliser un MMC pour le clustering des données temporelles. Ils ont développé deux algorithmes de clustering, à savoir un MMC à base de probabilités et un MMC avec un seuil. Suite à ces travaux, Kosaka et al. [Kosaka et al., 95] ont décrit un MMC pour un clustering par partitionnement utilisant la distance Bhattacharyya [Schweppe, 67]. [Bernhard et al, 02] propose une autre approche qui partitionne un ensemble de séquences temporelles en clusters à base des MMCs. Cette approche est motivée par l'algorithme k-means ;[Holmes et al, 00]. Plus tard, les travaux de [Smyth, 97] ont rapporté des résultats préliminaires d'un algorithme de clustering des données temporelles basées sur les modèles de mélange fini MMCs. Dernièrement l'utilisation des modèles de Markov cachés est beaucoup plus orientée sur l'étude et l'analyse des types spécifiques de données temporelles comme les trajectoires vidéo [HERVIEU, 09] et analyse des données temporelles en cardiologie [Dumont, 2009].

Les deux principales limites des approches de clustering à base des MMCs sont :

1. Le nombre de clusters n'est pas choisi d'une manière objective, il est soit à valeur définie à priori ou suite à un seuil.
2. Les clusters dans la partition sont supposés avoir une taille uniforme, spécifiée par la taille du MMC initial. Cette hypothèse ne permet pas d'aboutir à une bonne interprétation des clusters.

Dans ce qui suit, chacune de ces approches est examinée brièvement.

4.1 MMC clustering avec probabilité

Rabiner et al. [Rabiner, 89] a développé une approche de clustering qui met l'accent sur la probabilité $p(O/\lambda)$ utilisé comme critère de distribution. Étant donné un ensemble de séquences d'observations $O = (o_1, o_2, ..., o_n)$, le processus de clustering définit un MMC λ initial basée sur O. Puis on divise le modèle λ en deux, en gardant tous les paramètres des modèles fixes à l'exception des probabilités entre les états. Ces probabilités en effectuant de faibles perturbations sur de modèle original. Toutes les séquences sont affectées aux deux modèles basés sur la distance probabiliste *séquence-à-MMC*. Les paramètres des deux modèles sont estimés à nouveau après la distribution. Le processus de division est répété sur l'un des clusters dans la partition. Aucun critère d'arrêt explicite n'est fourni pour mettre fin au processus de clustering. Dans cette approche, les paramètres des deux MMCs créés à chaque étape de division sont superficiellement imposés et non pas induits à partir des données. Il est peu probable que les modèles (ou clusters)soient divisés de cette façon qui correspondent aux structures des données.

4.2 MMC clustering avec seuil

Le clustering à base des MMCs avec seuil [Rabiner, 89] tente d'améliorer la probabilité globale de données en améliorant la probabilité de séquences ayant une valeur faible. Cela se fait par l'identification des séquences dont la probabilité est faible sur les modèles actuels de la partition et de construire un modèle distinct pour tenir compte de ces séquences. L'algorithme commence par l'apprentissage d'un MMC basé sur l'ensemble des séquences d'observation. Ensuite, une valeur t comme un seuil de risque est déterminée de manière dynamique, et toutes les séquences ayant la probabilité *séquence-à-modèle* inférieure à t sont retirées de leur modèle actuel pour former un nouveau modèle λ. Après chaque extension de la partition, les paramètres de tous les modèles sont mis à jour avec les séquences d'observation en cours dans chaque cluster. La procédure est ensuite répétée sur tous les clusters dans la partition actuelle. Aucun critère explicite est fourni pour mettre fin à la procédure de clustering. le problème avec cette méthode, et sur celles à base d'un seuil, en général, est que le choix de la valeur seuil est une question de conjecture, et la performance du système de classification pourrait être très sensible à ce paramètre.

4.3 MMC clustering par reconnaissance d'erreur (CRE)

Le but de Dermatas et Kokkinakis [Dermatas et al, 96] dans le clustering par reconnaissance d'erreurs (CRE) du système est d'améliorer la précision de la reconnaissance globale

de l'ensemble des MMCs dérivés à partir des données d'apprentissage. La précision de la reconnaissance globale de l'ensemble des MMCs dans la partition de clustering est mesurée en appliquant une fonction discrète sur les probabilités de toutes les données compte tenu de ces MMCs. Initialement, un MMC unique est estimé à base de l'ensemble d'apprentissage. Si la précision de la reconnaissance est inférieure à un certain seuil, un nouveau cluster est construit. Dans un premier temps, le nouveau cluster ne contient qu'une seule séquence de données, celle ayant le plus faible risque sur le MMC en cours dans la partition. Une fois que ce groupe est formé et le MMC pour le cluster est établi à partir de cette séquence, l'ensemble des séquences sont redistribuées entre les clusters sur la base de vraisemblance *séquence-à-modèle*. Le MMC de chaque groupe est ensuite ré-estimés après chaque redistribution. Le processus d'estimation et de redistribution sont itérativement appliqués de nouveau jusqu'à ce que des séquences d'observation ne changent plus d'appartenance ou un nombre maximum d'itérations est atteint. Le système répète la procédure d'extension de la partition jusqu'à ce que une valeur seuil sur la précision de la reconnaissance soit atteinte, ou un nombre de clusters défini à priori soit construit. Cet algorithme repose sur une valeur seuil pour déterminer la partition finale de classification. Le choix d'une valeur de seuil peut être très subjectif, et peut biaiser les résultats finaux de clustering. En outre, le nouveau modèle de cluster est ajouté à partir d'une séquence d'observation unique. Les modèles construits sur la base d'une seule séquence peuvent ne pas être fiables, ce qui ajoute un autre niveau d'instabilité pour les résultats de clustering.

4.4 MMC clustering par mesures d'information théorique

La méthode [Lee, 90], avec une procédure de fusionnement, tente de généraliser des modèles tréphone pour trouver le nombre optimal de classes. Elle débute par un modèle de N clusters, un pour chacun des N contextes tréphone, une procédure d'agglomération de clusters est employée pour former *un dendrogramme* des classes tréphone par la fusion de la paire la plus similaire de clusters qui est évaluées en fonction d'une mesure d'information théorique. Après la fusion de groupes pair par pair, le processus de redistribution est ré-exécuté. Ce processus continue jusqu'à ce qu'il n'y a plus de mouvement de données entre les clusters, ou un critère de convergence est atteint. La valeur seuil, prédéterminée sur la mesure de distance, est utilisée pour déterminer le moment d'arrêt de la procédure d'agglomération.

4.5 MMC clustering par des modèles de mélanges finis

MMC clustering utilisant des modèles de mélanges finis a été largement mise en œuvre, par smyth [Symth, 1997]. Au départ il ya N clusters Singletons modéliser par N MMCs $\lambda_1; \lambda_2; ...; \lambda_N$

dont chacun est initialisé et formé sur une séquence unique de données. Une matrice de log-vraisemblance $N \times N$ est calculée (le coefficient (i, j) est le logarithme de la probabilité d'émission de la i^{eme} suite par le modèle de la j^{eme} suite), une mesure de similarité est dérivé à base de cette matrice D_{ij}. De bons résultats ont été obtenus à partir de cet algorithme [Antonilo et al, 1998] et permet de déduire les partitions naturelles sur les signaux EEG avec leurs caractères. L'une des premières limites de cette méthode, est que le nombre de clusters est fixé. D'autre part, un modèle Markovien uniforme à priori pour chaque cluster rend l'interprétabilité difficile. Une deuxième limite est qu'il est très gourmand en ressources. Le calcul de la distance initiale de matrice nécessite N^2 évaluations de probabilité plus N MMCs opérations d'apprentissage, chacune de ces évaluations est chère en calcul et en temps.

5 Bilan sur les travaux antérieurs

Nous avons discuté en détail les travaux antérieurs sur le clustering à base de MMCs. Les systèmes de clustering MMC abordés ont été conçus pour des tâches de reconnaissance vocale. L'objectif principal de clustering MMC est d'augmenter la précision de la reconnaissance en divisant l'ensemble des corpus de parole du même mot en plusieurs groupes pour créer différents MMCs simples, plutôt que de construire un MMC unique en utilisant l'ensemble des données. L'accent est mis sur la réalisation de haute précision de la reconnaissance et non pas sur la recherche du nombre de groupes approprié aux données. Tous les systèmes de clustering MMC discutés reposent sur un nombre déterminé à priori de clusters.
Lorsque l'objectif du clustering est de modéliser les données, il devient important que le nombre de MMCs dérivés des meilleures données représente le nombre de différents modèles sous-jacents dans les données. Par conséquent, il est essentiel que le système de classification soit en mesure de déterminer objectivement le nombre optimal de MMCs entièrement basés sur les données. Dans le chapitre suivant, nous présentons notre algorithme de clustering MMC qui sélectionne objectivement les meilleurs clusters en appliquant une fonction objectif basée sur le calcul de vraisemblance.
En général, la structure du modèle MMC utilisée dans la reconnaissance vocale est construite manuellement par des experts en langue avant que le processus d'apprentissage soit lancé. L'objectif de ces systèmes est d'améliorer la précision de la reconnaissance, au lieu de fournir une interprétation physique des phénomènes en cours de modélisation. En conséquence, tous les systèmes de clustering MMC que nous avons discuté supposent une structure fixe et spécifique pour les MMCs du modèle. Dans ce mémoire, nous ne supposons ni la structure des modèles

MMCs est fournie par les experts du domaine, ni que tous les modèles sont de la même structure. Pour de tels problèmes, il est essentiel que, durant le processus de clustering, le système de classification soit en mesure d'extraire automatiquement et dynamiquement la structure et les paramètres MMC pour caractériser des groupes de données. Cela correspond au traitement du problème de l'apprentissage des MMCs. Il est clair que la qualité des modèles MMC affecte directement la qualité du modèle de clustering finale. Ce mémoire propose une procédure d'apprentissage MMC efficace. En utilisant les données expérimentées suffisantes, on construit des structures MMC optimales pour les données. Notre approche sera présentée dans le chapitre suivant.

6 Conclusion

Durant ce chapitre nous avons présenté les méthodes d'analyse des séries temporelles, à savoir celles basées sur les données (avec et sans prétraitement des données). Nous avons ainsi met l'accent sur l'analyse des séries temporelles à base des modèles probabilistes.
D'autre part nous avons présenté un état de l'art suivi d'un bilan sur les différentes approches de Clustering MMC été présenté. Durant cette étude nous avons énuméré les deux principales limites partagées par ces approches de clustering MMC, à savoir le modèle de clustering (nombre de clusters et leur structure) n'est pas choisi d'une manière objective, il est soit à valeur définie à priori ou soit par un seuil. Les clusters (intermédiaire et final) dans la partition sont supposés avoir une taille uniforme, spécifiée par la taille du modèle MMC initial. Nous proposons de définir une fonction objectif basée sur le calcul de vraisemblance, qui fera le lien entre une variable supposée être la taille optimale du modèle et la vraisemblance de l'ensemble de données. L'objective principal est de trouver la taille du modèle qui maximise la vraisemblance de l'ensemble des données.

Chapitre III

Une nouvelle approche pour le clustering des données temporelles à base des modèles de Markov cachés

1 Introduction

Dans le chapitre précédent, nous avons examiné les travaux antérieurs sur le clustering des données temporelles à base des MMCs. Nous avons souligné les limites principales de ces approches pour le problème général de clustering et la modélisation de données temporelles. Ces approches utilisent une structure MMC prédéterminées et uniformes pour modéliser tous les clusters générés durant le processus de clustering. En outre, certaines approches telles que [Kosaka et al., 95], [Rabiner, 1989], font l'hypothèse que le nombre de clusters est fixé au préalable.

Pour ces approches, aucune fonction objectif n'a été utilisée afin d'évaluer et de sélectionner les modèles de chaque cluster. Dans ce chapitre, nous présentons notre méthodologie de clustering et de modélisation en fonction des MMC pour les données temporelles. Nous décrivons la méthodologie en termes de quatre étapes de recherche imbriquées et nous analysons sa complexité de calcul. Afin de réduire la complexité de calcul de l'algorithme de base, des procédures de recherche heuristiques basées sur la fonction objectif sont employées pour traiter les deux étapes clés de l'algorithme à savoir la sélection de modèle MMC pour chaque cluster dans la partition et la sélection de nombre de clusters optimal.

Notre attention dans ce chapitre est d'expliquer notre méthodologie de clustering à base des MMC est de montrer comment elle répond aux limites soulignées. Ainsi dans la première partie, nous proposons une fonction objectif adaptée pour déterminer la structure de chaque cluster. En deuxième partie nous présentons une heuristique de clustering MMC qui cherche le

nombre de clusters optimal pour l'ensemble de données en supposant que les tailles MMC sont uniformes pour tous les clusters, puis nous étudions l'intégration de la procédure de recherche de taille de modèle MMC pour chaque cluster dans l'ensemble du processus de clustering.

2 Le principe de notre méthodologie de clustering des données temporelles

Notre objectif pour la construction de structures de modèle temporels via des techniques de clustering est d'élaborer des modèles de données qui représentent et expliquent des phénomènes dynamiques dans une forme compacte et facile à interpréter. On notera qu'il existe peu d'information au préalable sur les structures de ces données. Par conséquent, il est essentiel que notre algorithme de clustering emploi des techniques objectives pour partitionner les données en clusters homogènes et élabore des modèles qui décrivent mieux les phénomènes associés à chacun des clusters.

Dans de nombreuses situations, nous sommes confrontés au problème de la différenciation entre les phénomènes, par exemple, la différenciation entre les schémas de réponse de patients à un plan de traitement spécifique. L'objectif de l'analyse est de mieux comprendre les effets d'un traitement spécifique sur les différents patients. Cela peut être obtenu en étudiant combien de modèles de réponse sont observées chez les patients, et comment ils diffèrent les uns des autres. Par exemple, dans le cas de la maladie de diabète, certains patients s'adaptent rapidement au traitement de cette maladie, dans ce cas, la dynamique de leur réponse peut être modélisée en termes d'états, tels qu'état instable, état stable, état récupérés, et d'autres patients peuvent souffrir des complications. Leurs comportements est mieux modélisée es comme : état instable ou état complication. Il existe encore d'autres patients qui ne peuvent pas supporter du tout le changement de métabolisme et leur comportement peut s'expliquer comme une autre séquence de transitions d'état : état instable, état critique et finalement la mort. Il est clair que les divers phénomènes chez les différents patients sont mieux représentés par un ensemble distinct de représentation d'états avec différentes transitions. Ce qui motive la nécessité pour l'intégration des techniques de sélection de la taille du modèle de chaque composante de notre système de clustering. L'objectif consiste à sélectionner la taille optimale du modèle MMC pour les clusters individuels qui représentent mieux les phénomènes associés à ce jeu de données.

Cela se traduit par deux tâches principales de notre algorithme de clustering :

- Trouver le nombre optimal de clusters qui représente la meilleure partition des données basées sur une mesure du critère de fonction objectif prédéterminé ;
- Construire le modèle MMC le plus approprié pour chaque cluster. Cela comprend la détermination de la taille optimale du modèle MMC et les paramètres du modèle.

Ces objectifs seront atteints par le développement :

- D'une mesure du critère objectif pour choisir la meilleure structure de partition (nombre de clusters) pour les données, et
- Une procédure explicite de sélection de la taille du modèle MMC qui détermine dynamiquement la meilleure taille MMC pour les clusters individuels au cours du processus de clustering.

  L'utilisation de ces deux points suggère la proposition d'un algorithme en quatre étapes de recherche imbriqués et interactives. Ces étapes sont :

La boucle1 : trouver le nombre optimal de clusters dans la partition. La boucle2 : trouver la distribution optimale des données aux clusters. La boucle3 : trouver la structure optimale de HMM pour chaque cluster. La boucle4 : trouver les paramètres optimaux pour les HMM de chaque cluster.

Tableau A.1 – structure de notre algorithme de clustering à base des MMC

Dans la boucle 1, la taille de la meilleure partition est déterminée en construisant progressivement la structure de la partition commençant avec une partition à un cluster et en augmentant au fur et à mesure le nombre de clusters dans la partition par un à chaque itération. L'idée est de choisir la partition qui donne la valeur la plus élevée pour la fonction de critère choisi. Dans une partition avec N séquences de données le nombre de clusters peut varier de 1 à N. Dans le pire des cas, cette boucle s'exécute N fois.

Dans la boucle suivante, pour une partition choisie de taille i, la distribution optimale de séquence sur l'ensemble des clusters est déterminée par la procédure d'affectation des séquences à cluster. Tous les i clusters possibles sur N séquences sont formés, et le clustering optimal est celui qui maximise la mesure du critère choisi. Pour chaque valeur i, il y au total i^N façon

d'attribuer N séquences à i cluster ;

La troisième boucle est consacré à la recherche de la taille optimale de modèle MMC de l'ensemble des séquences de données assignées au cluster. Une recherche exhaustive comprend la construction de modèles MMC de toutes les tailles possibles. La taille optimale du modèle peut être entre 1 et $N_k L$, où L est la longueur de la séquence de données temporelles et N_k est le nombre de séquences dans le cluster k. La recherche de la taille du modèle optimale pour chaque cluster nécessite la construction de $N_k(1 \leq k \leq i)L$ modèles. Ce processus doit être répété pour tous les clusters dans la partition.

La boucle interne de l'algorithme de recherche estime la configuration optimal des paramètres optimale au niveau de chaque cluster. En utilisant la méthode de Baum Welch discutée au chapitre I pour estimer les paramètres MMC pour N_k objets du cluster K, cela nécessite $O(LN_k)$ temps pour converger vers une configuration maximum de vraisemblance.

Pour l'algorithme proposé en quatre étapes, la complexité de recherche globale peut être exprimée comme :

$$\sum_{i=1}^{N} \sum_{j=1}^{i^N} \sum_{p=1}^{N_k L} O(LN_k)$$

Bien que cette approche puisse générer un modèle optimale de partition, elle est trop couteuse en calcul. Par conséquent, il est nécessaire d'introduire des heuristiques dans le processus de recherche pour réduire la complexité de calcul tout en s'assurant de la génération d'une bonne solution. Notre objectif est de concevoir une méthode de clustering efficace pour trouver une solution acceptable, tout en évitant l'exploration exhaustive de tout l'espace de recherche. Les deux tâches peuvent être considérées comme des problèmes de sélectionoptimale de modèle . Pour le problème d'apprentissage du modèle MMC, l'objectif est de trouver le meilleur ajustement MMC pour un ensemble donné de séquences temporelles. Il faut trouver le modèle MMC avec un nombre optimal d'états. Pour la sélection de nombre de cluster, le but est de trouver le modèle optimale en termes du nombre optimal de clusters et le meilleur ajustement MMC pour chaque cluster dans la partition.

Les deux problèmes peuvent être caractérisés comme des problèmes d'apprentissage non supervisé. L'apprentissage de la partition et la sélection de nombre de clusters consiste à trouver le meilleur ensemble de clusters regroupant les séquences d'observation. Pour les clusters

individuels dans cette partition, le problème de sélection de la taille du modèle MMC peut être considéré comme un problème de clustering secondaire, où le but est de trouver le meilleur ensemble des états MMC qui modélise les séquences temporelles attribuées à ce cluster.

Pour plus de clarté, nous proposons une approche heuristique de clustering de données temporelles à base des MMC en deux parties. La première partie se concentre sur la recherche de la taille du modèle MMC pour les séquences assignées aux clusters individuels et les valeurs de leurs paramètre d'apprentissage, c'est-à-dire l'exécution des étapes 3 et 4 du processus de clustering précédent. La deuxième partie se concentre sur la méthodologie de clustering de données temporelles, y compris la distribution de séquences aux clusters, et l'utilisation de la fonction objectif pour la sélection de nombre optimal de cluster.

Durant le chapitre 2 de l'état de l'art, nous avons identifié les principaux inconvénients de travaux antérieurs pour déterminer la structure du modèle MMC spécifique aux données. Par exemple, la Méthode [Stolcke, 94] " model fusion " est une méthode très couteuse à appliquer. Le modèle initial utilisé dans cette méthode associe un état pour chaque valeur observée, ce qui engendre un modèle de grande taille lorsqu'on traite un ensemble large d'observation. En outre, avant de fusionner toute paire d'états, nous calculons les probabilités postérieures de modèle permettant de combiner toutes les paires possibles. Cela n'est pas pratique dans le cas des grands modèles. De plus, la méthode ne permet pas de changement sur deux états fusionnés. Il est donc plus sensible à la convergence vers les modèles d'optimum locaux. De même. L'algorithme SSS [Takami et al, 92] n'emploie pas de fonction objectif permettant de comparer la qualité des différents modèles MMC. Pour remédier à ces problèmes, nous définissons une fonction objectif qui a pour but de déterminer le nombre optimal d'états pour avoir une structure cohérente de notre MMC dans une procédure de recherche heuristique simple.

3 La fonction de critère objectif

Soit un modèle M, un ensemble de données X de taille n tel que :
M est un modèle probabiliste. X est un ensemble de séquence $X = \{x_i/x_i = (o_1, ..., o_T); 1 \leq i \leq n\}$ Soit $P(x_i/M)$:la vraisemblance de la séquence d'observation x_i générée par un modèle M.

Définition $f : N \rightarrow R^{+}$
$f(\Gamma) = max_{(\Gamma=1,2,...)} \sum_{i=1}^{N} [logP(x_i/|M| = \Gamma)]$
Tel que :
Γ : La taille du modèle M.
$P(x_i/|M| = \Gamma)$: La vraisemblance de la séquence x_i sur le modèle M de taille
f : est la fonction objectif à maximiser

L'objectif de la sélection de la taille pour le modèle MMC est de choisir le MMC avec le nombre optimal d'états qui offre la meilleure représentation de données. En appliquant la fonction objectif pour un modèle M de Markov caché noté λ de paramètres (A, B, Π) et N_k l'ensemble des séquences assignées au cluster K modélisé par le MMC λ_k , le meilleur MMC est celui qui maximise la vraisemblance des N_k séquences. En d'autre termes,

$$f(\Gamma) = max_{(\Gamma=1,2,...)} \sum_{i=1}^{N} [logP(x_i/|\lambda| = \Gamma)]$$

λ_k est le modèle de Markov caché associé au cluster k.
Γ représente le nombre d'états du modèle MMC λ_k.
$P(x_i/|\lambda| = \Gamma)$: est la vraisemblance de x_i sur le modèle de Markov caché λ_k de taille Γ

4 La recherche heuristique pour sélectionner la taille du modèle MMC

Dans cette section, nous examinons comment la fonction objectif peut être utilisée pour identifier la meilleure taille du modèle MMC. On suppose que les données fournies sont suffisantes. A partir du modèle de taille minimale (un état), la fonction objectif est monotone avec la variable représentant la taille du modèle MMC, jusqu'à ce qu'elle atteint la valeur maximum et ensuite elle commence à diminuer. La valeur maximum de la fonction objectif correspond à la taille optimale du modèle MMC.

L'utilisation de cette fonction objectif offre une stratégie de recherche séquentielle avec un critère d'arrêt, contrairement à l'approche de clustering proposée précédemment qui est basée sur une recherche exhaustive par opposition à un processus de recherche exhaustive qui est implicite dans l'approche de clustering proposée. La procédure de recherche commence par le plus petit modèle MMC, (un seul état MMC). La taille du modèle est incrémentée d'un état à chaque étape. Pour chaque taille du modèle, ses paramètres sont estimés à partir des

données. A chaque étape de l'approche le modèle est évalué en utilisant la fonction objectif. Si le score du modèle actuel diminue par apport au modèle précédent, nous concluons que nous venons d'atteindre la valeur maximale et nous acceptons le modèle précédent comme meilleur modèle. Sinon, nous continuons avec le processus d'extension du modèle. La figure A.1 illustre le processus itératif.

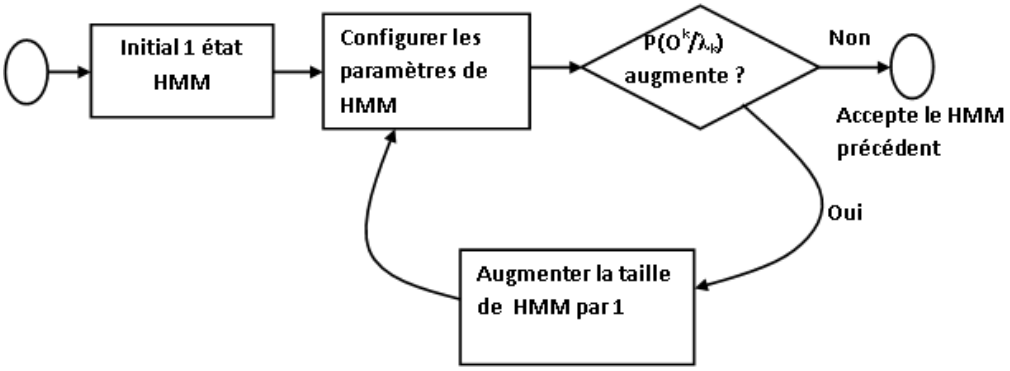

Figure A.1 – Déterminer la structure d'un cluster

Notre méthodologie est similaire à l'approche SSS (successive state splitting) [Ostendorf, 97]. Les deux approches démarrent avec un modèle de taille minimal et augmente la taille du modèle jusqu'à trouver le meilleur modèle. Cependant, notre méthode diffère de l'approche SSS de la façon dont les paramètres du modèle sont estimés. Au lieu de prendre les paramètres de tous les états dans le modèle globale, SSS effectue un ajustement local des valeurs des paramètres d'états après chaque opération de segmentation, qui est susceptible de générer des paramètres de valeurs moins optimales. Dans notre approche, à chaque étape d'extension de modèle, nous ajoutons un état au modèle et nous effectuons une ré-estimation des paramètres du modèle global à l'aide de la procédure de *Baum-Welth*.

5 Initialisation des paramètres du modèle MMC

Notre approche de sélection de la taille du modèle MMC dépend de la qualité de la procédure d'estimation des valeurs de paramètres MMC. Cela correspond à l'étape de recherche 4 dans notre processus de clustering de données temporelles générale. Elle estime la configuration de paramètre pour un modèle MMC de taille fixe à l'aide de la méthode *Baum Welch*. Cette approche est est une variante de l'algorithme EM [Dempster et al., 1977]. Elle parcourt entre une étape d'expectation (E-étape) et une étape de maximisation (M-étape). L'E-étape suppose la configuration actuelle des paramètres du modèle et calcule les valeurs attendues des statistiques

nécessaires. La M-étape utilise ces statistiques pour mettre à jour les paramètres du modèle afin de maximiser la probabilité attendue des paramètres [Ghahramani, 97]. Le processus itératif continue jusqu'à ce que la configuration des paramètres converge.

6 Notre méthodologie du clustering des données temporelle

Notre approche pour le clustering des données temporelles est basée sur l'approche par mélange de densité. Les composants (clusters) sont des modèles MMCs qui définissent le comportement temporel des séquences dans des clusters. Pour commencer, nous discutons la méthodologie du clustering par mélange fini, et son application au problème de clustering. Étant donné le nombre de clusters qui composent une partition (taille de la partition), nous examinons comment trouver la distribution optimale des séquences sur les clusters individuels. Cela correspond à la boucle deux de recherche, présentée dans le processus de clustering des données temporelles.

Ensuite, nous présentons une heuristique de clustering MMC qui cherche le modèle optimal pour la partition de clustering des données en supposant que la taille MMC est uniforme pour tous les composants. Cela correspond à la boucle interne dans le cadre de recherche de quatre étapes.

Enfin, nous mettons l'accent sur le problème de la sélection de la taille du modèle MMC discutée dans la section 4. Nous discutons la manière dont la procédure de sélection de la taille du modèle est incorporée dans l'ensemble du processus de clustering afin d'améliorer la qualité de la structure de partition clustering et les modèles associés aux clusters individuels.

6.1 Le clustering par mélange fini

Soit un ensemble de données $S = \{S_1, S_2, ..., S_n\}$ avec,$S_i = \{S_{i,1}, S_{i,2}, ..., S_{i,n_i}\}$ l'ensemble des ni séquences $S_{i,j}$ observées pour l'individu i et un nombre de classes k fixé à priori. Le principe général du clustering des données temporelles basé sur un modèle de mélange de densités consiste à :

- Sélectionner un individu i de la population,
- l'individu i est attribué à l'une des k classes (c=1...,k) de probabilité P(c)qui est la pro-

babilité a priori pour qu'une observation ait été générée par la composante c du mélange tel que $\sum_{c=1}^{k} P(c) = 1$.

– à chaque classe c correspond un modèle de génération de données $P(S_i/c_i = c, \Phi_c)$, où Φ_c sont les paramètres de cette distribution de probabilité, S_i est la donnée de l'individu i et c_i désigne la classe de l'individu i. Ce modèle permet en pratique de calculer la probabilité qu'un individu ait un vecteur S_i de données, sachant qu'il appartient à la classe c.

D'aprés ces différentes hypothèses, chaque individu i est attribué à une classe c_i $1 \leq c \leq k$. En supposant que les observations de l'individu i sont conditionnellement Indépendantes, et connaissant les paramètres du modèle de la classe c, S_i possède la densité de probabilité suivante :

$$P(S_i/c_i = c, \Phi_c) = P(S_i/\Phi_c) = \prod_{j=1}^{n_i} P(S_{i,j}/\Phi_c) \tag{III.1}$$

Pour le problème de classification automatique, les auteurs supposent que chaque observation S_i est issue d'un mélange de densité et ils cherchent à trouver les paramètres du modèle qui maximisent la vraisemblance des n observations $S = \{S_1, S_2, ..., S_n\}$, en supposant que ces observations sont indépendantes.
D'après l'équation III.1, la distribution de probabilité de S_i dont la classe c_i étant inconnue est une fonction linéaire des modèles composants. Elle est de la forme :

$$P(S_i/\Phi) = \sum_{c=1}^{k} P(S_i/c_i = c, \Phi_c) * P(c) \tag{III.2}$$

où $\Phi = \{\Phi_1, \Phi_2, ..., \Phi_k\}$ est l'ensemble des paramètres des classes c, $1 \leq c \leq k$. En considérant maintenant que les individus sont indépendants, la vraisemblance totale de l'ensemble des données $S = \{S_1, S_2, ..., S_n\}$ est donnée par l'équation suivante :

$$P(S/\Phi) = \prod_{i=1}^{n} P(S_i/\Phi) = \prod_{c=1}^{k} \sum_{c=1}^{k} P(S_i/c_i = c, \Phi_c) * P(c) \tag{III.3}$$

6.2 Le clustering par mélange de modèles de Markov cachés

Nous avons adopté pour le clustering avec un mélange fini de modèles de Markov cachés où les composantes d'un modèle de mélange fini sont représentées par les MMCs noté λ de paramètres (A,B,Π), et $p(S_i/\lambda)$ dans l'équation (III.2) est maintenant utilisée pour calculer la probabilité d'une séquence de données temporelles x_i, généré par un MMC λ_k,

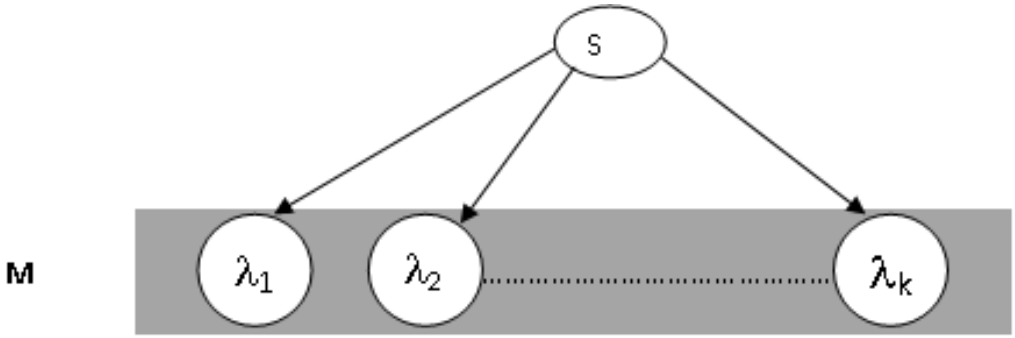

Figure A.2 – Un modèle M de K composante MMC

Le principe du clustering à base de modèles de Markov cachés est de calculer pour chaque séquence $S_i \in D$ le vecteur d'adhésion $Z_i = \{z_{i,\lambda_1}, z_{i,\lambda_2}, ..., z_{i,\lambda_k}\}$ ou z_{i,λ_i} est la probabilité que le i^{ieme} individu soit généré par le MMC i. Ensuite, nous distribuons les séquences à base des vecteurs d'adhésion (une séquence est affectée à la classe qui maximise sa probabilité de génération $max_{(1\leq j\leq k)}P(S_i/z_{i,\lambda_j})$. Nous ré-estimons ensuite les paramètres de chaque classe et refaire la distribution jusqu'à ce que le critère de convergence soit satisfait (i.e. qu'on ait atteint le maximum de vraisemblance) :

$$P(S/M) = \prod_{i=1}^{N} \sum_{c=1}^{k} P(s_i/\lambda_c) * p_c \tag{III.4}$$

Où :

$$Pc = \begin{cases} 0 & \text{si } S_i \in \text{à la classe c} \\ 1 & \text{sinon} \end{cases}$$

Les valeurs des probabilités de données calculées pour MMCs sont généralement très petites, et leurs transformations logarithmiques les rendent plus pratiques pour le calcul. L'équation III.4 sera :

$$log(P(X/M)) = log\left(\prod_{i=1}^{N}\sum_{c=1}^{k} P(s_i/\lambda_c) * p_c\right)$$
$$log(P(X/M)) = \sum_{i=1}^{N} log\left(\sum_{c=1}^{k} P(s_i/\lambda_c) * p_c\right)$$
$$log(P(X/M)) = \sum_{i=1}^{N}\left(\sum_{c=1}^{k} log(P(s_i/\lambda_c)) + log(p_c)\right)$$
$$log(P(X/M)) = \sum_{i=1}^{N}\sum_{c=1}^{k} log(P(s_i/\lambda_c)) + \sum_{i=1}^{N} log(p_c)$$

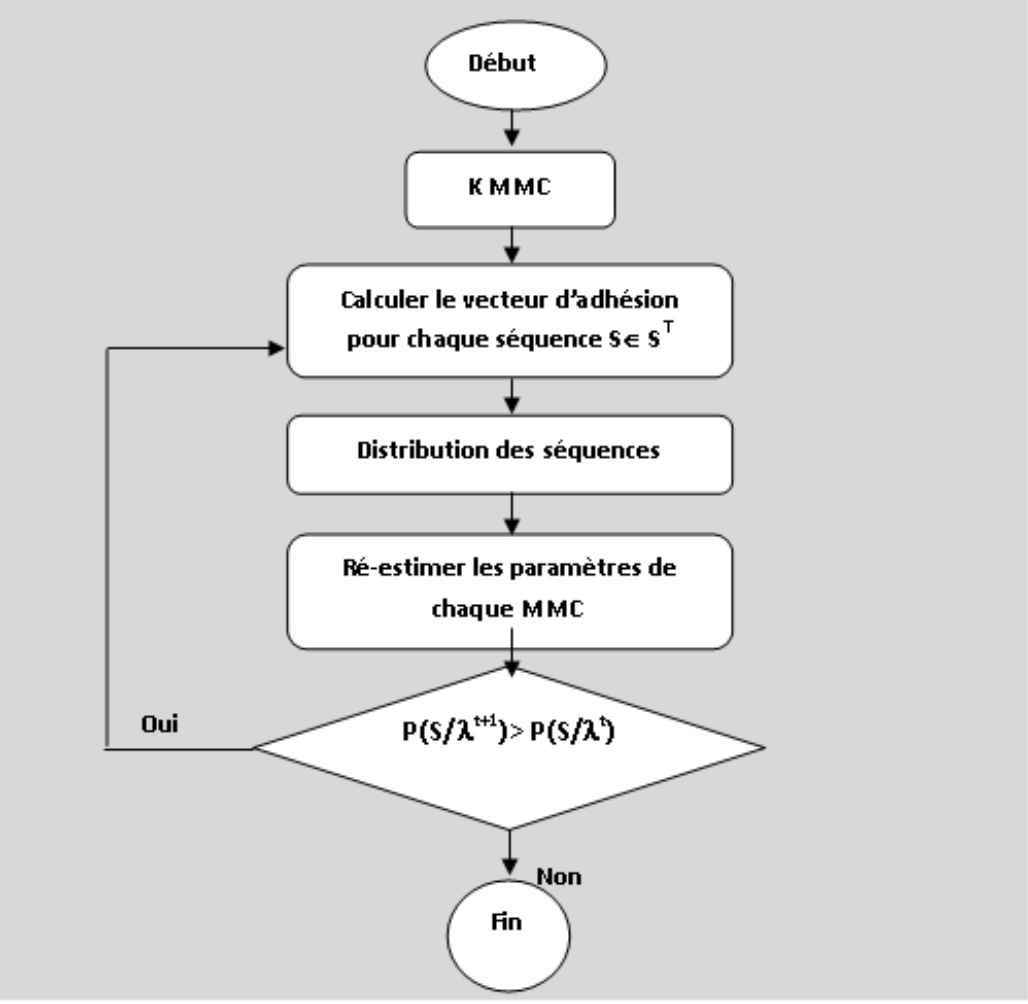

Figure A.3 – L'organigramme pour la classification par modèle de Markov caché

La méthodologie de clustering MMC proposée tente d'améliorer les méthodes des travaux antérieurs existant dans la littérature en :

1. Incorporant une fonction objectif pour la sélection de la taille du modèle MMC dans

l'algorithme de clustering afin de déterminer la structure cohérente des MMC pour chaque cluster.

2. Appliquant une fonction objectif pour sélectionner la partition optimale (déterminer le nombre de clusters ainsi leurs structures).

L'approche de clustering est décrite d'une façon incrémentale. Dans une première étape, nous décrivons un algorithme efficace pour trouver la structure optimale de la partition, c'est-à-dire, la distribution optimale de séquences aux clusters pour une taille de la partition prédéfinie. Ensuite, nous élargissons notre description de l'algorithme pour inclure la fonction objectif qui déterminera le nombre optimal de clusters avec des modèles MMC de taille uniforme. Enfin, une extension de l'hypothèse sur la taille uniforme du modèle MMC pour les différents clusters dans la partition sera discutée ainsi que la sélection de la taille optimale pour trouver le meilleur ajustement de la partition de données.

6.3 La recherche de la distribution optimale des données aux clusters

Dans la partie précédente, nous avons décrit notre algorithme de clustering de données temporelles en termes de quatre boucles imbriquées. Nous avons également discuté de l'heuristique approprié pour la structure optimale des MMCs associe aux clusters et une procédure de l'initialisation de leurs paramètres (deux boucles internes de cet algorithme). Pour mieux accélérer le processus, nous présentons une nouvelle approche pour trouver la distribution optimale de séquences aux clusters définis dans une partition de taille fixe (la boucle 2).

À cette étape, nous choisissons la mesure de distance probabiliste séquence-modèle pour la distribution des données aux clusters. On notera cette distance par P (S|λ) qui représente la probabilité qu'une séquence S générée par le modèle MMC λ. Lorsque la mesure de distance probabiliste séquence-à-MMC est utilisée pour les affectations de séquence-à-cluster, le processus de distribution applique automatiquement le critère de similarité au sein du cluster qui optimise la probabilité $P(S|\lambda)$.

$$C_i = max_{1 \leq j \leq k} P((V = S_i/\lambda_j)$$

Pour chaque partition, les appartenances initiales séquence-à-cluster sont déterminées par la distance probabiliste séquence-MMC. Les séquences sont ensuite redistribuées après la réestimation des paramètres MMC et le raffinement du modèle MMC est opéré. On réitère ce

processus jusqu'à ce qu'il n'y plus de séquence qui change d'adhésion. La figure ci-dessous illustre le processus de distribution.

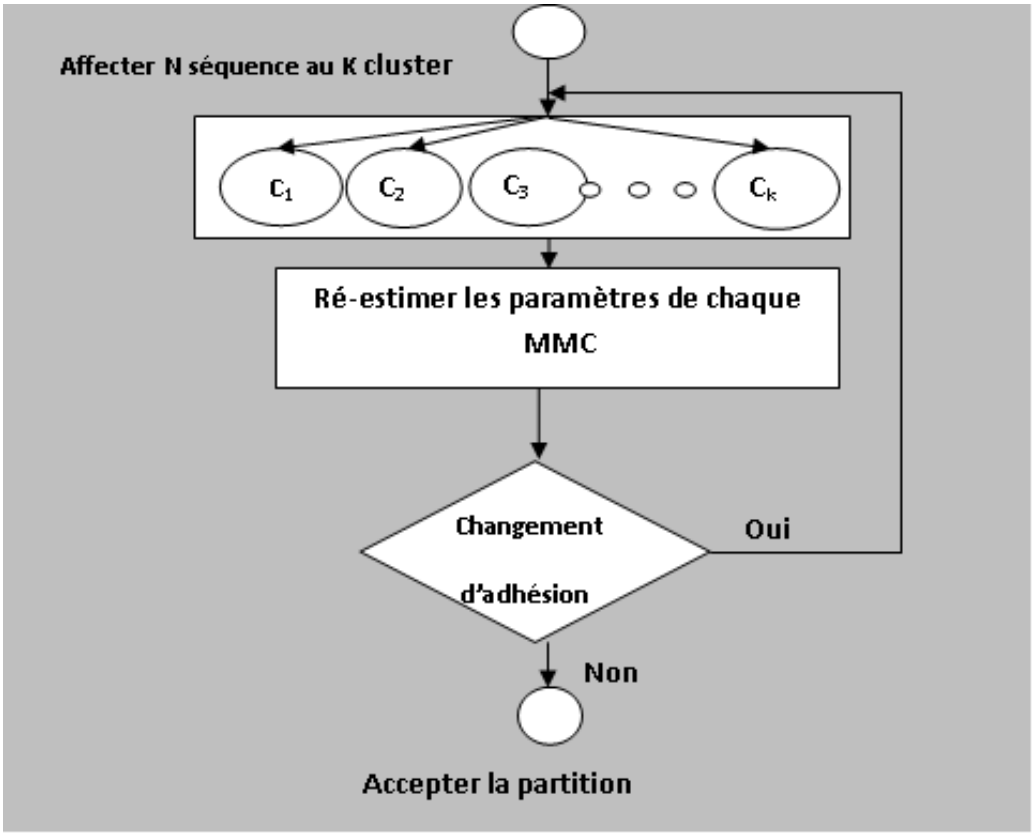

Figure A.4 – La recherche de la distribution optimale de données aux clusters

6.4 La recherche de nombre optimal de clusters avec une taille MMC uniforme

Nous avons supposé dans l'algorithme clustering MMC que la taille de la partition est connue, et l'objectif de clustering est de trouver la distribution optimale de séquence sur l'ensemble des clusters dans la partition. Dans cette section, nous généralisons notre algorithme de clustering afin de déterminer automatiquement la taille de la partition optimale pour les données. Nous supposons que suffisamment de données sont fournies, et que les modèles MMCs sont de tailles prédéterminées et uniformes.
La fonction objectif pour la sélection du nombre optimale de clusters est donnée comme suit :

$$f(\Gamma) = max_{(\Gamma=1,2,...)} \sum_{i=1}^{N} / \left[logP(x_i/|M| = \Gamma) \right]$$

Où :

M :le modèle global d'une partition qui représente un nombre fini de mélange des MMC

Γ : représente le nombre de modèles de Markov cachés dans le modèle global M d'une partition.

À l'aide de ces caractéristiques, l'algorithme commence avec le modèle de partition plus simple, c'est-à-dire un cluster. Dans les étapes suivantes, nous augmentons la taille du modèle en incrémentant d'une unité le nombre de clusters dans la partition à chaque étape. Pour chaque taille de la partition, nous trouvons la structure optimale de la partition en appliquant le processus de distribution de données. Nous évaluons la qualité du modèle par la valeur de la fonction objectif de la partition. Si sa valeur pour le modèle actuel diminue par apport à celle du modèle précédent, alors la valeur maximum est atteinte et on accepte le modèle précédent comme le modèle optimal. Sinon, nous continuons en élargissant le modèle par un cluster supplémentaire. L'algorithme ci-dessous explique les déférentes étapes.

Algorithm 5 Le clustering à base des Modèles de Markov caché avec une taille uniforme

Entrée : ensemble de données temporelles $X = x_1, ..., x_N$
Sortie : partition de modèle M=$\lambda_1, ..., \lambda_K$
K=1
Répéter
//Sélectionner un composant MMC :
S=1
Choisir au hasard une séquence
Sélectionner les n-1 séquences qui ont la plus grande vraisemblance générée avec ce composant
Pour i=1 à K **faire**
Choisir la première séquence pour le $i^{iém}$ composant, la séquence la moins présentée sur les composants précédents
Sélectionner les n-1 séquences pour le $i^{iém}$ composant
Fin pour
//**Redistribution :**
Distribué=vrai
Tant que continue **faire**
Distribuer des séquences aux clusters avec la plus haute probabilité
Si pas de changement d'adhésion **faire**
Distribué =faux
Sinon Reconfigurez les paramètres de tous les clusters avec les séquences assignées à chacun d'eux (appliquer l'algorithme de Baum Welch)
Fin si
Fin tant que
Calculer P(X/M) du modèle actuel
K=K+1
Jusqu'à $P(X/M_{actuel}) < P(X/M_{précedent})$
Accepter la partition précédente

6.5 Notre méthodologie MMC clustering avec la sélection de la taille de modèle de composant

Dans tous les algorithmes de clustering que nous avons présentés jusqu'à présent, il est supposé que tous les composants MMCs dans la partition ont la même taille. En outre, lorsqu'il est appliquée pour le clustering et la modélisation du comportement dynamique des problèmes du monde réel, cette hypothèse ne tient pas. La taille des modèles MMCs peut varier d'un cluster à un autre, et ils ne sont pas connus au préalable. Dans cette section, nous discutons comment incorporer la procédure de sélection de la taille du modèle MMC présentée dans la première

partie de ce chapitre dans le processus de clustering MMC pour déterminer automatiquement la taille optimale MMC. L' algorithme 6 présente les détails avec l'ajout d'une étape de sélection de la taille du modèle MMC indiqués en gras.

Algorithm 6 Le clustering à base des Modèles de Markov caché avec une taille dynamique

Entrée : ensemble de données temporelles $X = x_1, ..., x_N$
Sortie : partition de modèle M=$\lambda_1, ..., \lambda_K$
K=1
Répéter
//Sélectionner un composant MMC :
S=1
Choisir au hasard une séquence
Sélectionner les n-1 séquences qui ont la plus grande vraisemblance générée avec ce composant
Pour i=1 à K **faire**
Choisir la première séquence pour le $i^{iém}$ composant, la séquence la moins présentée sur les composants précédents
Sélectionner les n-1 séquences pour le $i^{iém}$ composant
Fin pour
Appliquer la procédure de sélection de taille optimale des modèles de Markov caché de chaque cluster
// **Redistribution :**
Distribué=vrai
Tant que continue **faire**
Distribuer des séquences aux clusters avec la plus haute probabilité
Si pas de changement d'adhésion **faire**
Distribué =faux
Sinon
Reconfigurez les paramètres de tous les clusters avec les séquences assignées à chacun d'eux (appliquer l'algorithme de Baum Welch)
Fin si
Fin tant que
Calculer P(X/M) du modèle actuel
K=K+1
Jusqu'à $P(X/M_{actuel}) < P(X/M_{précedent})$
Accepter la partition précédente

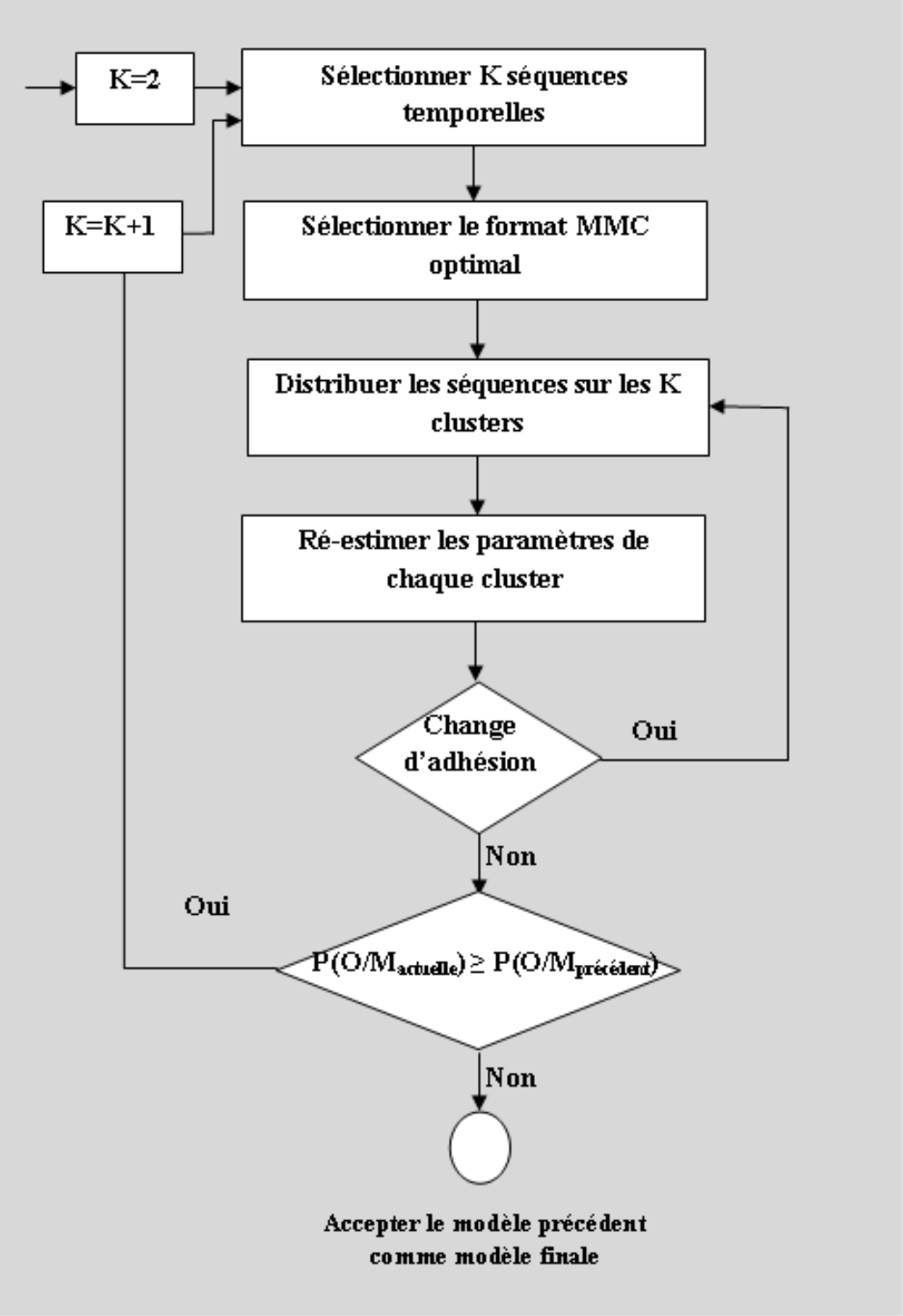

Figure A.5 – L'organigramme de l'algorithme de clustering des Séquences temporelles à base des MMC

7 Conclusion

Dans ce chapitre, nous avons proposé un framework permettant d'effectuer le clustering des données temporelle à base MMC. Nous avons analysé la complexité du processus en supposant une implémentation de 4 boucles. La complexité de calcul dans l'ensemble de cette procédure rend nécessaire l'introduction des techniques heuristiques dans l'algorithme de clustering.

Notre attention dans ce chapitre était sur la recherche de la taille optimale des modèles MMC données. Une fonction objectif a été adoptée pour l'évaluation des structures de modèle, nous commençons avec un modèle de taille minimale et nous augmentons progressivement sa taille toutes, on l'évalue. Le processus de recherche s'arrête lorsque l'extension de la taille du modèle n'améliore pas sa qualité.

Par la suite, nous avons présenté la méthodologie globale de clustering et de modélisation de données temporelles. Nous avons décrit la fonction objectif utilisée pour la sélection de la partition optimale de clustering, et finalement nous avons expliqué comment intégré l'apprentissage MMC dans le processus de clustering.

Dans le chapitre suivant, nous étudions expérimentalement l'efficacité de notre algorithme pour trouver des modèles de partition optimale. Nous menons une étude de cas pour l'analyse de la variabilité de comportement des diabétiques au cours du temps. Enfin, nous comparons la qualité des clusters générée à l'aide de notre algorithme par apport aux travaux antérieurs.

Chapitre IV

Etude de cas sur le diagnostique médical de la maladie du diabète

1 Introduction

Durant le chapitre précédent, nous avons présenté en détail notre méthodologie de clustering des données temporelles à base de modèles de Markov cachés pour déterminer le nombre optimal de clusters ainsi leurs structures.

Ce chapitre présente une étude de cas menée sur des données réelles du secteur médical à savoir la maladie de diabète et ses complications. Le but de cette étude est de répondre à un besoin intense et évolutif pour assurer les moyens et techniques ainsi d'améliorer la qualité des soins et par conséquent d'assurer une bonne qualité de vie pour les patients.
La section1 décrit le contexte de notre étude sur la maladie du diabète dans le cadre d'analyse des données temporelles comme un problème de clustering et l'apport de notre méthodologie de clustering à base des modèles de Markov cachés, puis une description et une modélisation des données traitée est présenté dans la section 2, enfin nous présentons l'application de notre algorithme et les résultats dégagés. D'aprés notre étude, nous avons identifié 4 clusters pertinents modélisant ainsi les différents comportements d'un diabétique. Ces résultats ont été obtenue par l'application de notre approche de clustering baseé sur les MMCs qui offre une meilleure modélisation pour l'évolution des séquences au niveau de chaque cluster et par conséquent obtenir une meilleure interprétabilité des clusters construits.

1.1 Description de la problématique dans le cadre de recherche

Le diabète de type 2 [1] représente un problème majeur de santé publique. Il affecte environ dix millions de citoyens européens, de tout âge et il est une cause principale de décès.

Saint-Vincent a publié en 1989 des travaux sous l'égide de l'Organisation mondiale de la santé (OMS) et la Fédération internationale du diabète (IDF)et a souligné la gravité de cette maladie et recommandé les actions d'urgence. Les principaux objectifs de la déclaration de Saint-Vincent est de prévenir et de soigner le diabète et ses complications et d'améliorer durablement l'état médical d'un diabétique permettant ainsi d'assurer une bonne qualité de vie.

D'un autre côté, les patients effectuent une fiche d'analyse et des soins régulièrement au cours du temps qui permet de suivre leur état de santé. Ces opérations produisent beaucoup de données qui se caractérisent par leur dimension temporelle. L'analyse de ces données temporelles nous permet de détecter la variabilité du comportement d'un diabétique ainsi de déduire son évolution dans le temps.

Le traitement efficace de ce volume important de données se ramène à un problème de clustering (classification non supervisée) de données temporelles qui a pour but de partitionner les données en clusters homogènes modélise les différents comportement des patients, afin de pouvoir identifier la classe d'un nouveau patient en exploitant son historique et d'évaluer ainsi son comportement futur (prévision).

Notre méthodologie de clustering à base des modèles de Markov caché est une méthode adaptée aux données temporelles et suppose que les séquences de ces données temporelles sont générées par un modèle de Markov caché. En effet, il existe une forte correspondance entre les MMC et les séquences temporelles, les états cachés d'un MMC peuvent être utilisés pour modéliser l'ensemble des états potentiellement valides dans un processus dynamique (dans notre cas le diabète est considéré comme étant un processus dynamique). Tandis que l'ensemble des états ou la séquence des états qui interviennent dans le système sont cachés et peuvent être estimés en observant le comportement des patients. Notre objective est de modéliser les différents comportements des patients diabétiques puis les partitionner en cluster homogène, afin d'effectuer le classement d'une nouvelle séquence et de déduire ses états futurs. L'intérêt

1. Appelé également diabète gras ou de la maturité, le diabète non insulinodépendant (DNID) est une maladie métabolique caractérisée par un excès chronique de sucre dans le sang (hyperglycémie).

de cette étude est d'offrir un moyen efficace pour la prévention des complications dangereuses et coûteuses ce qui permettent une réduction des taux de cécité, d'amputation, d'infarctus du myocarde.

2 Description et modélisation des données temporelles

2.1 Présentation des données

Dans le but d'étudier la variabilité des résultats en fonction du temps des analyses sur des patients diabétiques sous forme de séquences de données sont représentés sous forme d'un historique sur des patients diabétiques couvrant 4 années. L'ensemble des séquences contient 23 attributs nécessaires pour reconnaitre le statut du diabète d'un patient et la base de données contient environ 4000 patients par an.

La complexité de la maladie du diabète implique la participation de plusieurs personnes (infirmier, médecin spécialistes, généralistes) où chacun collecte les informations relatives à son expertise. Cette information individuelle n'est pas utilisable telle qu'elle se présente mais un prétraitement a priori est appliqué avant de lancer le processus de clustering. Un échantillon d'une séquence est donné dans la figure suivante.

Les données sont de type catégoriel de format séquentiel représenté comme suit : X_i=$\{y_1, y_2....y_T\}$ où : y_i=$\{x_1 : Type1, x_2 : Type2..., x_t : Typet\}$ représente les différents informations collectées à l'instant i pour le patient X_i.

Nous avons identifié, pour la description de nos séquences, 23 attributs à savoir 8 de type numérique (à savoir *l'âge, la taille, la longueur, la tension artérielle, la mesure de HBA1C, créatinine, albuminurie, cholestérol)*, 15 de type binaire (à savoir, *le sexe, le diabète famille, la Nicotine, la présence ou l'absence de la cécité, l'amputation, l'infarctus, l'AVC (accident vasculaire cérébral) et l'insuffisance rénale,etc.).*

CAMPAIGN	SEX	DM_TYPE	NICOT_YN	WEIGHT	HEIGHT	BP_SYS	BP_DIA	HBA1C
1994	2	2	2	55	147	150	80	7,1

CREA	ALBUMINURI	PROTEINURI	CHOL	AGE	insrenal	infarc	avc	cécité	amput
64	18		5,7	84			1	0	48,125

Figure A.1 – Un échantillon d'une séquence brute d'un patient

2.2 Description des données

Codage	Description	Type	Valeur
Le sexe	le sexe du patient	entier	0: masculin, 1: féminin
DM_TYPE	le type de diabète	entier	1 : diabète insuline-dépendant, 2: diabète non insulinodépendant
NICOT_YN	Le tabac	entier	0: fumeur 1: non-fumeur
WEIGHT	le poids	réel	En cm
HEIGHT	la taille	réel	En cm
BP_SYS	Pression artérielle systolique	entier	entier
BP_DIA	Pression artérielle diastolique	entier	Entier
HBA1C	L'hémoglobine glycosylée	réel	Réel(%)
CREA	Créatinine	réel	réel (en µmol/l)
ALBUMINURI	Micro-albuminurie	réel (en mg/j)	réel (en mg/j)
PROTEINURI	Protéinurie	Binaire	1 : oui 2 : non
CHOL	Cholestérol	réel	réel (en mmol/l)
Insrenal	insuffisance rénale	binaire	1 si clairance[1] (*)<15 ou sv_dialy =1 ou sv_dialy12=1 (dialyse ancienne /récente) 0 si les 2 sont à non (2) et clairance >=15 manquant dans les autres cas
Infarc	Infarctus	Binaire	1 si sv_m_inf =1 ou sv_m_inf12=1 (ancien ou récent) 0 si les 2 (ancien/récent) sont à NON (2), manquant sinon

avc:	Accident vasculaire cérébral	Binaire	1 si sv_strok=1 sv_strok12=1 (ancien ou récent) 0 si les 2 (ancien/récent) sont à NON (2), manquant sinon
cécité	:perdre de vue	Binaire	1 si sv_strok =1 ou sv_blind_12=1 (ancien ou récent) 0 si les 2 (ancien/récent) sont à NON, manquant sinon
amput	Amputation	Binaire	1 si sv_a_ank=1 ou sv_a_ank12=1 ou sv_b_ank=1 ou sv_b_ank12=1 (ancien ou récent) 0 si les 4 sont à NON (2) , manquant sinon
sv_dialy sv_dialy_12[1] sv_strok sv_strok_12	Insuffisance rénale ancienne/récente Accident vasculaire cérébral ancien/récent	Binaire Binaire	entier : 1. Oui, 2. Non entier : 1. Oui, 2. Non
sv_a_ank sv_a_ank12	Amputation supérieure à la cheville ancienne/récente	Binaire	entier : 1. Oui, 2. Non
sv_b_ank sv_b_ank12	Amputation inférieure à la cheville ancienne/récente	Binaire	entier : 1. Oui, 2. Non
sv_blind sv_blind_12 sv_m_inf sv_m_inf_12	Cécité ancienne Cécité récente Infarctus ancien Infarctus récent	Binaire Binaire	entier : 1. Oui, 2. Non entier : 1. Oui, 2. Non

Tableau A.1- Déscription de données

2.3 La modélisation des séquences de données

Nous avons remarqué qu'il existe une forte correspondance entre les attributs au cours du temps. En effet, les médecins déduisent que l'augmentation du critère HBA1C influe directement sur d'autres critères comme l'AVC (Accèdent Vasculaire Cérébrale), la maladie coronienne, cardiovasculaire. Ils ont déduit qu'un diabétique passe généralement par 3 états principaux qui sont : Equilibré, Variable, Mauvais. Les médecins jugent qu'un patient diabétique est en état stable s'il ne présente pas de facteur de risque, sinon il est dans un état variable, avec la présence

et il est dans un mauvais état s'il y a une complication.

Les transitions entre les états correspondent à l'évolution de l'état du patient dans le temps. Il peut être en état stable puis passe en état variable avec une probabilité P_{sv} La matrice de transition est initialisée suite à une matrice équiprobables.

$$A = \begin{bmatrix} 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \end{bmatrix}$$

Nos séquences de données seront transformées à des séquences de symboles déterminés avec l'intervention des médecins diététique à l'aide des attributs suivant :

a. *Les facteurs de risque*

- **HBA1C** : *L'hémoglobine glycosylée ou glyquée* (fraction HbA1C) est une valeur biologique permettant de déterminer la concentration de glucose dans le sang, la glycémie sur une période de 3 mois. Elle est particulièrement utile et constitue le paramètre de référence dans la surveillance de l'équilibre glycémique des patients diabétiques.
- ***Cholestérol*** : L'excès de mauvais cholestérol dans le sang ou hypercholestérolémie est un facteur de risques majeur des maladies cardio-vasculaires.
- ***ALBUMINURI*** : L'albuminurie correspond, plus spécifiquement, à la présence dans les urines d'une variété particulière de protéine : l'albumine qui est un facteur déclencheur d'une insuffisance rénale.
- ***PROTEINURI :*** Le terme protéinurie désigne la présence de protéines, de n'importe quelle nature, dans les urines.

Les médecins jugent que ces critères nous informent sur l'état du patient. En effet, si leurs valeurs sont dans les normes donc son état est équilibré sinon il sera un facteur de risque pour le début de complications

b. les complications Les complications liées au diabète ont une origine commune : l'excédent de glucose dans le sang. La présence d'une trop grande quantité de glucose dans le sang a des effets néfastes sur les reins (néphropathie), les yeux (rétinopathie), le système neurologique (neuropathie), le cœur (infarctus) et les vaisseaux sanguins (hypertension, artériosclérose, accident vasculaire cérébral, etc).

Un malade atteint d'une **cécité** ou **infarctus**, **neuropathie**, une **insuffisance rénale**, etc. est en état de complication.

Afin de déterminer l'état du patient à l'instant t, nous analysons les données prises à l'instant t. Un patient qui ne présente pas les facteurs de risque est considéré équilibré sinon il est en cas de risque et s'il a une des complications liée à la maladie il est en état de complication. À cette étape, nous avons transformé nos données en séquence de Symboles tels que :

E : équilibré.
R : en risque de complication.
C : en complication.

EER : une séquence qui est en état équilibré puis passe en état de risque

Nous pouvons modéliser nos données à l'aide d'un HMM défini comme suit :

- **L'alphabet S**= {stable, variable, mauvais} rerésente les états de la chaîne de Markov.
- **La matrice de transition A**= $a_{ij} = P(s_j|s_i)$: est la probabilité de passer de l'état si à l'instant t à l'état sj de la prochaine étape.
- **Les probabilités de départ** Π= π_i : définit la probabilité qu'un état donné soit l'état initial dans la séquence.
- **L'alphabet V**= {E, R, C} des symboles émis par les s_i pour un HMM discret.
- **Les probabilités d'émission B**=$b_i(E) = P(E|s_i)$: la probabilité de produire un dispositif évalue pour n'importe quel état donné.

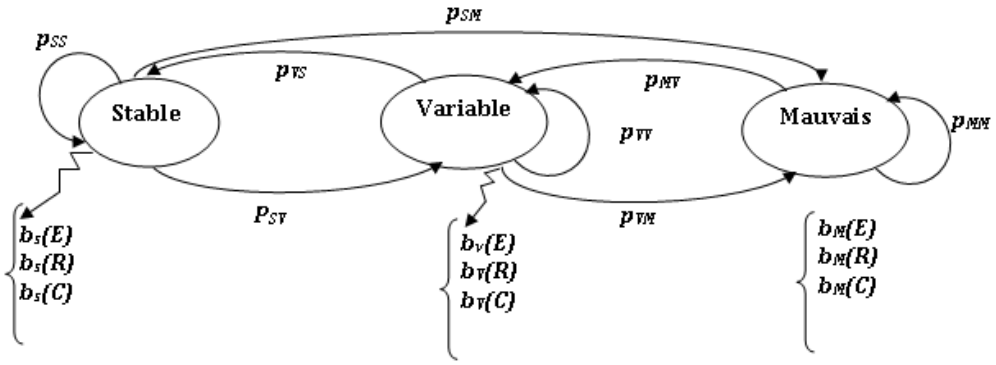

Figure A.2 – Un MMC généralisé modélise une séquence de données d'un diabétique

L'initialisation de notre MMC est effectuée suite à une étude statistique menée sur l'ensemble de données. Nous avons calculé l'effectif des patients qui débutent leur diabète en état équilibré, à risque ou en complication pour déterminer les probabilités initiales, puis nous effectuons le même raisonnement pour calculer le nombre moyens des patients qui passent d'un état s_i a l'état s_j afin d'initialiser la matrice de transition A. Pour la matrice B, nous l'avons initialisée par une loi équiprobable discrète.

$$A = \begin{bmatrix} 0.67 & 0.23 & 0.07 \\ 0 & 0.5 & 0.5 \\ 0 & 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.61 & 0.3 & 0.09 \\ 0.2 & 0.7 & 0.1 \\ 0.07 & 0.12 & 0.81 \end{bmatrix}$$

$$\Pi = \begin{bmatrix} 0.6 \\ 0.3 \\ 0.1 \end{bmatrix}$$

3 Application de notre algorithme de clustering MMC

Après avoir transformé nos données brutes en séquences homogènes, nous avons divisé notre ensemble de données en 3 tiers, Deux tiers pour l'apprentissage et un tiers pour le test. Les deux tiers d'apprentissage nous a permis de déduire le nombre de clusters ainsi leur structure cohérente et le tiers de test permis de tester la classification de nouvelles séquences. Les expériences sont décrites dans le reste de ce manuscrit.

3.1 Détermination de la structure des clusters

Cette étape tente de remplacer un modèle MMC existant pour un cluster de séquences par un modèle MMC plus précis et plus raffiné. Nous démarrons avec une configuration du modèle initial et progressivement nous augmentons le modèle via les états MMC pour choisir le modèle ayant une taille optimale. L'objectif est d'obtenir un modèle qui propose une meilleure modélisation des données, c'est-à-dire, avoir la plus grande vraisemblance de l'ensemble de séquences sur le modèle.

• **Description de l'expérience** Nous avons sélectionné un groupe de patient qui présente un état équilibré selon les médecins, puis nous avons tenté de trouver le MMC adéquat qui

représente au mieux l'utilisation de notre procédure de recherche en variant la taille des MMC est représenté par a figure A.3 suivante :

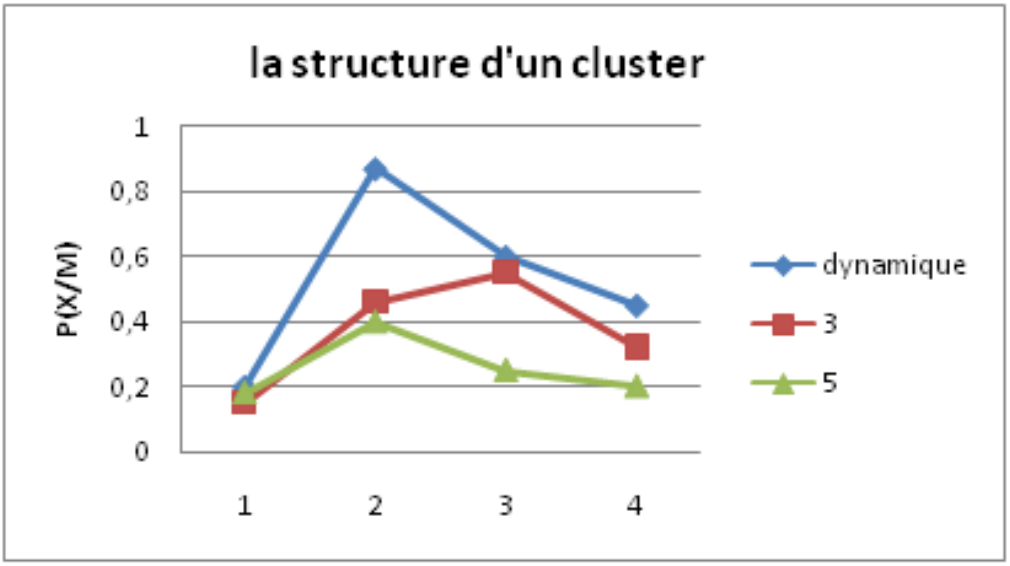

Figure A.3 – la variabilité de la vraisemblance globale en fonction de la taille MMC associe à un cluster

Nous remarquons qu'à travers cette expérience, nous avons montré l'efficacité de notre algorithme pour la sélection du modèle MMC cohérent à nos clusters. Le MMC de taille 3 arrive à modéliser les séquences de données avec une vraisemblance de 0.55 après l'exécution de 3 itérations de l'algorithme, tandis que le MMC de taille 5 arrive à les générer avec une vraisemblance de 0.4 après 2 itérations.

Notre procédure de sélection de la taille initialise le MMC à un état puis incrémente sa taille d'un état à chaque itération, cette dynamique offre une souplesse et une flexibilité dans la recherche de la taille adéquate du MMC, comme le montre la figure ci-dessus notre procédure sélectionne un MMC de taille 2 comme modèle optimal qui permet de représenter les séquences avec une vraisemblance de 0.87 et avec 2 itérations seulement.

3.2 Déterminer le nombre de clusters

À fin de montrer l'efficacité de notre algorithme pour la sélection de nombres optimale de clusters nous avons mené deux tests. Le premier avec des modèles de clusters de taille fixe et le second pour examiner l'apport de la procédure de sélection de taille de modèle MMC incorporé dans l'ensemble du processus clustering.

•Description de l'expérience 1 : Nous avons exécuté notre processus de clustering par la construction des modèles de taille fixe. Nous commençons par le modèle le plus simple à un cluster modélisé par un MMC de taille 3. Dans les étapes suivantes, nous allons progressivement augmenter la taille du modèle en augmentant le nombre de clusters (toujours modélisé par des MMC de taille fixe), à chaque itération du processus de clustering nous trouvons la structure optimale de la partition en appliquant le processus de distribution de données. Nous évaluons la qualité du modèle par la valeur de la fonction objectif de la partition. Les résultats sont montrés sur la figure A.4 :

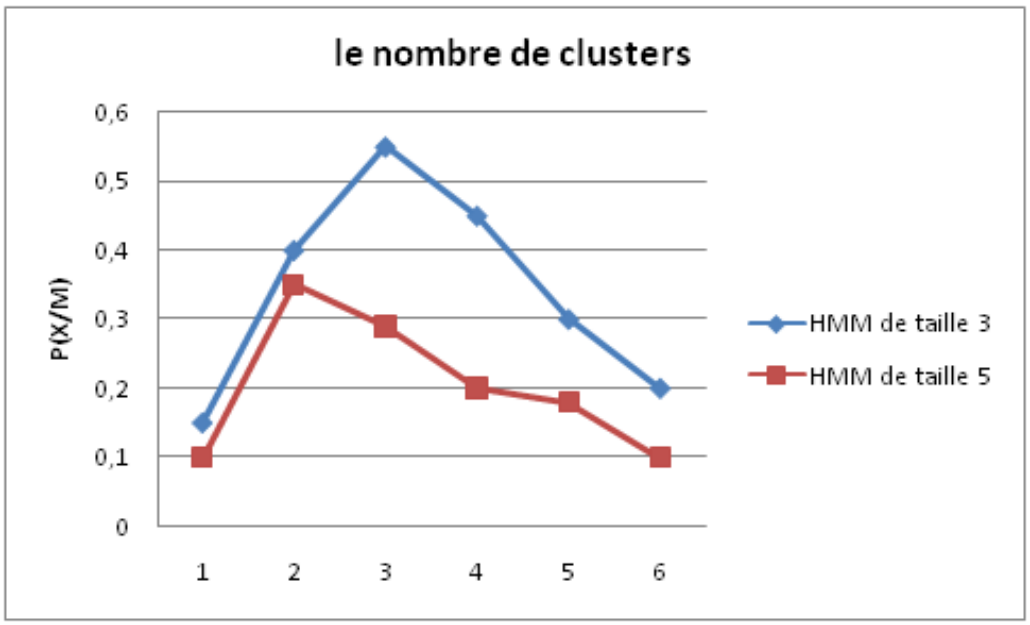

Figure A.4 – le nombre de clusters avec un mélange des MMCs de taille fixe

Cette figure montre l'influence de la variabilité de la taille des clusters sur la vraisemblance globale de la partition interpréter comme une mesure inter-clusters, avec des clusters modélisés par des MMCs de taille 3. La vraisemblance de la partition atteint 0.6 après avoir obtenu 2 clusters et diminue tout en augmentant le nombre de clusters. Avec des clusters de taille 5, on atteint une mesure d'inter-clusters de 0.35 et on obtient 3 clusters seulement. En effet, le MMC de taille 5 est largement suffisant pour générer plusieurs séquences de différent comportement cela ne permet pas d'offrir une bonne représentation.
Dans l'expérience suivante, nous avons exécuté notre procédure de sélection de taille de modèle MMC incorporé dans l'ensemble du processus clustering afin d'améliorer la qualité de la structure de partition clustering et les modèles associés aux clusters individuels.

•Description de l'expérience 2 Cette expérience commence par un modèle avec un cluster modélisé par un MMC de taille 1. La figure A.5 représente l'évaluation de notre modèle avec un nombre variable de clusters.

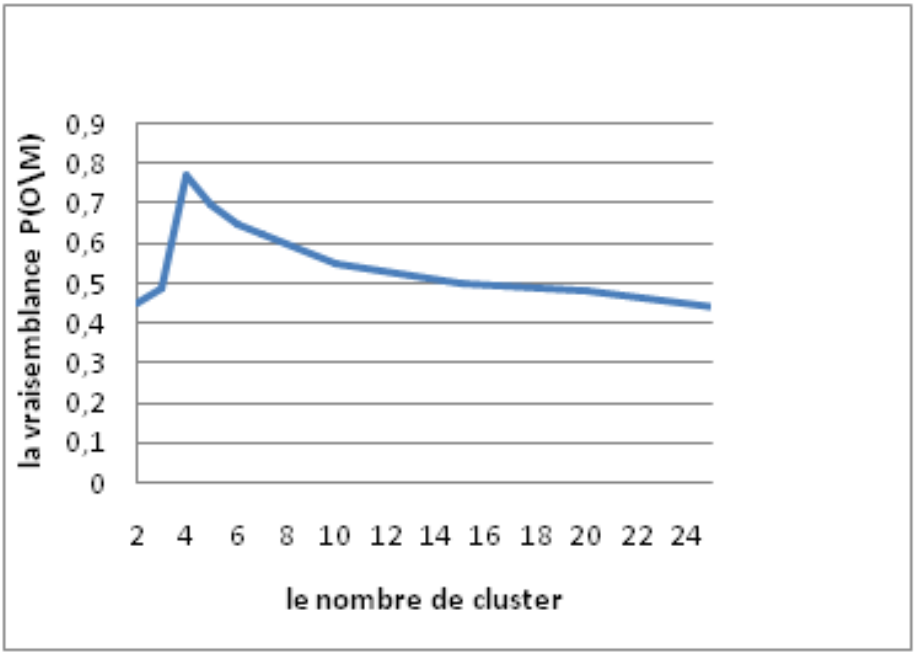

Figure A.5 – Le nombre de clusters déduit avec notre approche

Nous avons commencé avec 1 cluster, initialisé par un MMC de taille 1 formée sur l'ensemble de séquences. Une séquence est affectée aux clusters qui maximise sa vraisemblance. Le processus de distribution continue tout en mettant à jour les paramètres de chaque cluster. Cela permet aux séquences de changer leurs adhésions. Notons qu'après chaque distribution, la procédure de recherche de taille de modèle MMC est exécutée au niveau de chaque cluster, et avant toute extension de la partition, nous évaluons le modèle global par le calcul de la fonction objectif les résultats montre qu'avec 2 clusters la fonction objectif est égale à 0.45. En ajoutant un cluster, elle atteint 0.5 et nous remarquons que plusieurs séquences sont migrées vers le nouveau cluster, car ce dernier leur offre une meilleure modélisation et pour 4 clusters la fonction objectif atteint sont maximum avec une valeur égale à 0.77, puis plus que les nombres de clusters augmente plus la fonction objectif diminue.

Nous remarquons que l'intégration de la procédure de sélection de taille du modèle MMC dans l'ensemble du processus clustering améliore la qualité de la structure de partition clustering et les modèles associés aux clusters individuels. Ceci conduit également à une meilleure interprétabilité des clusters obtenue :

- *Le premier cluster* : regroupe des sujets équilibrés qui ont leur diabète stable pendant les 4 ans et n'ont pas de complication ni des déséquilibres fréquents dans leur suivie thérapeutique.
- *Le deuxième cluster* : regroupe des sujets qui présentent un facteur de risque généralement c'est des séquences suivies pendant 1 à 2 ans.
- *Le troisième cluster* : regroupe des sujets qui présentent une complication, un accident vasculaire cérébral, une cécité, etc.
- *Le quatrième cluster* : regroupe des sujets qui présentent plusieurs complications à la fois.

3.3 Détermination de la structure des clusters

Le clustering avec MMC de taille fixe			Clustering avec MMC de taille dynamique
3	8	15	
26,14%	41,23%	60,16%	13,78%

Tableau A.1 – Les séquences mal classées

Afin de montrer l'influence du changement dynamique de la structure des clusters, nous avons mené 3 tests avec notre approche en fixant la taille des MMC de chaque cluster à 3 pour le 1^{ier} test, 8 pour le $2^{iéme}$ et 15 pour le $3^{iéme}$. D'autre part, nous avons exécuté notre algorithme pour chercher la taille optimale de l'MMC associé à chaque cluster d'une manière dynamique. Après chaque évaluation du modèle, nous tentons de chercher le nombre d'états MMC qui proposent une meilleure modélisation des données associées à chaque cluster et nous avons une taille égale à 3 pour le cluster 1 et respectivement de 3 et 2 pour le cluster 2 et 4. Le changement de la taille de MMC de chaque cluster d'une manière dynamique réduit le taux de séquences mal classées. Avec un MMC de taille fixe, nous aurons 26.14% de séquences mal classées alors qu'avec des MMC de taille dynamique le taux de séquences mal classées est réduit de 50% ; ce qui est un score intéressant pour le raffinement des résultats du clustering.

Le cluster1 : ce cluster modélise les séquences qui basculent entre l'état stable et l'état variable, mais se termine toujours par s'équilibrer.

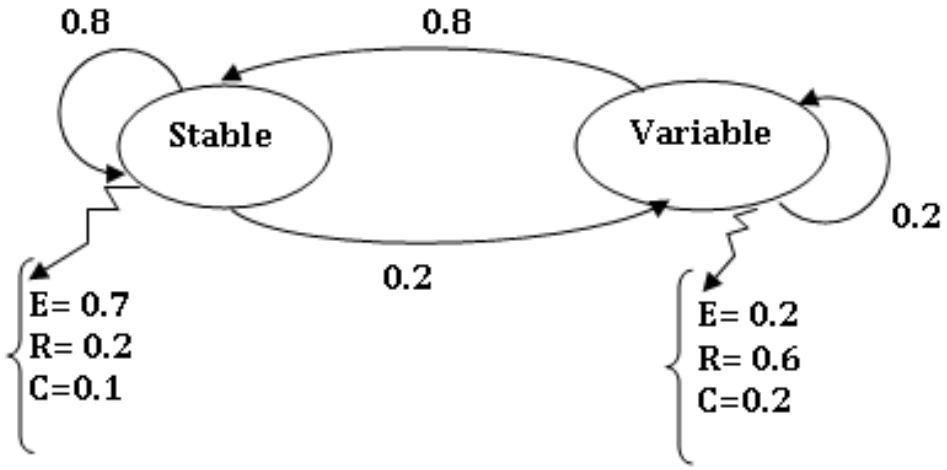

Figure A.6 – La structure de cluster 1 avec un MMC de taille 2

Le cluster 2 : les séquences des patients en complication suite à un déséquilibre ou un événement brutal comme une hypertension, un accident vasculaire cérébral,etc.

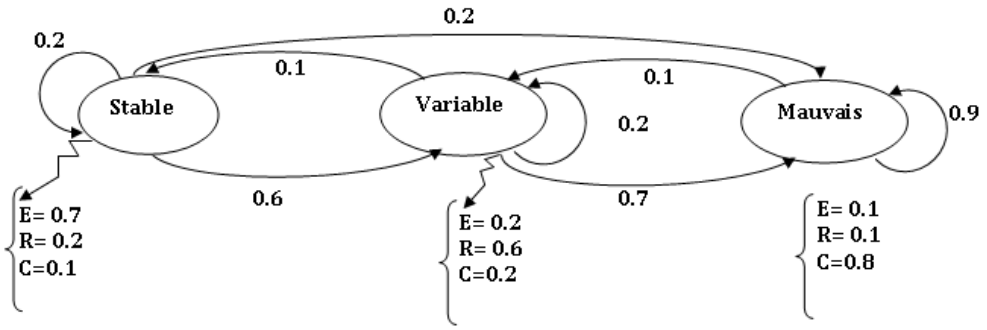

Figure A.7 – La structure de cluster 2 avec un MMC de taille 3

Le cluster 3 : ce cluster modélise le comportement des séquences qui ont un caractère variable. Il est interprété par les médecins comme des sujets à risque de complication nous remarquons que l'état variable absorbe ces séquences.

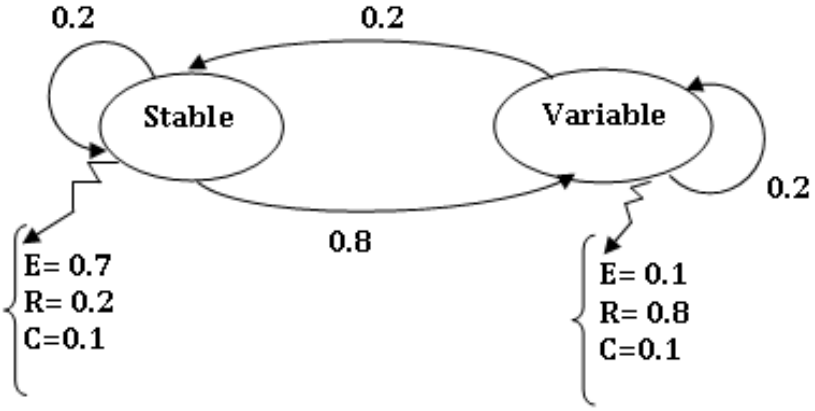

Figure A.8 – La structure de cluster 3 avec un MMC de taille 2

Le cluster 4 : ce cluster modélise le comportement des séquences ayant de multiples complications. Nous avons remarqué que la transition de l'état mauvais à l'état variable est considérablement grande ; ce qui interprètent l'évolution d'une complication chez le patient.

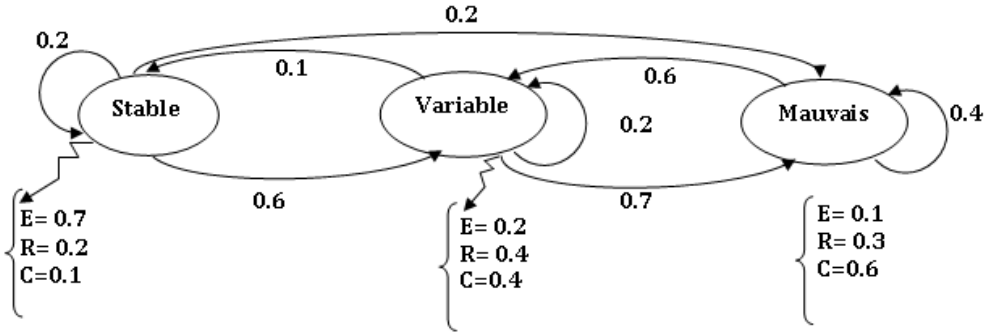

Figure A.9 – La structure de cluster 4 avec un MMC de taille 3

Nous remarquons que les MMCs modélisent clairement le comportement des séquences appropriés cela facilite l'interprétation des clusters ainsi que l'intégration de nouvelles séquences est rendue facile.

La création de chaque MMC est liée au comportement de la partie de données qu'il modélise et grâce à l'apprentissage de ces derniers, nous pouvons mieux modéliser leur comportement.

3.4 Etude comparative de notre approche

Nombre de cluster	Notre approche	Smyth
2	26,63%	48,22%
3	16,37%	32,81%
4	13,55%	19,2%
5	16,44%	25,27%
10	50,33%	48.22%
20	78,2%	67,67%

Tableau A.2 – Etude comparative de notre approche

Les résultats présentés dans le tableau ci-dessus montrent clairement que notre approche donne de meilleurs scores comparées à l'algorithme de Smyth. En effet, l'utilisation d'une fonction objectif à base de vraisemblance pour l'évaluation de la partition a montré dans cette étude son efficacité dans l'amélioration de la qualité des résultats de classification. En plus de la capacité de notre approche à déterminer le nombre de cluster ainsi leur structure, notre approche converge en moins de 10 itérations alors que l'approche de Smyth obtient un résultat beaucoup moins important après une exécution d'au moins 20 itérations.
Ainsi à l'aide d'une interprétation des résultats par des spécialistes (médecin) nous avons pu regrouper les patients en quatre classes : sujets équilibrés, sujets à risque de complication, sujets en complication et sujets à multiples complications. Les spécialistes ont trouvé nos résultats très intéressants. Ceci leur permettra de suivre les sujets à risque pour éviter d'éventuelles complications.

4 Conclusion

À travers ce chapitre, nous avons présenté un des problèmes majeurs lié à la santé publique à savoir le diabète. La nécessité de développer des outils d'analyse, afin d'évaluer et d'améliorer la qualité des soins du diabète.

L'utilisation de notre approche sur une base de données de patients diabétiques de type 2(non insoulino dépendant) a détecté 4 comportements qu'un diabétique peut avoir ainsi grâce aux MMCs qui offre une meilleure modélisation pour l'évolution des séquences au niveau de chaque cluster et par conséquent obtenir une meilleure *interprétabilité* des clusters construits, peut être appliquée pour classer de nouvelles séquences et estimer leurs évolutions futures.

Le premier comportement est celui des sujets équilibrés. Le deuxième concerne les sujets en complication. Le troisième détecte les sujets à risque de complication et permet au médecin de réagir afin de leur éviter d'éventuelle complication. Le dernier comportement est lié aux sujets à multiples complications.

Nous avons mené une étude comparative qui a montré l'efficacité de notre approche face à l'approche de (smyth ; 1997). En effet, la recherche de structure cohérente aux clusters à travers les MMCs montre leur efficacité interprétée par le taux minime de séquence mal classé face aux autres approches, ainsi que la recherche de nombre optimal de clusters avec la fonction objectif qui accélère la convergence de l'approche.

En perspective, nous envisageons d'appliquer cette approche à d'autres domaines, comme l'analyse des données financières ou les données écologiques.

Conclusion générale et perspectives

5 Bilan de notre contribution

Dans le cadre de ce mémoire, nous avons proposé une solution à la problématique de *l'analyse de séquences temporelles*. L'objectif était d'appliquer des techniques d'apprentissage non supervisées pour créer des modèles de processus dynamiques à partir de données afin d'offrir une bonne interprétation aux clusters obtenus.

Pour y parvenir, nous avons proposé un algorithme de clustering à base des MMCs pour les données temporelles. Le modèle de Markov cachés (MMC) possède plusieurs propriétés intérissantes pour modéliser le comportement des données temporelles. Le principal problème qui empêchait cette méthodologie d'être appliqués pour la modélisation dans différents domaines est la nécessité de prédéfinir la taille de MMC dans le processus de construction de modèle. Dans certains domaines, tels que le traitement de la parole, l'application de cette méthodologie à l'aide de structures MMC tirés d'analyse d'expert a été très réussite. Cependant, dans d'autres situations, comme les domaines financiers et médicaux, ce type de connaissances n'est pas facilement accessible. L'approche présentée dans ce mémoire résoud ce problème en induisant directement les tailles de modèle MMC des données. Nous incorporons les critères de sélection sous forme d'une fonction objectif dans l'apprentissage pour résoudre les problèmes de sélection de la taille de partition clustering et de la taille modèle MMC pour les clusters individuels dans la partition. Cela mène à la génération d'une meilleure partition de clustering et des modèles plus interprétables pour les données. Une étude réel a été menée sur l'analyse du comportement des diabétiques au cours du temps montre l'efficacité de notre algorithme.

Notre objectif dans ce mémoire est de développer un système dynamique utilisant une méthodologie de modélisation et de clustering à base des MMCs pour les données temporelles.

Nous nous somme basée sur le clustering à base d'un mélange fini d'MMC. Notre approche intègre une procédure pour la sélection de taille de modèle MMC dans le processus de clustering. Il en résulte des modèles de clusters qui sont mieux adaptées pour les données, et une partition de clustering de modèle plus stable et de meilleure qualité. Notre contributions dans ce mémoire pour la modélisation et le clustering des données temporelles et examinées ci-dessous.

Nous avons apporté une contribution au problème de sélection de modèle à l'aide d'une fonction objectif qui fait le lien entre une variable supposé être la taille optimale du modèle et la vraisemblance de l'ensemble de données dont l'objective est de trouver la taille du modèle qui maximise la vraisemblance de l'ensemble de données.

En appliquant le clustering par mélange fini d'MMC où le modèle est composé d'un ensemble de MMCs caractérisant les différents clusters d'objets de données homogènes, nous avons utilisé la fonction objectif pour sélectionner de modèle de partition clustering.

Des travaux antérieurs sur le clustering à l'aide de MMCs supposent que les modèles de clustering dans une partition ont tous la même taille. Notre méthodologie de clustering MMC est beaucoup plus général car elle ne nécessite pas de tailles de modèle uniforme de cluster, et déduit les informations de taille de modèle pour chaque cluster individuel pendant le processus de clustering de données. La sélection de la taille du modèle individuel améliore également la qualité du modèle partition dérivée, surtout quand les modèles sous-jacents de cluster sont de tailles différentes. En outre, les MMCs construits pour les clusters individuelles conviennent mieux pour la tâche de l'interprétation des données.

Nous avons conçue une approche de clustering en termes de quatre boucles imbriquées de la recherche et nous avons analysé sa complexité de calcul. Afin de réduire la complexité de calcul de l'algorithme de base, des procédures de recherche heuristiques basées sur la fonction objectif est employés pour traiter les deux étapes clés de l'algorithme : (i) la dérivation de modèle MMC pour chaque cluster dans la partition et (ii) la dérivation de modèle de partition (le nombre de clusters) clustering. Aussi, nous avons estimé les paramètres de modèle MMC à l'aide de la de l'algorithme Baum Walch. Une procédure itérative de redistribution d'objet a été appliquée pour réduire les itérations de redistribution objet requises pour la convergence du modèle partition. Ces quatre étapes ont été intégrées dans une structure de recherche efficace, qui exploite les caractéristiques de la fonction objectif utilisée pour la sélection de modèle. Les résultats expérimentaux montrent que ces heuristiques sont efficaces pour trouver la structure

optimale de la partition pour les données.

Cette solution a été appliquée aux données médicales qui présentent la variabilité au cours du temps des résultats des analyses des patients diabétiques. L'objectif d'étudier les résultats des analyses des patients diabétiques est d'évaluer et d'améliorer la qualité des soins du diabète en mettant en œuvre des mesures efficaces pour la prévention des complications dangereuses et coûteuses. Notre nouvelle approche été capable de capturer et modéliser les comportements critiques des patients et de dégager automatiquement 4 clusters pertinents. L'un d'eux regroupe les diabétiques à risque de complication ; cela est jugé d'une grande importance par les médecins pour de réagir en temps réel et de leurs éviter d'éventuelles complications.

Ainsi, nous avons mené une étude comparative qui a montré l'efficacité de notre approche par apport à (smyth ; 1999). En effet, la recherche de structure cohérente aux clusters à travers les MMCs prouve leur efficacité interprétée par le taux minime de séquence mal classées face aux autres approches. Ainsi, la recherche de nombre optimal de clusters avec la fonction objectif a prouvé sa convergence rapide.

6 Perspectives de recherche

Nous dégageons quelques pistes de recherche notamment les hypothèses de travail qui restent à reconsidérer ci-dessous :

6.1 Le problème de séquence de données de taille inégale

Dans ce mémoire, toutes les séquences de données temporelles sont supposées avoir la même longueur. Les séquences de données temporelles de longueurs inégales affectent directement le processus de sélection de premières séquences représentant les clusters. Dans notre algorithme actuel, une fois la première séquence est sélectionnée pour un composant (cluster), le reste des composants choisis leurs première séquence en fonction de leurs vraisemblances calculées à l'aide des MMC construites pour les composant précédent. À partir de deux séquence de données, une de longueur plus courte que l'autre, même si une séquence de données ayant une longueur plus semblable au premier objet du composant, la probabilité calculée pour l'objet ayant une longueur plus courte est probablement plus élevé. Ceci est dù au fait que la séquence de probabilité MMC est calculée en fonction de la multiplication des probabilités de points de données individuelles de différents états et les probabilités de transition entre les états. En

conclusion en se retrouve avec des séquences similaires mais pas dans le même cluster.

En conséquence, un cluster est plus susceptible de contenir des séquences de données provenant de différents modèles sous-jacents. Une solution à ce problème est de convertir les données de longueur inégale aux données de longueur égale par des valeurs fixes, par exemple la dernière valeur de chaque séquence. Une autre solution possible consiste à modifier le modèle de calcul de probabilité de la séquence en normalisation fondée sur la longueur de la séquence. Nous aimerions empiriquement étudier les effets de ces deux différentes approches.

6.2 Le problème de l'apprentissage des MMCs

Dans les modèles de Markov caché, l'apprentissage de leurs paramètres dépend des algorithmes utilisés. La vraisemblance maximale obtenue par ces algorithmes n'est donc pas forcément le maximum global [Wu, 1983]. En conséquence, la phase d'apprentissage est très importante car un mauvais apprentissage pourrait conduire à un modèle non adéquat et la convergence de l'algorithme peut devenir très lente [Fraley and Raftery, 1998]. Une solution pour essayer de trouver de bons modèles consiste à intégrer des heuristiques pour mieux explorer l'espace de données.

6.3 Le problème de collection de données

Finalement nous signalons le problème de collection de données réel, on effet notre étude de cas a été mené sur des données européenne vue le manque et la non tolérance de quelque protocole de récupérer les données local notamment pour le secteur médical et même d'autre organisme cela ne motive pas nos travaux de recherche et reste un obstacle pour les travaux futurs. Nous proposons d'avoir des collaborations entre la recherche scientifique et les déférents organismes extérieurs.

Annexe

1 Démonstration de l'algorithme de Baum-Welch

Dans le cas des MMC, on cherche à maximiser $P(V = O/\lambda)$ avec O une séquence de T observation. En appliquant l'algorithme EM à la maximisation de cette probabilité (Bimes,1998), on est amené à maximiser $\Gamma(\lambda, \lambda^{'})$ avec $\lambda = (A, B, \Pi)$ le nouveau modèle et $\lambda^{'}$ le modèle connu (ou actuel) :

$$\Gamma_o(\lambda, \lambda^{'}) = \sum_{Q \in S^T} P(S = Q/V = O, \lambda^{'}) lnP(V = O, S = Q/\lambda)$$

Sachant que :

$$P(V = O, S = Q/\lambda) = \pi_{q_1} \left(\prod_{t=1}^{T-1} a_{q_t q_{t+1}} \right) \prod_{t=1}^{T} b_{q_t}(o_t)$$

La fonction Γ_o se ré-écrit :

$$\begin{aligned} \Gamma_o(\lambda, \lambda^{'}) = & \sum_{Q \in S^T} ln\pi_{q_1} P(S = Q/V = O, \lambda^{'}) \\ & + \sum_{Q \in S^T} \left(\sum_{t=1}^{T-1} lna_{q_t q_{t+1}} \right) P(S = Q/V = O, \lambda^{'}) \\ & + \sum_{Q \in S^T} \left(\sum_{t=1}^{T} lnb_{q_t}(o_t) \right) P(S = Q/V = O, \lambda^{'}) \\ & = \Gamma_o^{\pi}(\lambda, \lambda^{'}) + \Gamma_o^{A}(\lambda, \lambda^{'}) + \Gamma_o^{B}(\lambda, \lambda^{'}) \end{aligned}$$

on peut alors remarquer que $\Gamma(\lambda, \lambda^{'})$ se décompose en somme de trois fonctions de paramétre distincts et indépendants, par conséquent il est possible de les maximiser indépendamement les uns des autres

1.1 Ré-estimation des π_i

On peut dans un premier temps remarquer que le premier coté de l'égalité n'impose que le premier état caché donc on a :

$$\begin{aligned}\Gamma_o^{\pi}(\lambda,\lambda^{'}) &= \sum_{Q\in S^T} \ln\pi_{q_1} P(S=Q/V=O,\lambda^{'}) \\ &= \sum_{i=1}^{N} \ln\pi_i P(S_1=s_i/V=O,\lambda^{'})\end{aligned}$$

en utilisant les multilicateurs de lagrange pour contraindre $\sum_{i=1}^{N}\pi_i = 1$ et en dérivant, on obtient (Bilmes,1998)

$$\begin{aligned}\frac{\partial}{\partial\pi_i}\left(\Gamma_o^{\pi}(\lambda,\lambda^{'})+\gamma\left(\sum_{i=1}^{N}\pi_i-1\right)\right) &= \frac{P(S_1=s_i/V=O,\lambda^{'})}{\pi_i}+\gamma \\ &= 0\end{aligned}$$

et

$$\begin{aligned}\frac{\partial}{\partial\gamma}\left(\Gamma_o^{\pi}(\lambda,\lambda^{'})+\gamma\left(\sum_{i=1}^{N}\pi_i-1\right)\right) &= \sum_{i=1}^{N}\pi_i-1 \\ &= 0\end{aligned}$$

en multipliant la premiére équation pas π_i et en sommant sur *i=1..N*, on obtient grâce à la deuxième équation $\gamma=-1$ et donc (Bilmes,1998)

$$\pi_i = P(S_1=s_i/O,\lambda^{'})$$

1.2 Ré-estimation des a_{ij}

Tout comme précédemment, il est possible de simplifier l'expression car :

$$\begin{aligned}\Gamma_o^{A}(\lambda,\lambda^{'}) &= \sum_{Q\in S^T}\left(\sum_{t=1}^{T-1}\ln\, a_{q_t,q_{t+1}} P(S=Q/V=O,\lambda^{'})\right. \\ &= \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{t=1}^{T-1}\ln\, a_{ij}\; P(S_t=s_i,S_{t+1}=s_j/V=O,\lambda^{'})\end{aligned}$$

car la premiére formule revient à fixer les étatx cachés en t et *t+1.*
En utilisant les multiplicateurs de Lagrange pour contraindre $\sum_{i=1}^{N} a_{ij}$ et en dérivant,on obtient (Bilmes,1998) :

$$\begin{aligned}\frac{\partial}{\partial a_{ij}}\left(\Gamma_o^A(\lambda,\lambda^{'})+\sum_{i=1}^{N}\gamma_n\left(\sum_{k=1}^{N}a_{n,k}-1\right)\right)\\ =\sum_{t=1}^{T-1}\frac{P(S_t=s_i,S_{t+1}=s_j/V=O,\lambda^{'})}{a_{ij}}+\gamma_i\\ =0\end{aligned}$$

et

$$\begin{aligned}\frac{\partial}{\partial \gamma_i}\left(\Gamma_o^A(\lambda,\lambda^{'})+\sum_{i=1}^{N}\gamma_n\left(\sum_{k=1}^{N}a_{n,k}-1\right)\right)=\sum_{k=1}^{N}a_{i,k}-1\\ =0\end{aligned}$$

En multipliant par a_{ij} et en sommant sur $j$la premiére équation, on obtient grâce à la deuxième équation :

$$\gamma_i=-\sum_{t=1}^{T-1}P(S_t=s_i/V=O,\lambda^{'})$$

et donc

$$a_{i,j}=\frac{\sum_{t=1}^{T-1}P(S_t=s_i,S_{t+1}=s_j/V=O,\lambda^{'})}{P(S_t=s_i/V=O,\lambda^{'})}$$

1.3 Ré-estimation $b_i(j)$

Pour des raisons similaire aux précédentes, il est possible de simplifier l'expression(Bilmes,1998) :

$$\begin{aligned}\Gamma_o^B(\lambda,\lambda^{'})&=\sum_{Q\in S^T}\left(\sum_{t=1}^{T}lnb_{q_t}(o_t)\right)P(S=Q/V=O,\lambda^{'})\\ &=\sum_{i=1}^{N}\sum_{t=1}^{T}\ln\, b_i(o_t)P(S=Q/V=O,\lambda^{'})\end{aligned}$$

En utilisant les multiplicateur de lagrange pour contraindre $\sum_{j=1}^{M}b_i(j)=1$ et en dérivant, on obtient :

$$\begin{aligned}&\frac{\partial}{\partial b_i(j)}\left(\Gamma_o^B(\lambda,\lambda^{'})+\left(\sum_{j=1}^{M}b_i(j)-1\right)\right)\\ &=\frac{P(S_t=s_i/V=O,\lambda^{'})\delta(o_t-1)}{b_i(j)}+\gamma_i\\ &=0\end{aligned}$$

et

$$\frac{\partial}{\partial \gamma_i}\left(\Gamma_o^B(\lambda,\lambda^{'})+\sum_{n=1}^{N}\gamma_n\left(\sum_{j=1}^{N}b_n(j)-1\right)\right)=\sum_{j=1}^{M}b_i(j)=0\ 1$$

la fonction $\delta(p)$ vaut 1 si le prédiat est vrai et 0 sinon. On remarque alors que $\sum_{j=1}^{M}\delta(o_t=j)=1$, car un et un seul symbole o_t peut se réaliser à un temps donnée.
Alors en multipliant par $b_i(j)$ la premiére équation et en sommant sur j, on obtient, grâce à la deuxième équation :

$$\gamma_i=-\sum_{t=1}^{T}P(q_t=i/V=O,\lambda^{'})$$

et donc

$$b_i(j)=\frac{\sum_{t=1}^{T}P(S_t=s_i/V=O,\lambda^{'})\delta(o_t=j)}{\sum_{t=1}^{T}P(S_t=s_i/V=O,\lambda^{'})}$$

1.4 Synthèse

On obtient :

$$\pi_i=P(S_1=s_i/O,\lambda^{'})$$

$$a_{i,j}=\frac{\sum_{t=1}^{T-1}P(S_t=s_i,S_{t+1}=s_j/V=O,\lambda^{'})}{\sum_{t=1}^{T-1}P(S_t=s_i/V=O,\lambda^{'})}$$

$$b_i(j)=\frac{\sum_{t=1}^{T-1}P(S_t=s_i/V=O,\lambda^{'})\delta(o_t=j)}{\sum_{t=1}^{T-1}P(S_t=s_i/V=O,\lambda^{'})}$$

Références bibliographiques

[Agrawal 93] R. Agrawal, C. Faloutsos et A. Swami. Efficient similarity search in sequence databases. Proc. of the Int. Conf. on Foundations of Data Organization and Algorithms, Chicago, Etats-Unis, octobre 1993.

[Aussem 95] A.Aussem. Théorie et application des réseaux de neurones récurrents et dynamiques à la prédiction, à la modélisation et au adaptatif des processus dynamiques. Thèse, université Réné Descartes- Paris V,1995.

[Antonilo et al 98] Antonello Panuccio, Manuele Bicego, and Vittorio Murino. A Hidden Markov Model-based approach to sequential data clustering, University of Verona, Italy, 1998.

[Amini 01] Amini, M-R. Apprentissage automatique et recherche de l'information : application à l'extraction d'information de surface et au résumé de texte. PhD thesis, université Paris 6, 2001.

[Aycard 97] O. Aycard, F. Charpillet, D. Fohr et J.-F. Mari. Place Learning and Recognition Using Hidden Markov Models. In Proceedings IEEE-RSJ on International Conference on Intelligent Robots and Systems, pages 1741 - 1746, Grenoble, France, Septembre 1997.

[Alan 88] R.Alan. processus stochastiques avec application aux phénomènes d'attente et de fiabilité, presses polytechniques romandes, Lausanne, suisse, 1988.

[Amaury 03] L.Amaury. Analyse et prédiction de séries temporelles par méthode non linéaires, thèse de doctorat en sciences appliquées, université de Louvain, octobre 2003.

[Berndt 94] D. Berndt et J. Clifford. Using dynamic time warping to find patterns in time series. Proc of KDD workshop, Seattle, juillet 1994.

[Bernhard et al 02] Bernhard Knab, Alexander Schliep, Barthel Steckemetz, and Bernd Wichern. Model-based clustering with Hidden Markov Models and its application to financial time-series data, Bayer AG, D-51368 Leverkusen, Germany. 2002.

[Biswas et al 95] G.Biswas, J.Weinberg, and C.Li . A conceptual clustering method for knowledge discovery in databases. In Artificial Intelligence in Petroleum Industry : Symbolic and Computational Applications, B.Braunschweig and R. Day, Eds. Teditions Technip, 1995.

[Benmiloud 95] B. Benmiloud et W. Pieczynski. Estimation des paramètres dans les chaînes de Markov cachés et segmentation d'images. Traitement du signal, 12(5) :433 - 454, 1995.

[Box et Jenkins 76] Box, G.E.P. et Jenkins, G. M. Time Series Analysis : Frcasting and control. Holden-day, San Francisco,1976.

[Berndt 96] D. J. Berndt. Finding Patterns in Time Series . In U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth et R. Uthurusamy, éditeurs, Advances in Knowledge Discovery and Data Mining, pages 229 - 248. AAAI Press / The MIT Press, 1996.

[Bilmes 98] J.Bilmes. A gentle tutorial of the EM algorithm and its application to parameter estimation for gaussian mixture and hidden Markov models. Technical report. University

of Berkeley,ICSI-TR-97-021,1998.

[Baum and Eagon 67] L.Baum, E.Eagon, J. A. An inequality with applications to statistical estimation for probabilistic function of Markov processes to a model for ecology. Butt American Mathematical Society. 73 :360-363,1967.

[Bize 99] L. Bize, F. Muri, F. Samson, F. Rodolphe, S. Dusko Ehrlich, B. Prum et P. Bessières. Searching Gene Transfers on Bacillus Subtilis Using Hidden Markov Models. In RECOMB'99, 1999.

[Bréhélin 99] L. Bréhélin, O. Gascuel et G. Caraux. Apprentissage de séquences de vecteurs booléens à l'aide de Modèles de Markov Cachés avec Patterns. Application au test de circuits intégrés. In Conférence d'apprentissage, pages 25 - 35, 1999.

[Ding 02] C. Ding, X. He, H. Zha et H. Simon. Adaptive dimension reduction for clustering high dimensional data. Proc. of the IEEE Int. Conf. on Data Mining, ICDM'02, Maebashi City, décembre 2002.

[Cheeseman et al 96] P.Cheeseman, J. Stutz. Bayesian classiffication (autoclass) : Theory and results. In Advances in Knowledge Discovery and Data Mining, U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Eds. Cambridge, MA : MIT press, pp. 153-1801996.

[Chan 99] K. Chan et A.W. Fu. Efficient time series matching by wavelets. Proc. of the IEEE Int. Conf. on Data Engineering,Sydney, mars 1999.

[Dempster et al 1977] A.Dempster, M.Rubin. Maximum-liklihood from incomplete data via the EM algorithm. Journal of royal Statistical Society B, 39(1) : 1-39,1977.

[Dorffner 96] G.Dorffner. Neural networks for time series processing. Neural Network World, 6(4) : 447-468,1996.

[Dumont 09] Fouille de dynamiques multivariées, application à des données temporelles en cardiologie.Phd thesis, Université De Rennes 1,2009.

[Dermatas et al 96] E.Dermatas, G.Kokkinakis. Algorithm for clustering continuous density hmm by recognition error. IEEE Transactions on Speech and Audio Processing 4,3, 231-234 May 1996.

[Faloutsos 94] C. Faloutsos, M. Ranganathan et Y. Manolopoulos. Fast subsequence matching in time-series databases. Proc. of the ACM SIGMOD Int. Conf. on Management of Data, Minneapolis, mai 1994.

[Fisher 87] D.Fisher. Knowledge acquisition via incremental conceptual clustering. Machine Learning 2 , 139-172,1987.

[Fu 01] C.Fu, L.Chung, V.Ng et R.Luk. Pattern discovery from stock time series using self-organizing maps. Proc. of the ACM International Conference on Knowledge Discovery and Data Mining, SIGKDD'01, San Francisco, août 2001.

[Goutte et al 1999] .Goutte,P.Toft et E.Rostrup. On clustering time series. Neuroimage, 9(3) : 289-310, 1999.

[Ghahramani 97] Z.Ghahramani, and M.Jordan. Factorial hidden markov models. Machine Leaning 29, 245-273, 1997.

[Gavrila 95] M. Gavrila et L.Davis. Towards 3-D model-based tracking and recognition of human movement : a multi-view approach. Proc. of the Int. Work. On Automatic Face and Gesture Recognition, FG'95, Zurich, juin 1995.

[Holmes et al 00] W.Holmes and J. Bruno. Finding regulatory elements using joint likelihoods for sequence and expression profile data. Proc Int Conf Intell Syst Mol Biol, 8 :202-10, 2000.

[HERVIEU 09] A.HERVIEU. Analyse de trajectoires vidéos à l'aide de modélisations markoviennes pour l'interprétation de contenus, Phd thesis, l'Université de Rennes 1, 2009

[Jelinek 76] F.Jelinek. Continuous Speech Recognition by Statistical Methods. IEEE Trans. on Acoutics, Speech and Signal Processing, 64(4) :532 - 556, April 1976.

[Juang and Rabiner 90] . B.Juang, H.Rabiner. The segmental k-means algorithm for estimating parameters of hidden Markov models. IEEE transactions on acoustics, speech and signal processing, 38(9) : 1639-1641, 1990.

[Jain 88] A.Jain, K.Dubes. Algorithms for clustering data. Prentice Hall, 1988.

[Keogh 02] E.J. Keogh. Exact indexing of dynamic time warping. In Proc. of Int. Conf. on Very Large Databases,VLDB'02, Hong Kong, août 2002.

[Keogh 01] E.J. Keogh et M.J. Pazzani. Derivative dynamic time warping. In Proc. Of SIAM Int. Conf. on Data Mining, SDM'01), Chicago, avril 2001.

[Korn 97] F. Korn, H. Jagadish et C. Faloutsos. Efficiently supporting ad hoc queries in large datasets of time sequences. Proc. of the ACM SIGMOD Int. Conf. on Management of Data, Tucson, mai 1997.

[Keogh 00] E.J. Keogh et M.J. Pazzani. Scaling up dynamic time warping for data minig applications. In Proc. of ACM int. conf. on Knowledge Discovery and Data Mining, SIGKDD'00, Boston, août 2000.

[Kohonen 97] T. Kohonen. Self-organizing maps. Springer-Verlag, Information Sciences Series, 1997.

[Kosaka et al 95] T.Kosaka, S.Masunaga and M.Kuraoka, Speaker-independent phone modeling based on speaker-dependent hmm's composition and clustering. In Proceedings of the Twentieth International Conference on Acoustics, Speech, and Signal Processing (1995), pp. 441-444.

[kakisawa et al 98] Y.Kakisawa et N.Tanigusshi, . Descrimination and clustering for multivariate time series, Amer. Stat. Assoc., 38(441) :328-340, 1998.

[Lee 90] K.Lee. Context-dependent phonetic hidden markov models for speakerindependent continuous speech recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing 38, 4 (1990), 599-609.

[Lin 04] J. Lin, M. Vlachos, E. Keogh et D. Gunopulos. Iterative incremental clustering of time series. Proc. of the Conf. on Extending Database Technology, EDBT'04, Crete, mars 2004.

[Liao 05] W.Liao. Clustering of time series data - a survey. Pattern Recognition,38 :1857-1874, 2005.

[Li 00] C.Li. A Bayesian Approach To Temporal Clustering Using The Hidden Markov Model Methodology. Thèse. Faculty of the Graduate School of Vanderbilt University,2000.

[Lecolinet 90] E.Lecolinet. Segmentation d'images de mots manuscrits : Application à la lecture de chaînes de caractères majuscules alphanumériques et à la lecture de l'écriture cursive. PhD thesis, Univ. Paris VI, 1990.

[Mari 97] J.-F. Mari et A.Napoli. Modèles stochastiques pour la classification de signaux temporels. In Actes des cinquièmes rencontres de la société francophone de classification, pages 51 - 54, Lyon, France, Septembre 1997.

[Moon 66] . Moon, T. K. (1966). The Expectation Maximisation algorithme, IEEE signal processing magasine, pages 47-60.

[Mannila et al 97] Mannila, H., Toivonen, H., et I., V. A. (1997). Amathematical theory of communication. Data Mining and Knowledge Discovery, 259-249.

[Mallat 99] S. Mallat. Towards ontology based cognitive vision. Academic Press, 1999.

[Myers 81] C. S. Myers et L. R. Rabiner. A comparative study of several dynamic time-warping algorithms for connected word recognition. The Bell System Technical Journal, 60(7) :1389-1409, septembre 1981.

[Mury 97] F. Mury. Comparaison d'algorithmes d'identification de chaînes de Markov cachées et application à la détection de régions homogènes dans les séquences d'ADN. Thèse de doctorat, Université René Descartes, Paris V, 1997.

[Ostendorf 97] M.Ostendorf and H.Singer, Hmm topology design using maximum likelihood successive state splitting. Computer Speech and Language 11, 17-41, 1997.

[Popivanov 02] I. Popavinov et R. J. Miller. Similarity search over time series data using wavelets. Proc. of the IEEE Int. Conf. on Data Engineering, ICDE'02, San Jose, février-mars 2002.

[Piccolo 90] D.Piccolo. A distance measure for classifying ARMA models, Time ser. Anal., 11(2) : 153-163, 1990.

[Rabiner 89] L.Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. In proceeding of the IEEE, volume 77, page 257-286,1989.

[Rabiner 78] L. Rabiner et R. Schafer. Digital processing of speech signals. Prentice-Hall, Signal Processing Series, 1978.

[Rabiner 93] L. Rabiner and B. Juang. Fundamentals of speech recognition. Prentice Hall Signal Processing Series, 1993.

[Schweppe 67] F.Schweppe, On the bhattacharyya distance and divergence between gaussian processes. Information and Control 11, 4 , 373-395, 1967.

[Stolcke 94] A.Stolcke and S.Omohundro. Best-first model merging for hidden markov model induction. Tech. Rep. TR-94-003, International Computer Science Institute, 1947 Center St. Berkeley, CA, Jan. 1994.

[Smyth 97] P.Smyth, Clustering sequences with hidden markov models. In Advances in Neural Information Processing, M. C. Mozer, M. I. Jordan, and T. Petsche, Eds. Cambridge :MA, MIT Press, pp. 648-654, 1997.

[Takami et al 92] J. Takami, S.Sagayama. A successive state splitting algorithm for eÆcient allophone modeling. In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing 1 , pp.573-576,1992.

[VanWijk 99] J.J. Van Wijk, E.R. Van Selow. Cluster and calendar based visualization of time series data. Proc. of the IEEE Symposium on Information Visualization, infovis'99, San Francisco, octobre 1999.

[Vlachos 03b] M.Vlachos, J.Lin, E.Keogh et D.Gunopulos. A wavelet-based anytime algorithm for k-means clustering of time series. Proc. of the SIAM Int. Conf. on Data Mining, San Francisco, mai 2003.

[Viterbi 67] A.Viterbi, A, J. Error bounds for convolutionnal cade and asymptotically optimum decoding algorithm. IEEE transaction on information theory, 13 :260-269,1967.

[Yi 98] B. Yi, H. Jagadish et C. Faloutsos. Efficient retrieval of similar time sequences under time warping. Proc of the IEEE Int. Conf. on Data Mining, ICDE'98, Orlando, février 1998.

[Wu 00] Y. Wu, D. Agrawal et A. El Abbadi. A comparison of DFT and DWT based similarity search in time-series databases. ACM Int. Conf. on Information and Knowledge Management, CIKM'00, McLean, novembre 2000.

[Wilpon 85] J. G. Wilpon et L. R. Rabiner. Modified k-means clustering algorithm for use in isolated word recognition. IEEE Transactions on Acoustics, Speech and Signal Processing, 33(3) :587 :594, 1985.

[Weigend 94] Weigend, A.S.and N.A. Gershenfled, Times Series Prediction : Forcasting the future and Understanding the Past. Addition-Wesley Publishing Company, 1994.

Résumé : L'extraction des connaissances a prouvé son sucée dans différents domaines d'applications comme le domaine médical, spatial, économique, etc. En raison de la disponibilité de quantités énormes de données, ceci devient un problème critique pour extraire des connaissances de manière automatique. Le traitement de ces grandes masses de données a été principalement orienté sur la classification ou le clustering. Un des problèmes qui n'est pas des moindres est le traitement de données avec des dépendances temporelles et spatiales. Cependant, la plupart des techniques d'exploration et d'analyse ont tendance à traiter les données temporelles comme une collection non ordonnée d'événements, ignorant ainsi la dimension temporelle de ces données. De plus, peu de théories et de méthodes générales d'analyse et de construction de modèles pour le traitement de données temporelles sont connues.

Les modèles de Markov cachés (HMMs) constituent une des meilleures techniques pour l'étude de telles données. Les HMMs sont basés sur les théories de probabilités et statistiques. Leur principal avantage est l'existence d'algorithmes d'apprentissage non supervisé, comme Forward, Backpropagation, Viterbi, et Baum Walch, qui permettent d'estimer les paramètres du modèle de l'ensemble de données d'observations et du modèle initial. Nous proposons une méthodologie de clustering pour l'analyse des séquences temporelles à base de modèles de Markov cachés (HMM), utilisant le critère de vraisemblance pour définir une fonction objective dans le but de déterminer le nombre optimal de clusters ainsi que leurs structures cohérentes. L'algorithme se résume en quatre étapes ; i) la recherche du nombre optimal de clusters, ii) la recherche de structure cohérente de chaque cluster, iii) distribution d'objet aux clusters, iv) configuration des paramètres de chaque cluster. Cette étude est validée par des résultats expérimentaux montrant l'efficacité de notre méthodologie.

Mots clés : *Données temporelles, HMM, Clustering, Viterbi, la vraisemblance.*

Abstract : Extraction of knowledge has proven its success in different domains such as medical and space applications, economics, etc. Due to the availability of large quantities of data, this becomes a critical issue for extracting knowledge automatically. Moreover most of these datasets tend to contain temporal and/or spatial dimensions, and until now they are treated as unordered collections of events, ignoring the temporal aspect of the data.

Hidden Markov models (HMMs) represent one of the best techniques for studying such data. HMMs are based on probabilities and statistics. Their main advantage is the existence of non-supervised learning algorithms such as feed-forward and Back-propagation, allowing a proper estimation of the parameters of all observations and building models of the original data. We propose a methodology for clustering temporal sequences based on hidden Markov Model (HMM). We use likelihood criteria to define an objective function and determine the optimal number of clusters as well as their coherent structures. The algorithm consists of four steps ; 1) search for the optimal number of clusters, 2) find coherent structure of each cluster 3) distribute the objects over the clusters, 4) configure settings for each cluster. This study is validated by experimental datasets demonstrating the efficiency of our methodology.

Keywords : *temporal data, HMM, Clustering.*