Cadre général et algorithmes de constructions pour des représentations symboliques adaptatives de séries temporelles

Bernard Hugueney[1]

Université PARIS-DAUPHINE
LAMSADE
Place du Maréchal de Lattre de Tassigny
75775 PARIS CEDEX 16
bernard.hugueney@lamsade.dauphine.fr,
http ://www.lamsade.dauphine.fr/~hugueney

Résumé Les séries temporelles constituent un domaine très important de la fouille de données. En effet, les très gros volumes de données numériques généralement entreposés ne se prêtent pas à une analyse directe. Dans un but à la fois de réduction de la dimensionnalité et d'extraction d'information, la fouille de données de séries temporelles donne généralement lieu à un changement de représentation des séries temporelles. Dans un objectif d'intelligibilité de l'information extraite lors du changement de représentation, on peut avoir recours à des représentations symboliques de séries temporelles. Nous proposons un cadre général de représentation de séries temporelles, ainsi que deux représentations particulières (Clustering-Based Symbolic Representations : CBSR et Segmentation-Based Symbolic Representation with Linear models of 0th order : SBSR-L0) s'inscrivant dans ce cadre général.

Keywords : fouille de données, séries temporelles, changements de représentations, représentations symboliques, recherche de motifs récurrents.

1 Introduction

Les séries temporelles constituent un domaine très actif de la fouille de données. En effet, les bases de données de séries temporelles sont caractérisées non seulement par leur très grand volume, mais aussi par le fait que les informations recherchées (tendances, corrélations,...) ne sont pas directement accessibles à partir des données brutes. Pour cette raison, des changements de représentation doivent être effectués. Nous nous intéressons plus particulièrement à des représentations symboliques, plutôt que numériques, car elles sont intelligibles par les utilisateurs. Nous présentons tout d'abord un cadre général permettant de formuler une très large gamme de représentations symboliques, notamment SAX (Symbolic Aggregate approXimation) qui est une représentation symbolique de séries temporelles classiquement utilisée. Nous proposons deux nouvelles représentations symboliques qui peuvent être considérées comme des extensions adaptatives de SAX : CBSR (Clustering-Based Symbolic Representations) et SBSR-L0 (Segmentation-Based Symbolic Representation with Linear models of 0th order). Pour chacune de ces représentations symboliques, nous proposons un algorithme de construction. En conclusion, nous suggérons

d'autres représentations symboliques à partir du même cadre général, ainsi que diverses applications possibles de ces représentations symboliques.

Il n'est plus rare d'avoir des bases de données constituées de centaines de séries temporelles, elles-mêmes constituées de dizaines de milliers de points. En effet, les séries temporelles sont parmi les données qu'il est le plus facile d'accumuler en très grande quantité car il suffit de disposer de capteurs numériques pour remplir les bases de données, à une vitesse qui n'est limitée que par la fréquence d'échantillonnage des capteurs. On peut donc très facilement se retrouver dans le cas typique de la fouille de données, à savoir que des masses de données ont été recueillies à un niveau de détail sans rapport avec les utilisations qui en seront faites. La figure 1 montre par exemple un extrait de série temporelle issu de la base que nous allons utiliser pour valider notre approche. On y voit un extrait d'une série temporelle sur une période d'une semaine,la base contenant 420 séries temporelles enregistrées pendant un an.

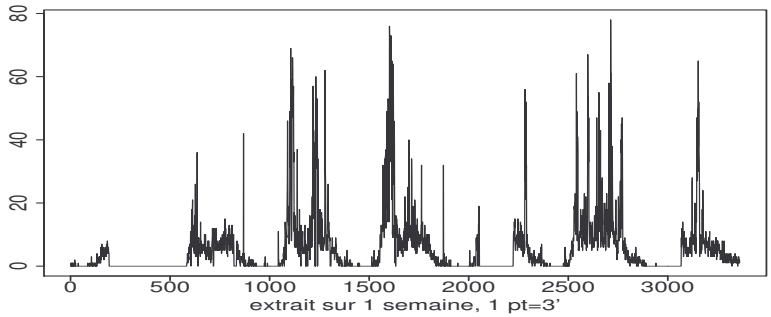

FIG. 1 – Extrait d'une série temporelle de la base.

Les changements de représentation peuvent être effectués pour deux raisons distinctes. Ces raisons ne sont pas exclusives mais induisent des variations sur les critères d'évaluation des représentations.

La motivation première est souvent la simple réduction de dimensionnalité. Le volume des données en jeu dans l'exemple vu plus haut ne permet pas de manipuler directement celles-ci, soit parce qu'elles ne tiennent pas en mémoire centrale, soit que le temps de calcul de n'importe quel algorithme non trivial est prohibitif. On cherche alors à construire des représentations préservant au maximum l'information présente dans les données, mais sans avoir de connaissances sur ce qui constitue justement cette information. On en est alors réduit à utiliser des représentations qui permettent de reconstruire des séries temporelles et à chercher les paramètres de ces représentations de façon à minimiser l'erreur de modélisation, généralement au sens des moindres carrés. Par exemple, la représentation APCA (Adaptive Piecewise Constant Approximation) présentée par [Keogh et al.(2001a)] permet une plus faible erreur quadratique de modélisation (à complexité de représentation égale) que la représentation PAA (Piecewise Aggregate Approximation [Keogh et al.(2001b)] , [Yi and Faloutsos(2000)]). Elle permet aussi une indexation basée sur une distance L^p plus efficace que cette dernière.

La finalité de l'analyse de données étant la présentation d'informations aux utilisateurs, un autre objectif des changements de représentation peut être la construction des éléments d'information de plus haut niveau qui seront pertinents. L'évaluation de telles représentations devra alors prendre en compte une sémantique adaptée à la problématique du domaine.

2 Représentations symboliques de séries temporelles

Les utilisateurs finaux d'outils d'analyse de données réfléchissent à propos de celles-ci en des termes symboliques. Les outils d'analyse de données ayant intérêt à mettre en œuvre un "univers du discours" associé à la sémantique du domaine, cela amène naturellement à considérer des représentations symboliques des données à analyser et non simplement des "résultats numériques". Diverses représentations symboliques de séries temporelles ont déjà été proposées. Nous présentons un cadre général permettant d'unifier celles-ci ainsi que deux représentations symboliques particulières s'incrivant dans ce cadre général : l'une (CBSR) basée sur la classification d'extraits de séries temporelles, l'autre (SBSR-L0) basée sur la segmentation par modèles linéaires d'ordre 0.

2.1 Cadre général pour les représentations symboliques de séries temporelles

Soit une série temporelle ST de N points définie par $\mathrm{ST} = \{(d_i, v_i)\}_{i\in\{1\cdots N\}}$, avec $d_i \in D$ et $v_i \in V$, avec D le domaine de définition temporel et V l'espace numérique des valeurs de la série temporelle, éventuellement muni de l'élément NA indiquant une valeur manquante.

Nous proposons de définir une représentation symbolique de ST par :

- un découpage en P épisodes $E = \{e_j = [d_{j_{\mathrm{debut}}}, d_{j_{\mathrm{fin}}}[\}_{j\in\{1\cdots P\}}$, avec$(d_{j_{\mathrm{debut}}}, d_{j_{\mathrm{fin}}}) \in D^2$ et $d_{j_{\mathrm{debut}}} < d_{j_{\mathrm{fin}}}$,
- un alphabet de K symboles $\Lambda = \{s_m\}_{m\in\{1\cdots K\}}$,
- la représentation symbolique proprement dite RS : $E \to \Lambda, e_j \mapsto \mathrm{RS}(e_j)$ qui est alors l'application associant chacun des épisodes de E à un symbole de Λ.

Cette définition ne spécifie absolument pas la sémantique associée aux éléments de l'alphabet symbolique. Ceci lui permet d'être indépendante des domaines d'application spécifiques. En revanche, afin de permettre une évaluation des représentations symboliques, nous imposons qu'elles permettent de reconstruire une série temporelle numérique sur le domaine de définition temporel D de la série d'origine.

Une représentation symbolique sera pertinente si elle satisfait au mieux les impératifs contradictoires suivants :

- concision maximale de la représentation : avec des cardinalités de E et Λ minimales et des symboles d'interprétation aussi simples que possible
- fidélité maximale : permettant une reconstruction aussi proche que possible de la série d'origine, au sens d'une distance qui peut être liée au domaine d'application. En l'absence de distance propre au domaine, on pourra utiliser une distance point à point et minimiser par exemple la somme des erreurs quadratiques.

Bien sûr l'ensemble des représentations symboliques qu'il est possible de définir est infini et dépend principalement de la sémantique associée à l'alphabet symbolique. Pour une sémantique donnée, l'espace des représentations symboliques possibles d'une série temporelle de N points avec P épisodes associés à des symboles d'un alphabet de cardinalité K est $K^P C_N^P$. Il est donc évident qu'il sera hors de question d'énumérer les solutions possibles. Les représentations symboliques devront donc être choisies de façon à permettre leur construction de façon efficace. Les représentations symboliques que nous proposons ci-après atteignent cet objectif de diverses façons. SAX propose une partition régulière de l'espace temporel et un alphabet indépendants des données, tandis que CBSR et SBSR-L0

se basent respectivement sur des algorithmes de classification et sur des algorithmes de segmentation de séries temporelles.

Dès lors que les représentations des séries temporelles d'une base de données sont adaptées aux données, se pose la question de décider si les paramètres (découpage en épisodes et interprétation des symboles de l'alphabet) doivent être adaptés spécifiquement pour chacune des séries temporelles. En effet, il est aussi possible de choisir une adaptation globale afin de déterminer un seul découpage et un seul ensemble d'interprétations de symboles communs à l'ensemble des séries. Afin de faciliter la comparaison avec SAX, nous présentons ici les résultats correspondants à une adaptation globale pour l'ensemble des séries de la base.

2.2 Représentation symbolique existante : Symbolic Aggregate approXimation (SAX)

Dans un objectif de réduction de la dimensionnalité, il est important de définir des unités temporelles permettant de regrouper les points des séries temporelles. On définit généralement des épisodes comme des intervalles du domaine de définition temporel des séries temporelles à représenter. Très généralement, en raison du coût minime d'acquisition et de stockage des données, les séries temporelles sont enregistrées dans les bases de données sous la forme la plus détaillée possible, indépendamment de l'échelle de temps à laquelle se développent les comportements à identifier. On pourra alors regrouper les points en épisodes sans perdre d'information essentielle. C'est le principe de la représentation symbolique SAX, présentée par [Lin et al.(2003)].

SAX est une représentation symbolique de séries temporelles univariées centrées réduites qui n'est pas adaptative :

- le domaine temporel est divisé en épisodes de même taille
- les classes d'équivalence des valeurs prises par les séries temporelles sont fixées a priori en fonction du nombre de symboles à utiliser, de façon à obtenir un découpage en classes de même effectif sous réserve que la distribution centrée et réduite des valeurs soit normale.

La non adaptativité de la représentation et la référence à la distance euclidienne donnent à SAX les avantages suivants :

- la construction des représentations est extrêmement efficace en temps de calcul ($O(N)$) pour une représentation d'une série temporelle de N points
- toutes les représentations (basées sur un même nombre de symboles et des épisodes de même taille) sont trivialement commensurables.

En contrepartie, cette représentation souffre des inconvénients qui sont intrinsèquement liés :

- l'erreur de modélisation, pour une réduction donnée de la dimensionnalité, n'est pas minimale puisque le modèle n'est pas localement adapté aux données
- les classes d'équivalence auxquelles les symboles sont associés ne sont pas forcément pertinentes car elles ne sont pas adaptées aux données.

Si l'on dispose des ressources nécessaires, il peut donc être intéressant de construire des représentations symboliques adaptatives. De nombreuses représentations symboliques peuvent être envisagées, non seulement en fonction des données à analyser, mais aussi en fonction des tâches d'analyse à effectuer. Dans le cadre générique que nous avons présenté, nous détaillons maintenant deux types de représentation symbolique qui peuvent être

considérés comme des extensions adaptatives des principes qui sous-tendent SAX. L'un est basé sur des classifications automatiques d'extraits de séries temporelles, l'autre sur des segmentations de séries temporelles.

2.3 Représentations symboliques adaptatives basées sur des classifications automatiques d'extraits de séries temporelles

2.3.1 Principes et définitions

SAX effectue des regroupements des points des séries temporelles représentées suivant des épisodes de même taille. Or les séries temporelles que nous étudions sont issues de la mesure de phénomènes en rapport avec des activités humaines et présentent des périodicités liées aux cycles d'activité (journalier et hebdomadaire). Un découpage pertinent en épisodes de même taille de telles séries pourrait tirer parti de ces périodicités.

SAX représente les épisodes par un symbole associé à un intervalle de valeurs. La représentation symbolique obtenue sera d'autant plus fidèle que les points d'un épisode donné appartiennent bien à un même intervalle de valeurs. Une telle interprétation ne permet donc pas de représenter correctement des épisodes d'une journée ou d'une semaine au cours desquels les séries temporelles présentent de trop grandes variations.

Pour cette raison, nous proposons d'associer à ces épisodes de même taille des formes que nous appellerons *formes prototypiques* car chacune sera associée à un ensemble d'extraits de séries temporelles. Ces extraits représentés par un même symbole sont donc conceptuellement réunis au sein d'une classe d'équivalence. Afin de déterminer automatiquement ces classes, nous utilisons un algorithme de classification automatique numérique non-supervisée (par exemple les k-moyennes). Comme dans le cas de la compression vectorielle, il est possible de reconstruire une série temporelle à l'aide des formes prototypiques. Ainsi que vu en 1, il est possible de chercher à minimiser soit l'erreur en reconstruction (dans une optique de réduction de la dimensionnalité), soit l'erreur de classification (dans une optique d'extraction d'information). Afin de permettre une évaluation des résultats sans avoir recours à une interprétation experte sur la pertinence des classes, nous privilégions ici le critère d'erreur en reconstruction.

2.3.2 Algorithme de construction

Pour construire une représentation CBSR, il suffit d'effectuer une classification numérique non-supervisée des extraits de séries temporelles après avoir déterminé les deux paramètres :

- d la durée des épisodes
- $o \in \{1, \cdots, d\}$, indice, que nous appelons *offset*, de début du premier épisode (qui ne correspond donc pas forcément à 1, indice du premier instant de la série temporelle à modéliser).

La durée des épisodes est généralement donnée par des connaissances a priori sur les séries temporelles à modéliser, par exemple une journée ou une semaine. Lorsque cela n'est pas le cas, il est possible d'utiliser la densité spectrale des séries temporelles afin d'estimer des périodicités intéressantes. Deux méthodes sont couramment utilisées à cet effet :

- la transformée de Fourier de la fonction d'auto-corrélation (théorème de Wiener-Kinchine) : cette méthode a une bonne résolution spectrale, mais le calcul de la fonction d'auto-corrélation est assez coûteux en temps de calcul

– le périodogramme moyenné : il faut moyenner le périodogramme car celui-ci représente le spectre instantané. La précision de la localisation des pics de puissance qui nous intéressent est affectée par le lissage, mais le calcul du périodogramme par transformée de Fourier rapide (FFT) est efficace en temps de calcul.

Pour des raisons d'efficacité algorithmique, on utilise généralement le périodogramme moyenné.

Une fois la durée des épisodes déterminée, plusieurs approches (détaillées dans [Hugueney(2003)]) peuvent être utilisées pour déterminer un offset pertinent. Nous avons ici utilisé la plus simple, à savoir celle qui place les points de coupure entre épisodes aux endroits de la série temporelle où les variations locales sont les plus faibles, ceci de façon à essayer d'éviter de couper un comportement (lié à des variations de la série temporelle) entre deux épisodes.

2.3.3 Illustration des résultats

Le contexte de l'activité humaine à laquelle sont liées nos séries temporelles nous a amené à choisir des épisodes d'une journée. La figure 2 montre les formes prototypiques obtenues pour un alphabet de cinq symboles. Bien qu'il soit toujours possible d'approximer aussi fidèlement que nécessaire des séries temporelles quelconques à l'aide de modèles constants par morceaux, les profils observés montrent clairement qu'il faudrait alors un certain nombre d'épisodes pour modéliser correctement une journée. Au contraire, le fait que les séries temporelles de la base de données soient associées à des activités humaines, donc périodiques et répétitives, permet une réprésentation fidèle par CBSR avec un seul épisode par jour.

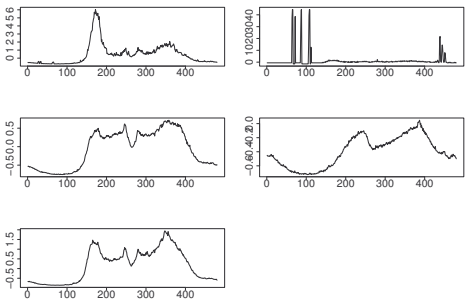

FIG. 2 – Formes prototypiques associées à des épisodes d'une journée.

2.3.4 Évaluation de la modélisation

Le tableau 1 montre les sommes d'erreurs quadratiques des représentations symboliques CBSR comparées à celles des représentations de type SAX pour lesquelles on associerait chaque symbole à la valeur moyenne (sous hypothèse de distribution normale) des valeurs des séries temporelles dans l'intervalle associé au symbole. On constate que le fait que les séries temporelles de la base de données soient associées à des activités humaines, donc périodiques et répétitives, permet une bien meilleure modélisation par CBSR.

nb. de symboles	CBSR	SAX
2	3.48738e+07	8.62192e+09
3	3.09921e+07	8.31765e+09
4	2.89774e+07	8.22382e+09
5	2.74725e+07	7.9138e+09
6	2.66821e+07	7.81554e+09
7	2.60723e+07	7.80194e+09
8	2.5515e+07	7.78745e+09
9	2.50619e+07	7.72299e+09

TAB. 1 – Erreurs de modélisation (SSE) pour des représentations CBSR et SAX.

2.4 Représentations symboliques adaptatives basées sur des segmentations linéaires d'ordre 0 (SBSR-L0)

2.4.1 Principes et définitions

Une représentation symbolique telle que définie en 2.1 est caractérisée par l'interprétation donnée à l'alphabet symbolique. Nous proposons une représentation symbolique de séries temporelles univariées dont les symboles correspondent à des niveaux constants. Les séries temporelles reconstruites à partir de telles représentations symboliques seront donc constantes par morceaux. Alors que dans le cas d'une segmentation classique par modèles linéaires d'ordre 0, chaque épisode est associé à un niveau qui est optimisé localement de façon à représenter au mieux la série sur cet épisode, dans le cas d'une représentation SBSR-L0, chaque niveau est associé à un symbole de l'alphabet dont la cardinalité est $K \ll P$. Chaque niveau représente ainsi un ensemble d'épisodes et nous parlons donc de niveaux *prototypiques*. Pour simplifier, nous utiliserons l'association biunivoque entre les symboles et leur interprétation par un niveau prototypique pour parler d'un alphabet de niveaux prototypiques Λ plutôt que d'un ensemble de niveaux associés aux symboles de l'alphabet. On note DS_Λ l'application qui associe le niveau prototypique à un épisode de D.

Par rapport à SAX, qui est la représentation symbolique la plus proche, les différences sont les suivantes :

- les symboles sont associés à des niveaux et non pas un intervalle. On associe un épisode au symbole correspondant au niveau le plus proche de la moyenne des valeurs prises par la série sur cet épisode plutôt qu'à un symbole correspondant à l'intervalle contenant cette valeur moyenne
- Les niveaux associés aux symboles sont adaptés aux valeurs prises par les séries temporelles à représenter, alors que SAX utilise un partitionnement prédéfini de façon à obtenir une distribution équi-répartie si la distribution des valeurs est normale
- Le découpage en épisodes est adapté aux valeurs prises par les séries temporelles à représenter, alors que SAX utilise un découpage en épisodes de même taille.

Il faut bien sûr décider d'un critère pour adapter l'ensemble des niveaux et le découpage en épisodes. Conformément à l'objectif de fidélité aux données présenté en 2.1, nous cherchons à minimiser la somme des erreurs quadratiques entre les séries à représenter et les reconstructions permises par leur représentation symbolique.

Formellement, la définition de cette représentation symbolique est constituée à par-

tir du cadre général présenté en 2.1 en précisant la définition de l'alphabet de niveaux prototypiques associé à l'alphabet symbolique $\Lambda = \{l_m\}_{m\in\{1\cdots K\}}$ avec $l_m \in \mathbb{R}$.

Puisque le découpage en épisodes et l'alphabet symbolique sont optimisés en fonction des données, deux séries temporelles dont on calcule les représentations symboliques n'auront a priori pas les mêmes ensembles d'épisodes et de niveaux prototypiques. Afin de simplifier les calculs des distances entre représentations symboliques de séries temporelles et d'avoir une comparaison plus immédiate avec SAX, nous avons décidé d'optimiser E et Λ sur l'ensemble des séries temporelles de la base. Cela est rendu possible par le fait que toutes nos séries temporelles sont définies sur le même domaine temporel D et que leurs valeurs numériques sont commensurables. Il serait cependant tout à fait envisageable de calculer des découpages et/ou alphabets de niveaux symboliques propres à chaque série en définissant un calcul de distance adapté, par exemple sur le modèle de celui défini pour une autre représentation adaptative (voir [Keogh and Pazanni(1998)]).

2.4.2 Algorithme de construction

Avec les définitions vues en 2.4.1, l'erreur de reconstruction que nous cherchons à minimiser est $\mathrm{SSE}(\mathrm{ST}, R(\mathrm{RS}(\mathrm{ST}, P, K)))$, avec SSE la somme des erreurs quadratiques, ST la série temporelle à représenter et $R(\mathrm{RS}(\mathrm{ST}, P, K))$ la reconstruction obtenue à partir de la représentation symbolique de ST avec un découpage du domaine temporel en P épisodes et un alphabet de K niveaux prototypiques. Cette erreur de modélisation vaut $\sum_{e_j\in E}\sum_{i\in e_j}\left(v_i - \mathrm{DS}_\Lambda(e_j)\right)^2$ qui exprime que l'erreur à minimiser est la somme des erreurs quadratiques sur tous les épisodes de E et que l'erreur quadratique sur un épisode e_j est calculée entre les valeurs v_i de la série à modéliser et la description symbolique par un niveau prototypique de Λ de cet épisode. Pour minimiser cette erreur, il faut aussi avoir $\mathrm{DS}_\Lambda(e_j) = \mathrm{argmin}_{l\in\Lambda}(\sum_{i\in e_j}\left(v_i - \mathrm{DS}_\Lambda(e_j)\right)^2)$.

Calculer une SBSR-L0 optimale revient donc à trouver les $P-1$ points de rupture entre épisodes qui définissent E et les K niveaux qui définissent Λ de façon à minimiser ce coût.

Pour des raisons évidentes de complexité algorithmique, il n'est pas envisageable de calculer simultanément E et Λ car la complexité algorithmique serait alors rédhibitoire, comme vu en 2.1. Il est néanmoins possible de construire une solution en procédant de façon itérative, en optimisant alternativement E et Λ :

1. Étape initiale : calcul d'un découpage en épisodes E. En l'absence d'un alphabet de niveaux, on effectue une segmentation classique avec modèle linéaire d'ordre 0.
2. Calcul d'un ensemble de K niveaux permettant de minimiser l'erreur en reconstruction si chaque épisode de E est associé à l'un de ces niveaux. Cet ensemble définit l'interprétation de Λ.
3. Calcul d'un ensemble de $P-1$ points délimitant les P épisodes qui définissent E de façon à minimiser l'erreur en reconstruction si chaque épisode est associé à l'un des K niveaux trouvés en 2.
4. Itérer en retournant à l'étape 2 jusqu'à convergence de l'erreur en reconstruction. Cette convergence est garantie par le fait que les étapes 2 et 3 diminuent l'erreur en reconstruction et que celle-ci ne peut être négative.

Cet algorithme ne peut garantir qu'une optimalité locale de la solution trouvée, mais le volume des données à traiter est tel qu'il impose d'énormes contraintes sur la complexité

algorithmique acceptable.

2.4.3 Complexité algorithmique

Il n'est pas possible de donner une expression exacte de la complexité algorithmique totale car le nombre d'itérations jusqu'à convergence dépend des données. Néanmoins, il est possible de calculer les complexités algorithmiques des différentes étapes.

L'étape initiale est une segmentation classique par modèles linéaires d'ordre 0. L'algorithme optimal de segmentation classique par modèles linéaires d'ordre 0 qui peut être lui aussi considéré comme une partition sous contrainte d'ordre total (voir [Lechevallier(1990)]) pour laquelle il est possible d'utiliser un algorithme de programmation dynamique permettant de trouver la solution optimale en $O(\mathrm{PN}^2)$([Fisher(1958)]).

L'étape de calcul des niveaux peut être considérée comme une étape de classification des épisodes. Les épisodes étant regroupés selon une seule dimension (liée au niveau moyen de la série sur l'épisode), il est donc aussi possible d'utiliser un algorithme de classification sous contrainte d'ordre total. La classification optimale des P épisodes en K classes de niveaux est donc réalisable en $O(\mathrm{KP}^2)$.

L'étape 3 est algorithmiquement très proche de l'étape 1 puisque la seule différence est qu'il faut trouver le niveau optimal parmi les K éléments de Λ plutôt que de prendre la valeur moyenne. La complexité de l'algorithme optimal est alors de $O(\mathrm{KPN}^2)$.

Dans un contexte de fouille de données comme celui de notre application, les ordres de grandeur sont les suivants :

- N, le nombre de points des séries temporelle est de l'ordre de 10^4 ou 10^5
- P, le nombre d'épisodes extraits de chaque série temporelles est de l'ordre de 10^2 ou 10^3
- K, le nombre de niveaux prototypiques utilisés pour représenter les séries temporelles est de 10.

On est alors obligé d'utiliser des algorithmes plus rapides mais ne garantissant pas l'optimalité de la solution au problème de segmentation. Une revue détaillée de tels algorithmes dépasse le cadre de cet article. Différents compromis entre le temps d'exécution et la qualité de la solution sont présentés en [Hugueney(2003)].

Comme nous l'avons indiqué en 2.1, nous avons choisi de calculer un découpage en épisodes et un alphabet de niveaux prototypiques communs à toutes les séries temporelles de la base. Cela n'a pas d'influence sur le temps d'exécution global car la complexité algorithmique des différentes étapes est seulement modifiée d'un facteur M, avec M le nombre de séries temporelles de la base, et qu'il n'est plus nécessaire d'appliquer l'algorithme qu'une seule fois pour les M séries au lieu d'une fois pour chacune des M séries.

2.4.4 Illustration des résultats

Sur la gauche de la figure 3, nous montrons un extrait de la distribution des valeurs prises par les séries temporelles, centrées et réduites, de notre base de données, ainsi que les intervalles correspondants aux cinq symboles d'un alphabet d'une représentation SAX en traits pleins. Il ne s'agit que d'un extrait de la distribution car il y a dans la base quelques valeurs atypiques supérieures à 20. On constate que l'hypothèse d'une distribution normale qui sous-tend SAX n'est pas respectée. En effet, il n'y a pas de valeurs inférieures à zéro dans notre base, car nous étudions les mesures de quantités positives ou nulles, ce qui

induit un “déséquilibre” de la distribution que l'on retrouve après avoir centré et réduit les séries temporelles. De plus, l'information sur les valeurs atypiques est complètement perdue par la représentation SAX. Sur cette même figure de gauche, les traits en pointillés et mixtes représentent les valeurs prototypiques que l'on peut associer respectivement aux symboles SAX et aux symboles SBSR-L0. Seulement quatre des cinq valeurs prototypiques de SBSR-L0 sont visibles sur la figure car la cinquième valeur, qui correspond aux valeurs atypiques, est située hors de l'extrait de distribution représenté. Sur la figure de droite, on a représenté le découpage en épisodes obtenu par SBSR-L0 sur un extrait d'une série temporelle. On remarque que celui-ci permet effectivement de mieux modéliser la série en question alors même qu'il est issu d'un découpage adapté globalement à l'ensemble de toutes les séries de la base.

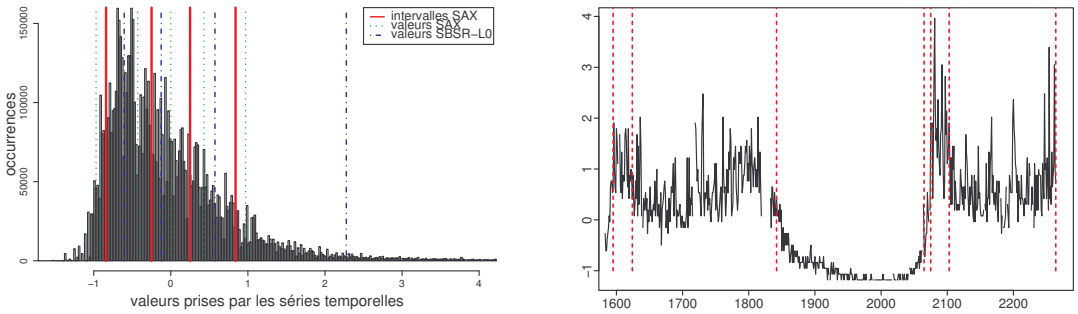

FIG. 3 – Distribution des valeurs des séries temporelles et valeurs prototypiques. Découpage adapté aux données.

2.4.5 Évaluation de la modélisation

Le tableau 2 montre les sommes d'erreurs quadratiques des représentations symboliques SBSR-L0 comparées à celles d'une représentation de type SAX pour laquelle on associerait chaque symbole à la valeur moyenne (sous hypothèse de distribution normale) des valeurs des séries temporelles dans l'intervalle associé au symbole. Le nombre d'épisodes est encore une fois fixé à quatre (en moyenne) par jour.

nb. de symboles	SBSR-L0	SAX
2	2.50095e+09	8.62192e+09
3	2.32939e+09	8.31765e+09
4	2.21192e+09	8.22382e+09
5	2.17804e+09	7.9138e+09
6	2.14144e+09	7.81554e+09
7	2.11467e+09	7.80194e+09
8	2.10549e+09	7.78745e+09
9	2.09249e+09	7.72299e+09

TAB. 2 – Erreurs de modélisation (SSE) pour des représentations SBSR-L0 et SAX.

3 Conclusions et recherches en cours

Au sein du cadre très général que nous avons présenté en 2.1, de nombreuses autres représentations symboliques sont envisageables.

3.1 Découpages et alphabets adaptés à des classes de séries temporelles

Pour permettre une comparaison équitable avec l'existant, nous avons étudié ici le cas de découpages et alphabets de symboles adaptés à l'ensemble des séries temporelles d'une base de données. Il serait bien sûr possible de calculer des découpages et/ou des alphabets adaptés à chacune des séries temporelles. Néanmoins, une telle multiplication des découpages et alphabets peut s'avérer préjudiciable lorsqu'il s'agit de comparer les séries temporelles entre elles. Pour cette raison, on aura intérêt à utiliser lorsque cela est possible des découpages et alphabets communs. A cette fin, on adaptera les découpages et alphabets à des classes homogènes de séries temporelles au sein de la base de données.

3.2 Généralisation à d'autres représentations symboliques basées sur des segmentations

SBSR-L0 est une représentation symbolique basée sur des segmentations par modèles linéaires d'ordre 0. Bien évidemment, le principe est généralisable à tous types de segmentations basés sur d'autres modèles dont on peut classifier automatiquement tout ou partie des paramètres. Nous avons par exemple étudié des représentations symboliques basées sur des segmentations linéaires d'ordre 1 associant les symboles à des classes de pentes.

3.3 Traitements en ligne de flux de données numériques

L'inconvénient majeur des représentations symboliques adaptatives est la nécessité de disposer de l'ensemble des séries temporelles à traiter, ainsi que le temps de calcul nécessaire aux algorithmes de construction de ces représentations. Ceci interdit a priori leur usage pour du traitement en ligne de flux de données. Néanmoins, dans la plupart des cas, il sera possible d'apprendre un alphabet hors-ligne sur un historique des données et d'utiliser celui-ci pour réaliser en-ligne la représentation symbolique d'un flux de données numériques. Il faudra cependant veiller à s'assurer que l'alphabet précalculé reste pertinent pour représenter les nouvelles données et éventuellement procéder aux réactualisations nécessaires.

Références

[Fisher(1958)] Fisher, W. D. (1958). On grouping for maximum homogeneity. *Jasa* (53), 789–798.

[Hugueney(2003)] Hugueney, B. (2003). *"Représentations symboliques de longues séries temporelles"*. Ph. D. thesis, LIP6.

[Keogh et al.(2001a)] Keogh, E., K. Chakrabarti, M. Pazzani, and S. Mehrotra (2001a). Locally adaptive dimensionality reduction for indexing large time series databases. *SIGMOD Record (ACM Special Interest Group on Management of Data) 30*(2), 151–162.

[Keogh and Pazanni(1998)] Keogh, E. and M. J. Pazanni (1998). An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback. In D. Heckerman, H. Mannila, D. Pregibon, and R. Uthurusamy (Eds.), *Proceedings of the Forth International Conference on Knowledge Discovery and Data Mining (KDD-98)*. AAAI Press.

[Keogh et al.(2001b)] Keogh, E. J., K. Chakrabarti, M. J. Pazzani, and S. Mehrotra (2001b). Dimensionality reduction for fast similarity search in large time series databases. *Knowledge and Information Systems 3*(3), 263–286.

[Lechevallier(1990)] Lechevallier, Y. (1990). Recherche d'une partition optimale sous contrainte d'ordre total. Technical report, INRIA.

[Lin et al.(2003)] Lin, J., E. Keogh, S. Lonardi, and B. Chiu (2003). A symbolic representation of time series, with implications for streaming algorithms. In *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pp. 2–11. ACM Press.

[Yi and Faloutsos(2000)] Yi, B.-K. and C. Faloutsos (2000). Fast time sequence indexing for arbitrary L_p norms. In A. El Abbadi, M. L. Brodie, S. Chakravarthy, U. Dayal, N. Kamel, G. Schlageter, and K.-Y. Whang (Eds.), *VLDB 2000, Proceedings of 26th International Conference on Very Large Data Bases, September 10–14, 2000, Cairo, Egypt*, Los Altos, CA 94022, USA, pp. 385–394. Morgan Kaufmann Publishers.