UNIVERSITÉ CAEN NORMANDIE

Université de Caen Normandie

Régression à l'aide du modèle de Markov caché

*Auteurs :*S.Blin A.Bourjal
C.Champarou
M2 Statistiques Appliquées et Analyse Décisionnelle

Tuteur projet : M. F.Chamroukhi

Année universitaire 2018-2019

Table des matières

Modèle

Dans la régression de modèle de Markov cachée (HMMR), chaque série temporelle est représentée par une séquence de variables univariées observées. $(\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_n)$ où l'observation $\boldsymbol{y}$ à l'heure $\boldsymbol{t}$ est supposée être généré par le modèle de régression suivant :

$$\boldsymbol{y}_i = \beta \boldsymbol{t}_i + \sigma \epsilon_i \ ; \ \epsilon_i \sim \mathcal{N}(0,1), \ (i = 1, \ldots, n) \tag{1}$$

Ainsi le modèle précédent peut être réécris sous la forme matricielle suivante :

$$\boldsymbol{y}_i = \mathbf{B} \boldsymbol{t}_i + \boldsymbol{e}_i \ ; \ \boldsymbol{e}_i \sim \mathcal{N}(0, \sigma^2), \ (i = 1, \ldots, n) \tag{2}$$

où $\boldsymbol{y}_i$ est la ith observation de la séries temporelle dans $\mathbb{R}$

1 Paramètre

Le modèle Multiple HMMR est donc entièrement paramétré par le vecteur de paramètre $\boldsymbol{\Psi} = (\pi, \mathbf{A}, \mathbf{B}, \sigma_1^2, \ldots, , \sigma_K^2)$. La sous-section suivante décrit la technique d'estimation des paramètres en maximisant la vraisemblance des données observées au moyen de l'algorithme Expectation-Maximization (EM).

2 Estimation

Soit $\boldsymbol{\Psi}$: le vecteur de paramètre du modèle à estimé.

2.1 Estimation par Maximum de vraisemblance

Le vecteur de paramètre $\boldsymbol{\Psi}$ est estimé en utilisant la méthode bien connue du maximum de vraisemblance grâce à ses propriétés attractives très connues de cohérence, de normalité asymptotique et d'efficacité. En effet, dans nos expériences, un nombre considérable de points de données est acquis au fil du temps, ce qui rend la taille de l'échantillon appropriée pour tirer parti des propriétés limites de l'estimateur du maximum de vraisemblance. Le log-vraisemblance à maximiser dans ce cas est écrit comme suit :

$$\mathcal{L}(\boldsymbol{\Psi}) = \log \boldsymbol{p}(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n; \boldsymbol{\Psi}) \tag{3}$$

$$\mathcal{L}(\boldsymbol{\Psi}) = log \sum_{z_1, \ldots, z_n} \boldsymbol{p}(z_1, \pi) \prod_{i=2}^{n} \boldsymbol{p}(z_i | z_{i-1}; \mathbf{A}) \prod_{t=1}^{n} \boldsymbol{p}(\boldsymbol{y}_i; \boldsymbol{\Psi}). \tag{4}$$

Cette log-vraisemblance est difficile à maximiser directement ,d'ou l'utilisation d'un algorithme performent qui est nommé l' algorithme d' expectation maximisation (EM).

2.2 Estimation par l'ALgorithme EM

L'algorithme espérance-maximisation (en anglais expectation-maximization algorithm, souvent abrégé EM) est un algorithme itératif qui permet de trouver les paramètres du maximum de vraisemblance d'un modèle probabiliste lorsque ce dernier dépend de variables latentes non observables. De nombreuses variantes ont par la suite été proposées, formant une classe entière d'algorithmes.

2.2.1 Etape E

L'étape E calcul une estimation de la log-vraisemblance des données complètes :

$$Q(\boldsymbol{\Psi}, \boldsymbol{\Psi}^{(q)}) = \mathbb{E}\big[\log \boldsymbol{p}(\mathbf{Y}, \boldsymbol{z}|\boldsymbol{t}; \boldsymbol{\Psi})|\mathbf{Y}, \boldsymbol{t}; \boldsymbol{\Psi}\big] \tag{5}$$

Il est facile de montrer que l'etape E n'a besoin, seulement, de calculer les probabilités à posteriori de $\epsilon_{ilk}^{(q)}$ et $\tau_{ik}^{(q)}$. Ces dernier calculer par les récursions du forward-backward.

Forward-Backward

Le processus du forward calcul recursivement les probabilités :

$$\tau_{ik}^{(q)} = \boldsymbol{p}(z_i = \boldsymbol{k}|\mathbf{Y}, \boldsymbol{t}, \boldsymbol{\Psi}^{(q)}) \tag{6}$$

Les probabilités à postériori jointes de l'état $\boldsymbol{k}$ à un temps i et l'état l à l'état $i-1$ donne la sequence d'observations :

$$\epsilon_{ilk}^{(q)} = \boldsymbol{p}(z_i = \boldsymbol{k}, z_{i-1} = l|\mathbf{Y}, \boldsymbol{t}; \boldsymbol{\Psi}^{(q)}) \tag{7}$$

2.2.2 Etape M

Dans cette étape, la valeur du paramètre est mise à jour en calculant le paramètre qui maximise l'attente conditionnelle par rapport à $\boldsymbol{\Psi}$. On peut montrer que cette maximisation conduit aux règles de mise à jour suivantes. Les mises à jour des paramètres gouvernant la chaîne de Markov cachée sont ceux d'un HMM standard et sont donnés :

$$\pi_k^{(q+1)} = \tau_{1k}^{(q)} \tag{8}$$

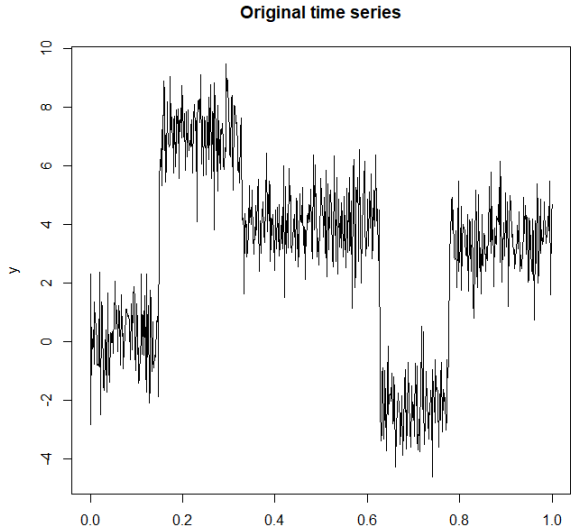
$$\mathbf{A}_{lk}^{(q+1)} = \frac{\sum_{i=2}^{n} \epsilon_{ilk}^{(q)}}{\sum_{i=2}^{n} \tau_{ik}^{(q)}} \tag{9}$$

$$\mathbf{B}_k^{(q+1)} = (\boldsymbol{X}^T \boldsymbol{W}_k^{(q)} \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{W}_k^{(q)} \mathbf{Y} \tag{10}$$

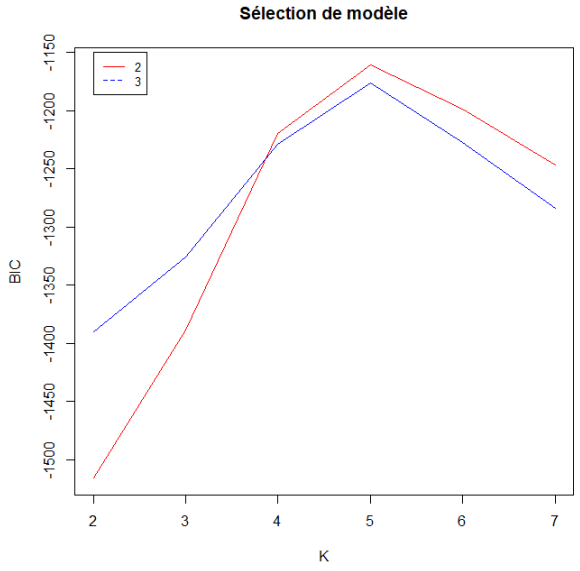
$$\sigma_k^{2(q+1)} = \frac{1}{\sum_{i=1}^{n} \tau_{ik}^{(q)}} (\mathbf{Y} - \mathbf{X}\mathbf{B}_k^{(q+1)})^T \boldsymbol{W}_k^{(q)} (\mathbf{Y} - \mathbf{X}\mathbf{B}_k^{(q+1)}) \tag{11}$$

3 Application

Dans cette section, nous présenterons l'utilisation de la régression à l'aide du modèle de Markov sur des données simulées. Celle-ci sont représenté sur le graphe suivant :

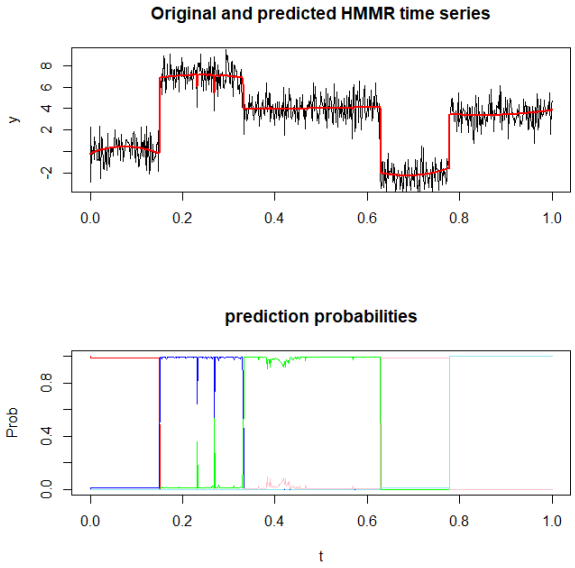


Nous pouvons constater que le graphe peut être séparé en 5 parties disctinctes. Nous supposons donc que cela réprensentera 5 états différents dans l'exemple que nous étudions sur ces données simulées. Afin de confirmer nos conjectures, nous utilisons le Bayesian information criterion (BIC) afin de déterminer le nombre de classes et l'odre de notre regression polynomiale.

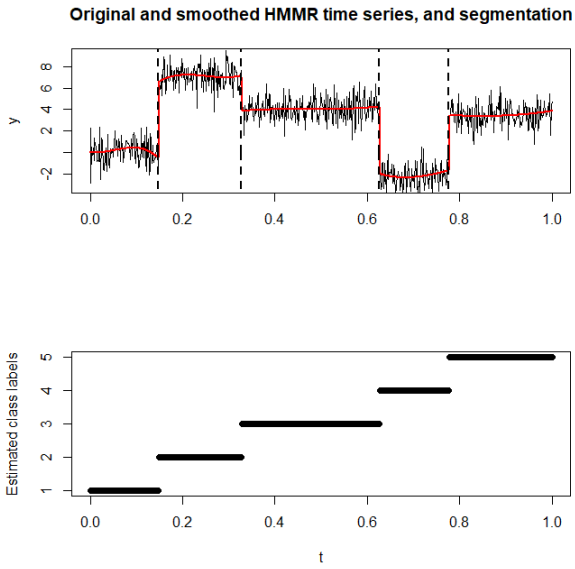


Dans ce graphique, on peut constater qu'il y a un pic en **K**=5 pour les 2 lignes. Ce pic confirme notre hypothèse faite précédemment sur le nombre de classe. On choisira l'ordre de notre regression polynomiale égale à 2 car c'est la valeur la plus élevée pour **K**=5.

Pour la suite de notre étude, on choisit **K**=5, ***p***=2 et l'hétéroscédaticité pour nos données. On utilisera l'algorithme EM pour estimer nos paramètre.



La prédiction nous donne le graphique ci-dessus. Grâce à cela nous avons donc découpé le graphique en fonction du nombre de classe que nous disposions nous donnant le graphique suivant :



Enfin, nous avons estimé les différents segments de tel manière à ce que chaque segment aie une fonction polynomiale qui l'ajuste