Learning of probabilisitc generative models : Sparse Bayesian approaches (and deep architecutres) for functional data and large scale high-dimensioanl data

Faicel Chamroukhi
Maître de Conférences
USTV, LSIS UMR CNRS 6168

email: chamroukhi@univ-tln.fr
web: chamroukhi.univ-tln.fr

Séminaire LSIS - DYNI

Plan

Contexte

- Analyse de courbes (signaux, séquences, fonctions, séries temporelles, ..)
- Données disponibles : mesures issues de capteurs (e.g., puissance, spectre, accélérations, ..)

1. Plusieurs régimes se succédant au sein d'une même courbe (1er aspect temporel) ⇒ Régression dynamique (générative par morceaux)
2. Plusieurs courbes hétérogènes à analyser ⇒ Classification et clustering
3. Prise en compte de l'aspect dynamique entre les courbes (2ème aspect temporel) ⇒ Analyses de séquence de courbes

Objectifs scientifiques

Modélisation statistique de données complexes

- Approches principalement génératives pour régression et classification
- Formalisation probabiliste de la complexité (e.g., aspect dynamique, hétérogénité, aspect séquentiel, parcimonie, ...)

Objectifs scientifiques

Modélisation statistique de données complexes

- Approches principalement génératives pour régression et classification
- Formalisation probabiliste de la complexité (e.g., aspect dynamique, hétérogénité, aspect séquentiel, parcimonie, ...)

Cadre Baysésien parcimonieux

- to control the complexity by integrating priors on the models (another view of regularization of deterministic objective cost functions)
- A main focus on the unsupervised framrwok

Objectifs scientifiques

Modélisation statistique de données complexes

- Approches principalement génératives pour régression et classification
- Formalisation probabiliste de la complexité (e.g., aspect dynamique, hétérogénité, aspect séquentiel, parcimonie, ...)

Cadre Baysésien parcimonieux

- to control the complexity by integrating priors on the models (another view of regularization of deterministic objective cost functions)
- A main focus on the unsupervised framrwok

Représentation non linéaire de haut niveau via des architectures profondes

- Extraction de caractéristiques non linéaires de haut niveau éventuellement parcimonieuses
- Entraînement en ligne

Learning from functional data

Régression à processus logistique latent (RHLP) proposé :

Neural Networks - Elsevier, 22(5-6) :593-602, 2009.

Données temporelles $\mathbf{y} = (y_1, \ldots, y_m)$ observées régulièrement aux instants $\mathbf{t} = (t_1, \ldots, t_m)$

$$y_j = f(t_j) + \sigma \epsilon_j, \quad \epsilon_j \sim \mathcal{N}(0,1)$$

f : polynôme, (B) spline, polynôme par morceaux, polynôme régi par une chaîne de Markov, ...

Définition du modèle

$$y_j = \boldsymbol{\beta}_{z_j}^T \mathbf{t}_j + \sigma_{z_j} \epsilon_j \quad ; \quad \epsilon_j \sim \mathcal{N}(0,1), \quad (j = 1, \ldots, m)$$

- z_j label caché du modèle de y_j
- $\mathbf{z} = (z_1, \ldots, z_n)$ est un processus logistique caché
- Estimation des paramètres du modèle par MV via RHLP

Classification supervisée de courbes

Données : n courbes indépendantes étiquetées $((\mathbf{y}_1, c_1), \ldots, (\mathbf{y}_n, c_n))$

Approche proposée : *(Neurocomputing - Elsevier, 73(7-9) :1210-1221, 2010.)*

- Approche générative dans l'espace initial des données
- Résumer une classe de courbes en une courbe "modèle" (l'espérance)
- Distribution d'une classe homogène de courbes

$$p(\{\mathbf{y}_i\}_{i=1}^n | \mathbf{t}; \boldsymbol{\theta}) = \prod_{i=1}^{n} \underbrace{\prod_{j=1}^{m} \sum_{r=1}^{R} \pi_r(t_j; \mathbf{w}) \mathcal{N}(y_{ij}; \boldsymbol{\beta}_r^T \mathbf{t}_j, \sigma_r^2)}_{p(\mathbf{y}_i | \mathbf{t}; \boldsymbol{\theta})}$$

- Estimation des paramètres par EM similaire au cas d'un signal
- Classification directe dans l'espace des courbes par la règle du MAP :

$$\widehat{c}_i = \arg\max_g p(c_g) p(\mathbf{y}_i | \mathbf{t}; \widehat{\boldsymbol{\theta}}_g)$$

Classification non supervisée de courbes

Données : n courbes indépendantes $(\mathbf{y}_1, \ldots, \mathbf{y}_n)$ observées
classes $(h_1, \ldots, h_n)$ cachées, régimes $(z_{1k}, \ldots, z_{mk})$ cachés de la classe k

Classification non supervisée de courbes

Données : n courbes indépendantes $(\mathbf{y}_1,\ldots,\mathbf{y}_n)$ observées
classes $(h_1,\ldots,h_n)$ cachées, régimes $(z_{1k},\ldots,z_{mk})$ cachés de la classe k

Apprentissage non supervisée pour la classification et la segmentation

(Advances in Data Analysis and Classification (ADAC) 5(4) : 301-321, 2011.)

- Modèle génératif : mélange de modèles RHLP (MixRHLP)

$$p(\mathbf{y}_i|\mathbf{t};\boldsymbol{\Psi}) = \sum_{k=1}^{K} \alpha_k \underbrace{\prod_{j=1}^{m} \sum_{r=1}^{R} \pi_{kr}(t_j;\mathbf{w}_k)\mathcal{N}\left(y_{ij};\boldsymbol{\beta}_{kr}^T\mathbf{t}_j,\sigma_{kr}^2\right)}_{p(\mathbf{y}_i|h_i=k,\mathbf{t};\boldsymbol{\theta}_k)}$$

Classification non supervisée de courbes

Données : n courbes indépendantes $(\mathbf{y}_1,\ldots,\mathbf{y}_n)$ observées
classes $(h_1,\ldots,h_n)$ cachées, régimes $(z_{1k},\ldots,z_{mk})$ cachés de la classe k

Apprentissage non supervisée pour la classification et la segmentation

(Advances in Data Analysis and Classification (ADAC) 5(4) : 301-321, 2011.)

- Modèle génératif : mélange de modèles RHLP (MixRHLP)

$$p(\mathbf{y}_i|\mathbf{t};\boldsymbol{\Psi}) = \sum_{k=1}^{K} \alpha_k \underbrace{\prod_{j=1}^{m}\sum_{r=1}^{R} \pi_{kr}(t_j;\mathbf{w}_k)\mathcal{N}\left(y_{ij};\boldsymbol{\beta}_{kr}^T\mathbf{t}_j,\sigma_{kr}^2\right)}_{p(\mathbf{y}_i|h_i=k,\mathbf{t};\boldsymbol{\theta}_k)}$$

- Vraisemblance :

$$\mathscr{L}(\boldsymbol{\Psi};\mathbf{Y},\mathbf{t}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \alpha_k \prod_{j=1}^{m}\sum_{r=1}^{R} \pi_{kr}(t_j;\mathbf{w}_k)\mathcal{N}\left(y_{ij};\boldsymbol{\beta}_{kr}^T\mathbf{t}_j,\sigma_{kr}^2\right)$$

Classification non supervisée de courbes

Données : n courbes indépendantes $(\mathbf{y}_1,\ldots,\mathbf{y}_n)$ observées
classes $(h_1,\ldots,h_n)$ cachées, régimes $(z_{1k},\ldots,z_{mk})$ cachés de la classe k

Apprentissage non supervisée pour la classification et la segmentation

(Advances in Data Analysis and Classification (ADAC) 5(4) : 301-321, 2011.)

- Modèle génératif : mélange de modèles RHLP (MixRHLP)

$$p(\mathbf{y}_i|\mathbf{t};\boldsymbol{\Psi}) = \sum_{k=1}^{K} \alpha_k \underbrace{\prod_{j=1}^{m}\sum_{r=1}^{R} \pi_{kr}(t_j;\mathbf{w}_k)\mathcal{N}\left(y_{ij};\boldsymbol{\beta}_{kr}^T\mathbf{t}_j,\sigma_{kr}^2\right)}_{p(\mathbf{y}_i|h_i=k,\mathbf{t};\boldsymbol{\theta}_k)}$$

- Vraisemblance :

$$\mathscr{L}(\boldsymbol{\Psi};\mathbf{Y},\mathbf{t}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \alpha_k \prod_{j=1}^{m}\sum_{r=1}^{R} \pi_{kr}(t_j;\mathbf{w}_k)\mathcal{N}\left(y_{ij};\boldsymbol{\beta}_{kr}^T\mathbf{t}_j,\sigma_{kr}^2\right)$$

- Maximisation de la vraisemblance par un algo de type EM (ADAC, 2011)

Algorithme EM

- **Initialisation :** $\mathbf{\Psi}^{(0)}$, $q \longleftarrow 0$ (q itération)

Algorithme EM

- **Initialisation :** $\boldsymbol{\Psi}^{(0)}$, $q \longleftarrow 0$ (q itération)

1. **Étape E : Espérance**

$$
\begin{aligned}
Q(\boldsymbol{\Psi}, \boldsymbol{\Psi}^{(q)}) &= \mathbb{E}\left[\mathscr{L}_c(\boldsymbol{\Psi}; \mathbf{Y}, \mathbf{t}, \mathbf{h}, \mathbf{z}_1, \ldots, \mathbf{z}_K) | \mathbf{Y}, \mathbf{t}; \boldsymbol{\Psi}^{(q)}\right] \\
&= \sum_{k=1}^{K}\sum_{i=1}^{n} \tau_{ik}^{(q)} \log \alpha_g + \sum_{k=1}^{K}\sum_{r=1}^{R_k}\sum_{i=1}^{n}\sum_{j=1}^{m} \tau_{ik}^{(q)} \gamma_{ijkr}^{(q)} \log \pi_{kr}(t_j; \mathbf{w}_k) \\
&+ \sum_{k=1}^{K}\sum_{r=1}^{R_k}\sum_{i=1}^{n}\sum_{j=1}^{m} \tau_{ik}^{(q)} \gamma_{ijkr}^{(q)} \log \mathcal{N}(y_{ij}; \boldsymbol{\beta}_{kr}^T \mathbf{r}_j, \sigma_{kr}^2)
\end{aligned}
$$

$\tau_{ik}^{(q)}$ = $p(h_i = k | \mathbf{y}_i; \boldsymbol{\Psi}^{(q)})$: probabilités a posteriori d'appartenance à la classe k

$\gamma_{ijkr}^{(q)}$ = $p(z_{jk} = r | h_i = g, y_{ij}; \boldsymbol{\Psi}^{(q)})$: probabilités a posteriori d'appartenance aux régimes au sein de la classe k

Algorithme EM

- **Initialisation :** $\boldsymbol{\Psi}^{(0)}$, $q \longleftarrow 0$ (q itération)

1. **Étape E : Espérance**

$$
\begin{aligned}
Q(\boldsymbol{\Psi}, \boldsymbol{\Psi}^{(q)}) &= \mathbb{E}\left[\mathscr{L}_c(\boldsymbol{\Psi}; \mathbf{Y}, \mathbf{t}, \mathbf{h}, \mathbf{z}_1, \ldots, \mathbf{z}_K) | \mathbf{Y}, \mathbf{t}; \boldsymbol{\Psi}^{(q)}\right] \\
&= \sum_{k=1}^{K}\sum_{i=1}^{n} \tau_{ik}^{(q)} \log \alpha_g + \sum_{k=1}^{K}\sum_{r=1}^{R_k}\sum_{i=1}^{n}\sum_{j=1}^{m} \tau_{ik}^{(q)} \gamma_{ijkr}^{(q)} \log \pi_{kr}(t_j; \mathbf{w}_k) \\
&+ \sum_{k=1}^{K}\sum_{r=1}^{R_k}\sum_{i=1}^{n}\sum_{j=1}^{m} \tau_{ik}^{(q)} \gamma_{ijkr}^{(q)} \log \mathcal{N}(y_{ij}; \boldsymbol{\beta}_{kr}^T \boldsymbol{r}_j, \sigma_{kr}^2)
\end{aligned}
$$

$\tau_{ik}^{(q)}$ = $p(h_i = k | \mathbf{y}_i; \boldsymbol{\Psi}^{(q)})$: probabilités a posteriori d'appartenance à la classe k

$\gamma_{ijkr}^{(q)}$ = $p(z_{jk} = r | h_i = g, y_{ij}; \boldsymbol{\Psi}^{(q)})$: probabilités a posteriori d'appartenance aux régimes au sein de la classe k

2. **Étape M : Maximisation** : $\boldsymbol{\Psi}^{(q+1)} = \arg\max_{\boldsymbol{\Psi}} Q(\boldsymbol{\Psi}, \boldsymbol{\Psi}^{(q)})$

Algorithme EM

- **Initialisation :** $\boldsymbol{\Psi}^{(0)}$, $q \longleftarrow 0$ (q itération)

1. **Étape E : Espérance**

$$
\begin{aligned}
Q(\boldsymbol{\Psi}, \boldsymbol{\Psi}^{(q)}) &= \mathbb{E}\left[\mathscr{L}_c(\boldsymbol{\Psi}; \mathbf{Y}, \mathbf{t}, \mathbf{h}, \mathbf{z}_1, \ldots, \mathbf{z}_K) | \mathbf{Y}, \mathbf{t}; \boldsymbol{\Psi}^{(q)}\right] \\
&= \sum_{k=1}^{K}\sum_{i=1}^{n} \tau_{ik}^{(q)} \log \alpha_g + \sum_{k=1}^{K}\sum_{r=1}^{R_k}\sum_{i=1}^{n}\sum_{j=1}^{m} \tau_{ik}^{(q)} \gamma_{ijkr}^{(q)} \log \pi_{kr}(t_j; \mathbf{w}_k) \\
&+ \sum_{k=1}^{K}\sum_{r=1}^{R_k}\sum_{i=1}^{n}\sum_{j=1}^{m} \tau_{ik}^{(q)} \gamma_{ijkr}^{(q)} \log \mathcal{N}(y_{ij}; \boldsymbol{\beta}_{kr}^T \boldsymbol{r}_j, \sigma_{kr}^2)
\end{aligned}
$$

$\tau_{ik}^{(q)}$ = $p(h_i = k | \mathbf{y}_i; \boldsymbol{\Psi}^{(q)})$: probabilités a posteriori d'appartenance à la classe k

$\gamma_{ijkr}^{(q)}$ = $p(z_{jk} = r | h_i = g, y_{ij}; \boldsymbol{\Psi}^{(q)})$: probabilités a posteriori d'appartenance aux régimes au sein de la classe k

2. **Étape M : Maximisation** : $\boldsymbol{\Psi}^{(q+1)} = \arg\max_{\boldsymbol{\Psi}} Q(\boldsymbol{\Psi}, \boldsymbol{\Psi}^{(q)})$

- $q \longleftarrow q+1$

Modèle autorégressif markovien non-homogène

Observations : séquence multidimensionnelle $(\mathbf{y}_1,\ldots,\mathbf{y}_n)$, $\mathbf{y}_t \in \mathbb{R}^d$
Séquence d'états latents : $(z_1,\ldots,z_n)$, $z_t \in \{1,\ldots,K\}$

- Modélisation autorégressive régie par une chaine de Markov non-homogène, *(WCRR 2011)*
- Modèle :

$$\mathbf{y}_t = \mathbf{B}_{z_t}^T \mathbf{y}_{t-1} + \mathbf{e}_t, \quad \mathbf{e}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{z_t}) \quad (t = 2,\ldots,n),$$

  - $\mathbf{B}_{z_t}$ matrice des coefficients d'autorégression
  - $\mathbf{z} = (z_1,\ldots,z_n)$ chaîne de Markov non-homogène
- Probabilités des transitions modélisées par des fonctions logistiques
- Estimation des paramètres par MV via EM

Applications traitées

- Diagnostic et télésurveillance de l'infrastructure ferroviaire :
  - Estimer l'état de fonctionnement du composant (diagnostic)
  - Surveiller son état au cours du temps (suivi temporel)
- Energie des Transports :
  - Estimation de la durée de vie des piles à combustible (PAC).
  - Modélisation de signaux de spectre d'impédance
- Robotique Assistive :
  - Reconnaissance de postures à partir de mesures d'accélération pour l'aide à la personne
  - Segmentation jointe de données temporelles multidimensionnelles
- Structuration et visualisation de séquences par GTM, ...

Perspectives

- Modélisation de données fonctionnelles pour la classification : Functional Mixture Discriminant Analysis (FMDA) : (soumis à ESANN 2012), si ok, à étoffer un peu :
  - e.g., use temporel Gene expression Data, phoneme data in experiments

Perspectives

- Modélisation de données fonctionnelles pour la classification : Functional Mixture Discriminant Analysis (FMDA) : (soumis à ESANN 2012), si ok, à étoffer un peu :
  - e.g., use temporel Gene expression Data, phoneme data in experiments
- Classification likelihood learning for curve clustering and segmentation
  - CEM(-like) algorithm for the mixture of hidden process regression models
  - Advantages :
    - Dedicated to classification rather than estimation (as in ADAC) ;
    - acceleration of the EM learning version
    - (even it provides biased solutions ..)

Piecewise regression mixture for functional data clustering and optimal segmentation

Piecewise regression mixture for functional data clustering and optimal segmentation

- (Hébrail et al. Neurocomputing (2010) : a distance criterion optimized by a K-means-like algorithm :
  $E(\mathbf{z}, \{l_{kr}\}, \{\mu_{kr}\}) = \sum_{k=1}^{K} \sum_{i|z_i=k} \sum_{r=1}^{R_k} \sum_{j \in l_{kr}} (y_{ij} - \mu_{kr})^2$

Piecewise regression mixture for functional data clustering and optimal segmentation

- (Hébrail et al. Neurocomputing (2010) : a distance criterion optimized by a K-means-like algorithm :
  $E(\mathbf{z}, \{l_{kr}\}, \{\mu_{kr}\}) = \sum_{k=1}^{K} \sum_{i|z_i=k} \sum_{r=1}^{R_k} \sum_{j \in l_{kr}} (y_{ij} - \mu_{kr})^2$
- Complete-data log-likelihood for optimized by a CEM algorithm

$$\mathscr{L}_c(\mathbf{z}, \boldsymbol{\Psi}) = \sum_{k=1}^{K} \sum_{i|z_i=k} \log \alpha_k + \sum_{k=1}^{K} \sum_{i|z_i=k} \sum_{r=1}^{R_k} \sum_{j \in l_{kr}} z_{ik} \log \mathscr{N}(y_{ij}; \boldsymbol{\beta}_{kr}^T \mathbf{t}_j, \sigma_{kr}^2)$$

Piecewise regression mixture for functional data clustering and optimal segmentation

- (Hébrail et al. Neurocomputing (2010) : a distance criterion optimized by a K-means-like algorithm :
  $E(\mathbf{z}, \{l_{kr}\}, \{\mu_{kr}\}) = \sum_{k=1}^{K} \sum_{i|z_i=k} \sum_{r=1}^{R_k} \sum_{j \in l_{kr}} (y_{ij} - \mu_{kr})^2$
- Complete-data log-likelihood for optimized by a CEM algorithm

$$\mathscr{L}_c(\mathbf{z}, \boldsymbol{\Psi}) = \sum_{k=1}^{K} \sum_{i|z_i=k} \log \alpha_k + \sum_{k=1}^{K} \sum_{i|z_i=k} \sum_{r=1}^{R_k} \sum_{j \in l_{kr}} z_{ik} \log \mathcal{N}(y_{ij}; \boldsymbol{\beta}_{kr}^T \mathbf{t}_j, \sigma_{kr}^2)$$

Proposition : Optimizing $\mathscr{L}_c(\mathbf{z}, \boldsymbol{\Psi})$ by a CEM algorithm for the piecewise regression mixture model, is equivalent to optimizing $E(\mathbf{z}, \{l_{kr}\}, \{\mu_{kr}\})$ by the K-means-like algorithm of (Hébrail et al. Neurocomputing (2010), if :

- $\alpha_k = \frac{1}{K} \ \forall K$ (identical mixing proportions)
- $\sigma_{kr}^2 = \sigma^2 \ \forall r = 1, \ldots, R_k$ and $\forall k = 1, \ldots, K$ (isotropic model)
- piecewise constant approximation rather than a polynomial

Piecewise regression mixture for functional data clustering and optimal segmentation

- (Hébrail et al. Neurocomputing (2010) : a distance criterion optimized by a K-means-like algorithm :
  $E(\mathbf{z}, \{l_{kr}\}, \{\mu_{kr}\}) = \sum_{k=1}^{K} \sum_{i|z_i=k} \sum_{r=1}^{R_k} \sum_{j \in l_{kr}} (y_{ij} - \mu_{kr})^2$
- Complete-data log-likelihood for optimized by a CEM algorithm

$$\mathscr{L}_c(\mathbf{z}, \boldsymbol{\Psi}) = \sum_{k=1}^{K} \sum_{i|z_i=k} \log \alpha_k + \sum_{k=1}^{K} \sum_{i|z_i=k} \sum_{r=1}^{R_k} \sum_{j \in l_{kr}} z_{ik} \log \mathcal{N}(y_{ij}; \boldsymbol{\beta}_{kr}^T \mathbf{t}_j, \sigma_{kr}^2)$$

Proposition : Optimizing $\mathscr{L}_c(\mathbf{z}, \boldsymbol{\Psi})$ by a CEM algorithm for the piecewise regression mixture model, is equivalent to optimizing $E(\mathbf{z}, \{l_{kr}\}, \{\mu_{kr}\})$ by the K-means-like algorithm of (Hébrail et al. Neurocomputing (2010), if :

- $\alpha_k = \frac{1}{K} \ \forall K$ (identical mixing proportions)
- $\sigma_{kr}^2 = \sigma^2 \ \forall r = 1, \ldots, R_k$ and $\forall k = 1, \ldots, K$ (isotropic model)
- piecewise constant approximation rather than a polynomial

$\Rightarrow$ the proposed CEM algorithm for piecewise polynomial regression mixture is the probabilistic version for hard curve clustering and optimal segmentation of the K-means-like algorithm (Hébrail et al. Neurocomputing (2010))

Bayesian Learning

Bayesian Learning

Why Bayesian ?

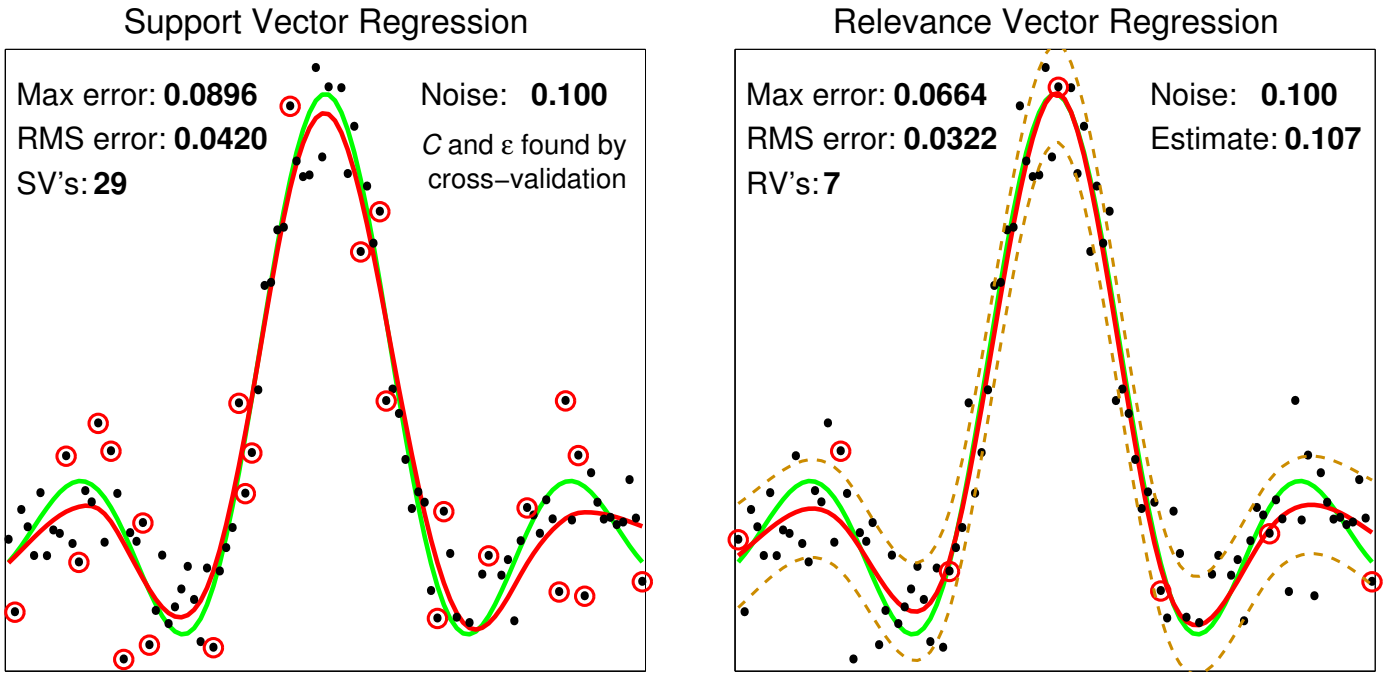
- Bayesian approches allow for explicitly integrating prior knowledge
- A well-established background to deal uncertainty (probabilistic framework)
- soft decision (confidence interval in regression and posterio prob in classification)
- e.g, for generative models : help for understanding the Generative process behind the data
- In many cases cost error functions for deterministic approaches are equivelent to particular cases for MAP criterions of corresponding Bayesian models
- Overcome some numerical problems for ML estimations

Sparse Bayesian Learning : RVM vs SVM (Regression and CLassification)

- The number of relevance vectors in the RVM is significantly smaller than the number of support vectors used by the SVM.
- For a wide range of regression and classification tasks, the RVM gives significantly improvemes the speed of processing on test data.
- Greater sparsity achieved with little or no reduction in generalization error compared with the SVM.
- confidence interval in regression (error bars)
- posterior probabilities in classification (soft classification rather that hard classification)

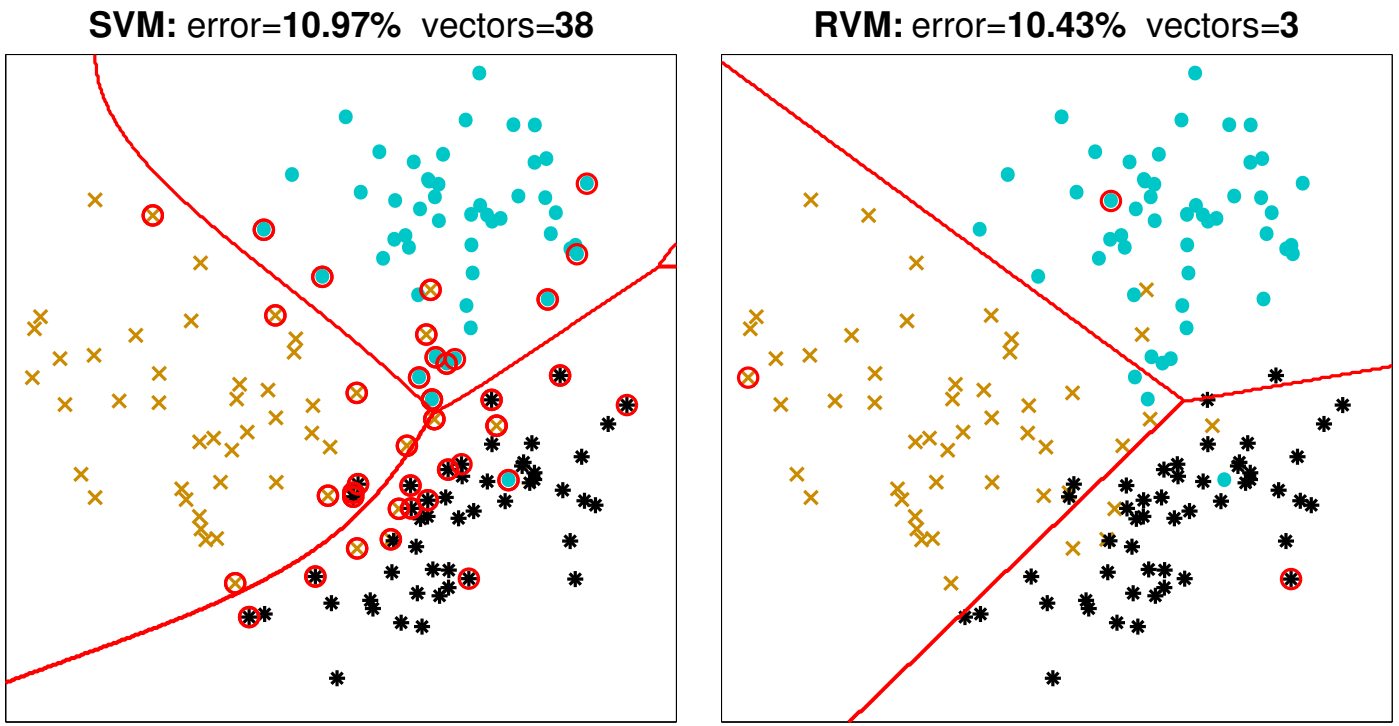
Sparse Bayesian Learning : RVM vs SVM (Regression)

figs from M. Tipping (JMLR 2001)



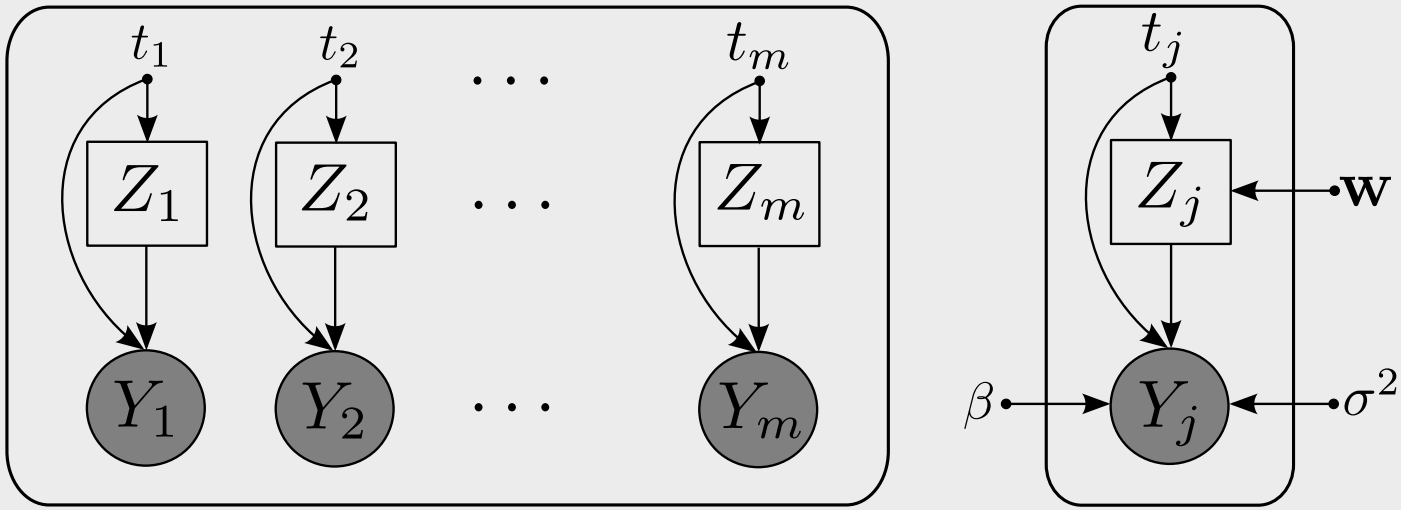
Sparse Bayesian Learning : RVM vs SVM (Classification)

figs from M. Tipping (JMLR 2001)



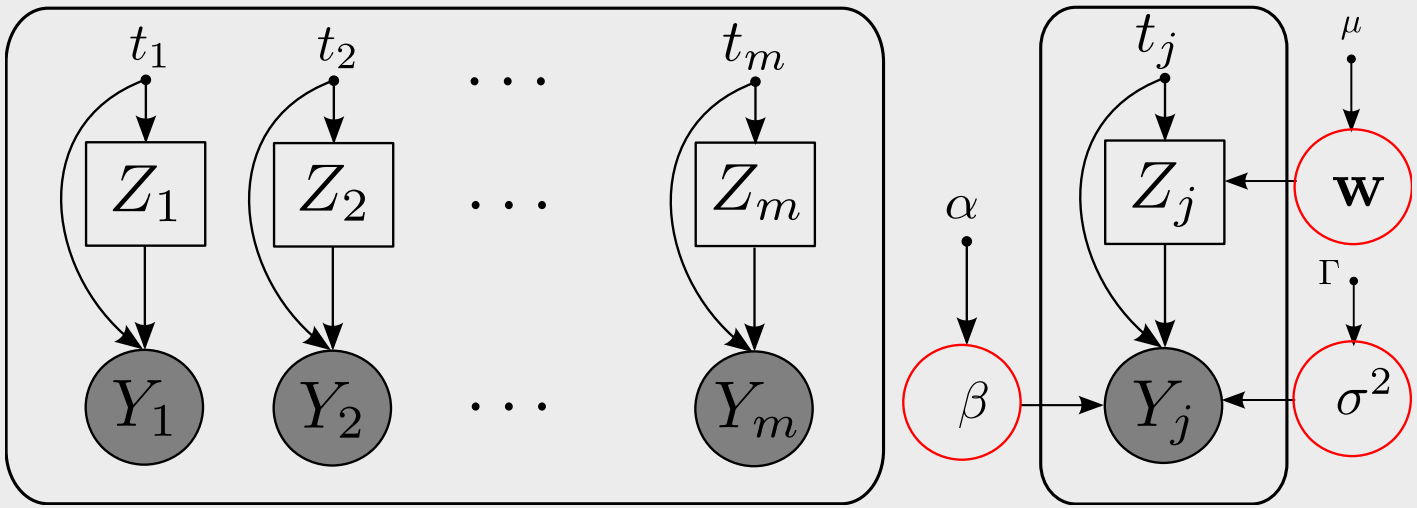
Sparse Bayesian Learning : Hidden Process Regression and Classification

Hidden Process Regression and classification



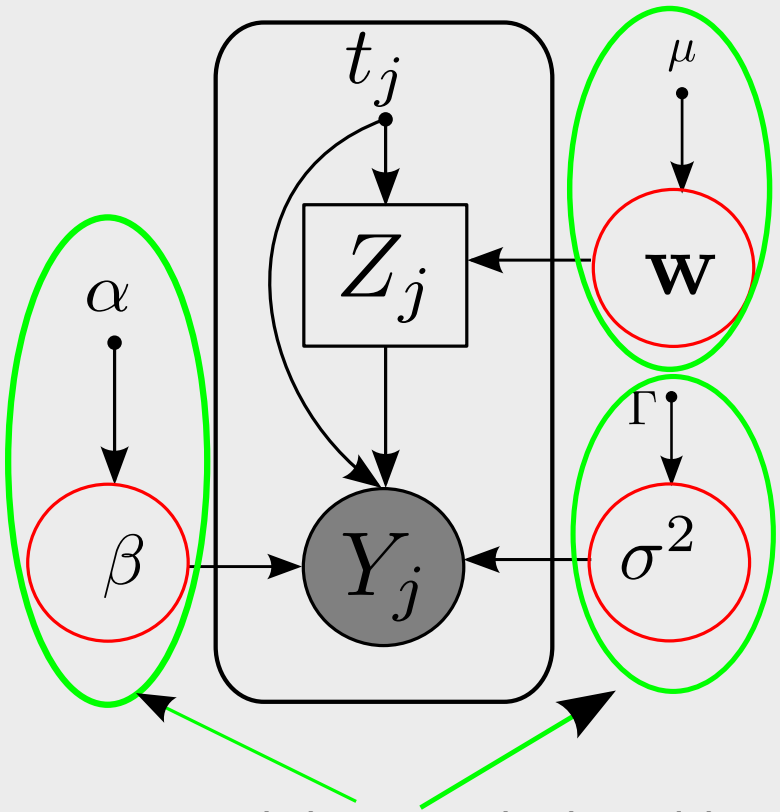
Sparse Bayesian Learning : Hidden Process Regression and Classification

Bayesian Hidden Process Regression and classification



Sparse Bayesian Learning : Hidden Process Regression and Classification

Bayesian Hidden Process Regression and classification



Control the complexity with regard to the classification

Control the complexity with regard to the regression

Sparse Bayesian Learning of Hidden Process Regression and classification

- Sparse Bayesian Learning of Hidden Process regression from curves with regime changes
- Simultaneously performs regression (approximate the true function) and classification (approximate the process generating the true function)
  - Gaussian (or Laplacian) prior on the logistic process (to say that only the data "near" the transitions are important for the segmentation)
  - Gaussian prior on the regression paramters (for each polynomial regime which is equivalent to performing classical bayesian polynomial regression per regime) (and inverse Wishart on the covariance) ..
  - $\Rightarrow$ A Bayesian piecewise polynomial regression framework

Bayesian Predictive Sprase Decomposition

- Fast Inference in Sparse Coding Algorithms with Applications to Object Recognition (Kavukcuoglu et al. CoRR 2010)
- $\mathbf{x} \in \mathbb{R}^n$ to be represented by $\mathbf{z} \in \mathbb{R}^m$ $(m > n)$
- PSD replace the Basis Pursuit Denoising problem

$$E(\mathbf{x}, \mathbf{z}; B) = \frac{1}{2}\|\mathbf{x} - B\mathbf{z}\|_2^2 + \lambda\|\mathbf{z}\|_1$$

by considering a non-linear regressor that maps $\mathbf{x}$ to the representation $\mathbf{z}$ $(\mathbf{z} = f(\mathbf{x}; G, W, D) = G\tanh(WY + D))$:

Cost function :

$$E(\mathbf{x}, \mathbf{z}; B, \boldsymbol{\theta}) = \frac{1}{2}\|\mathbf{x} - B\mathbf{z}\|_2^2 + \lambda\|\mathbf{z}\|_1 + \alpha\|\mathbf{z} - f(\mathbf{x}; \boldsymbol{\theta})\|_2^2$$

- $\boldsymbol{\theta}$ the parameter of a non-linear regressor (e.g., $\boldsymbol{\theta} = (G, W, D)$ for $f(\mathbf{x}; G, W, D) = G\tanh(WY + D)$)

Bayesian Predictive Sprase Decomposition

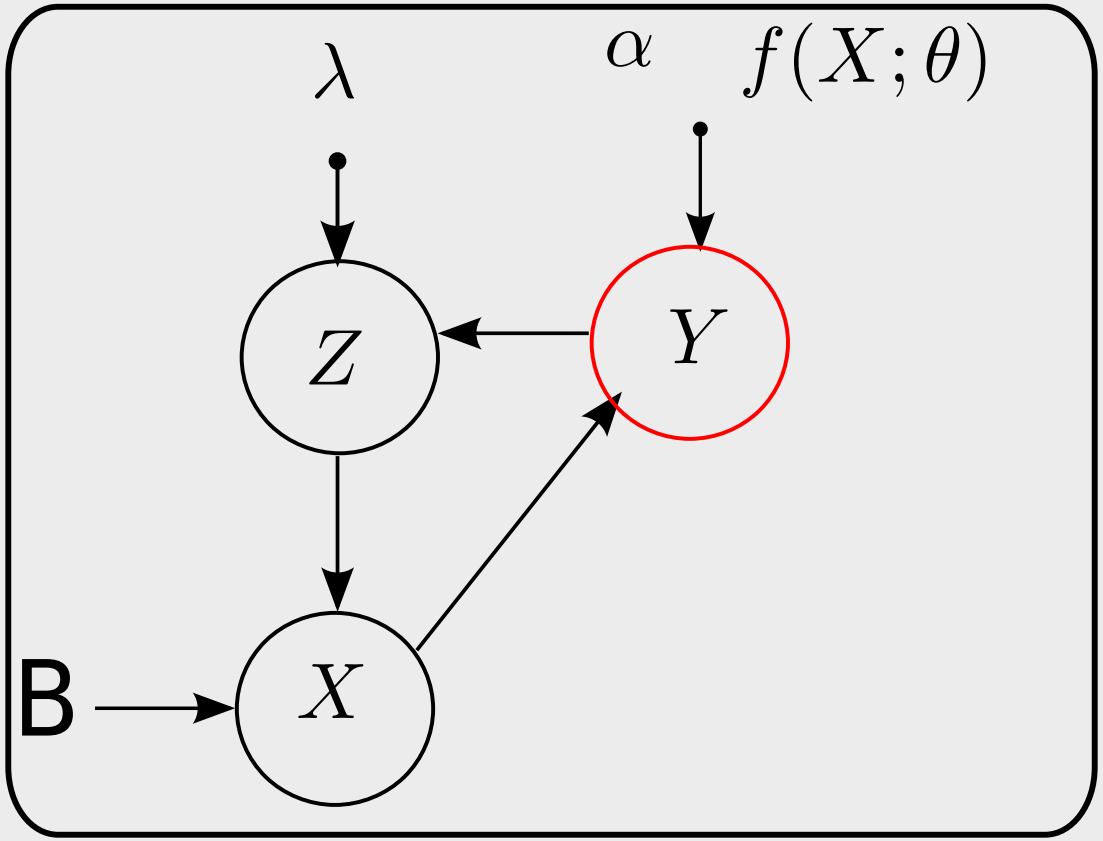
- The Cost function seen before could be equivalent to a MAP criterion of a Bayesian non linear regression model

  MAP criterion :

$$\mathscr{L}(\mathbf{x},\mathbf{z};B,\boldsymbol{\theta}) \propto \underbrace{\frac{1}{2}\|\mathbf{x}-B\mathbf{z}\|_2^2}_{\text{Likelihood term}} + \underbrace{\underbrace{\lambda\|\mathbf{z}\|_1}_{\text{Laplacian Prior}} + \underbrace{\alpha\|\mathbf{z}-f(\mathbf{x};\boldsymbol{\theta})\|_2^2}_{\text{Gaussian Prior centred at f}}}_{\text{Prior term}}$$

- $\Rightarrow$ The model parameters : $(B,\mathbf{z})$ and the hyperpamaraters : $(\alpha,\lambda,\boldsymbol{\theta})$
- tp be optimized under a MAP framwerok

Bayesian Predictive Sprase Decomposition



?? to be continued

Unsupervised Learning of Deep architectures

High level data representation
Combine the power of representation with models for sequential data
Online training (to accelerate the learning process)

Autres

- Structuration de flux de données audiovisuelles
- *Objectif :* Visualisation et classification non supervisée
- Approche topographique
- Mélange hiérarchique topographique de modèles de Markov cachés.

  $\Rightarrow$ *Generative Topographic Mapping* hiérarchique temporel

Merci de votre attention !