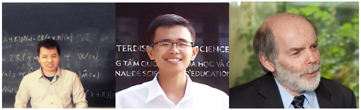
Functional Mixtures-of-Exeperts

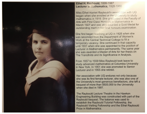
Faicel Chamroukhi

with V-H. Hoang, N-T. Phan, G-J. McLachlan



The University of Queensland, School of Mathematics and Physics

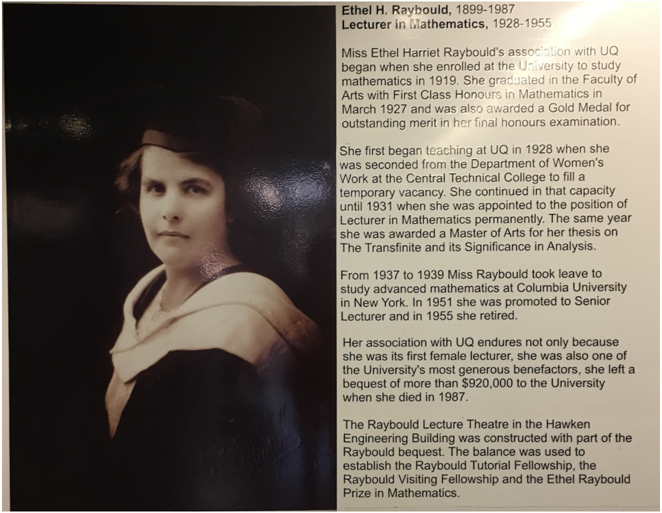
Brisbane, Australia, 20 august 2019



Research supported in part by an **Ethel Raybould Visiting Fellowship**

THE UNIVERSITY OF QUEENSLAND
AUSTRALIA

School of
Mathematics and Physics



* picture taken by my phone from a poster at one of the UQ buildings.

Outline

Scientific context

- The data are assumed to represent samples from random variables with unknown probability distributions
- The area of statistical learning and analysis of complex data.
- **Data :** Complex data $\hookrightarrow$ *heterogeneous, temporal/dynamical, high-dimensional/functional, incomplete,...*
- **Objective :** Transform the data into knowledge :
  $\hookrightarrow$ Reconstruct hidden structure/information, groups/hierarchy of groups, summarizing prototypes, underlying dynamical processes, etc

Modeling framework

- Latent variable models : $f(x|\boldsymbol{\theta}) = \int_z f(x, z|\boldsymbol{\theta})\mathrm{d}z$
  Generative formulation :

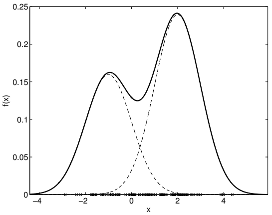
$$\begin{aligned} z &\sim q(z|\boldsymbol{\theta}) \\ x|z &\sim f(x|z, \boldsymbol{\theta}) \end{aligned}$$

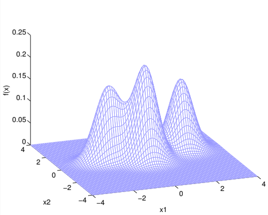
$\hookrightarrow$ Mixture models : $f(x|\boldsymbol{\theta}) = \sum_{k=1}^{K} \mathbb{P}(z = k) f(x|z = k, \boldsymbol{\theta}_k)$ and extensions

Mixture models [McLachlan and Peel., 2000]

Mixture modeling framework

- Mixture density : $f(x|\boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k f_k(x|\boldsymbol{\theta}_k)$





- Hig power for density approximation : [Nguyen et al., 2019] get pdf here

- Generative model

$$
\begin{aligned}
z &\sim \mathcal{M}(1; \pi_1, \ldots, \pi_K) \\
x|z &\sim f(x|\boldsymbol{\theta}_z)
\end{aligned}
$$

$\hookrightarrow$ learn $\boldsymbol{\theta}$ from the data

Mixtures and the EM algorithm

Finite Mixture Models [McLachlan and Peel., 2000]

$f(\boldsymbol{x};\boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k f_k(\boldsymbol{x};\boldsymbol{\theta}_k)$ with $\pi_k > 0 \; \forall k$ and $\sum_{k=1}^{K} \pi_k = 1$.

Maximum-Likelihood Estimation

$$\widehat{\boldsymbol{\theta}} \in \arg\max_{\boldsymbol{\theta}} \log L(\boldsymbol{\theta})$$

log-likelihood : $\log L(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k f_k(\boldsymbol{x}_i;\boldsymbol{\theta}_k)$.

The EM algorithm [Dempster et al., 1977, McLachlan and Krishnan, 2008]

$$\boldsymbol{\theta}^{new} \in \arg\max_{\boldsymbol{\theta}\in\Omega} \mathbb{E}[\log L_c(\boldsymbol{\theta})|\mathcal{D}, \boldsymbol{\theta}^{old}]$$

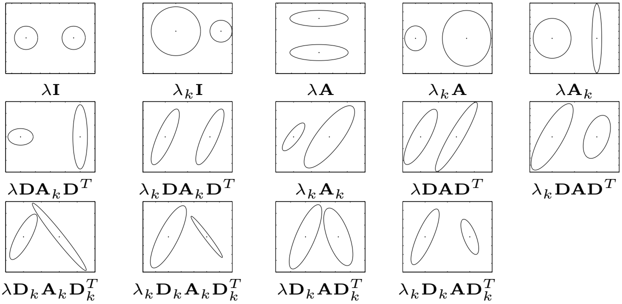
complete log-likelihood : $\log L_c(\boldsymbol{\theta}) = \sum_{i=1}^{n} \sum_{k=1}^{K} Z_{ik} \log\left[\pi_k f_k(\boldsymbol{x}_i;\boldsymbol{\theta}_k)\right]$ where $Z_{ik} = \mathbb{I}(Z_i = k)$

Clustering

$$\widehat{z}_i = \arg\max_{1\leq k\leq K} \mathbb{P}(Z_i = k|\boldsymbol{x}_i;\widehat{\boldsymbol{\theta}}), \quad (i = 1,\ldots,n)$$

Mixtures in a high-dimensional setting

- Parsimonious GMMs [Banfield and Raftery, 1993, Celeux and Govaert, 1995] :
  - Eigenvalue decomposition of the covariance mat. $\mathbf{\Sigma}_k = \lambda_k \mathbf{D}_k \mathbf{A}_k \mathbf{D}_k^T$
  - λ_k the volume of the kth cluster (the amount of space of the cluster).
  - $\mathbf{D}_k = (\mathbf{v}_{k1}, \ldots \mathbf{v}_{kp})$ orthogonal matrix of eigenvectors $\mathbf{v}$ of $\mathbf{\Sigma}_k$: determines the orientation of the cluster.
  - $\mathbf{A}_k = \text{diag}(\lambda_{k1}, \ldots, \lambda_{kp})/|\mathbf{\Sigma}_k|^{1/p}$ a normalized diagonal matrix (its determinant is 1) of the eigenvalues of $\mathbf{\Sigma}_k$ arranged in a decreasing order. This matrix is associated with the shape of the cluster.



Mixtures in a high-dimensional setting

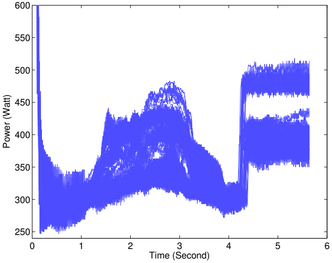
for $p > n$:

- LASSO Regularization : [Pan and Shen, 2007] [Celeux et al., 2019]
- Mixtures of Factor Analyzers [McLachlan et al., 2003] (or MCFA extension)
  $\mathbf{\Sigma}_k = \mathbf{B}_k \mathbf{B}_k^T + \mathbf{\Lambda}_k$:
  $\mathbf{B}_k$ is a $p \times q$ (with $q < p$) matrix and $\mathbf{\Lambda}_k$ is a diagonal matrix.
  $\hookrightarrow \left(\mathbf{B}_k \mathbf{B}_k^T + \mathbf{\Lambda}_k\right)^{-1}$ and $|\mathbf{B}_k \mathbf{B}_k^T + \mathbf{\Lambda}_k|$ are calculated in a q-dimensional space !

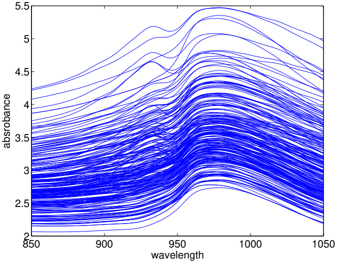
$\hookrightarrow$ Here we consider the case where the data are entire functions : $\{X(t); t \in \mathcal{T}\}$

Outline

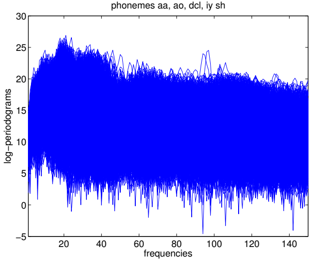
Functional data are increasingly frequent



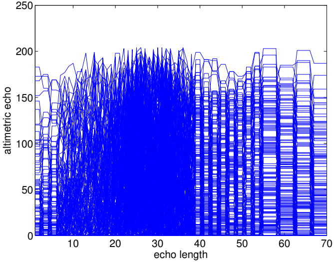
Railway time-series trajectories



Tecator data



Phonemes curves



Satellite waveforms

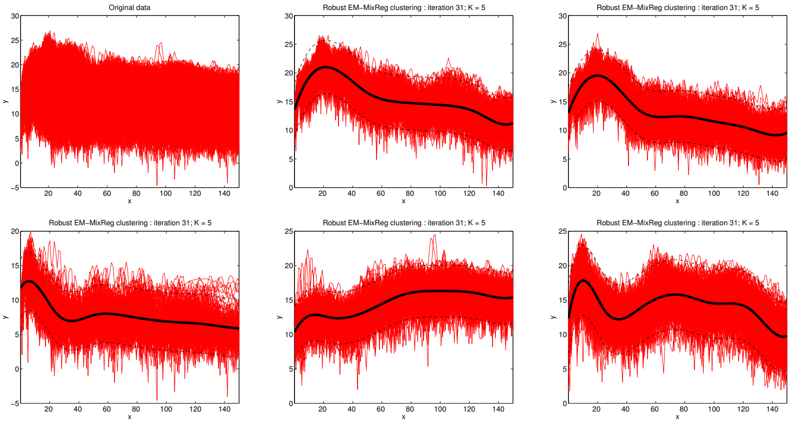
Statistical analysis of functional data

A broad literature :
[James and Hastie, 2001, James and Sugar, 2003]
[Ramsay and Silverman, 2005]
[Ferraty and Vieu, 2006]
[Ramsay et al., 2011]
[Bouveyron and Jacques, 2011]
[Samé et al., 2011]
[Delaigle et al., 2012]
[Jacques and Preda, 2014]
[Bouveyron et al., 2018]
[Qiao et al., 2018]
A review can be found in [Chamroukhi and Nguyen, 2018] pdf available here

- Functional regression
- Functional classification
- Functional clustering, including model-based
- Functional graphical models
- ...

Classification of functional data

Phonemes data set [1] : $n = 1000$ log-periodograms for $m = 150$ frequencies



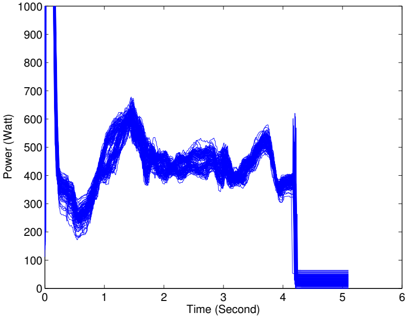
1. Data from `http://www.math.univ-toulouse.fr/staph/npfda/`, used in Ferraty and Vieu [2003]

Clustering of functional data

Clustering real curves of high-speed railway-switch operations

Data : $n = 115$ curves of $m \simeq 510$ observations

$K = 2$ clusters : operating state without/with possible defect

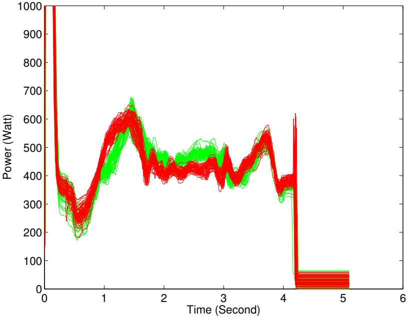


Clustering switch operations

Clustering real curves of high-speed railway-switch operations

Data : $n = 115$ curves of $m \simeq 510$ observations

$K = 2$ clusters : operating state without/with possible defect



Mixture-of-Experts modeling (for vectorial data)

- Data : an observed i.i.d sample of the pair $(\boldsymbol{X}, Y)$ where the response $Y \in \mathbb{R}$ for the vector of predictors $\boldsymbol{X} \in \mathbb{R}^p$ is governed by a hidden categorical variable Z

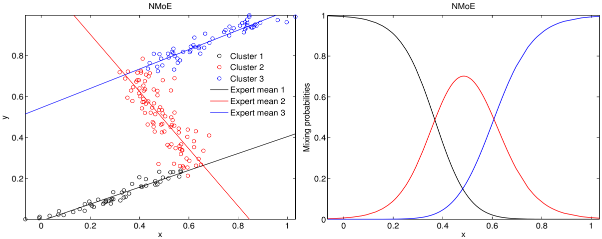
  $z_i \in [K]$ is the expert label for $(\boldsymbol{X}_i, Y_i)$
- Mixture of experts (ME) [Jacobs et al., 1991, Jordan and Jacobs, 1994] :

$$f(y|\boldsymbol{x};\boldsymbol{\Psi}) = \sum_{k=1}^{K} \underbrace{\pi_k(\boldsymbol{x};\mathbf{w})}_{\text{Gating network}} \underbrace{f_k(y|\boldsymbol{x};\boldsymbol{\theta}_k)}_{\text{Expert Network}}$$

- Gating network (e.g softmax) : $\pi_k(\boldsymbol{x};\mathbf{w}) = \frac{\exp(w_{k0}+\boldsymbol{w}_k^T\boldsymbol{x})}{1+\sum_{\ell=1}^{K-1}\exp(w_{\ell 0}+\boldsymbol{w}_\ell^T\boldsymbol{x})}$
- Experts network (e.g Gaussian regressors) : $f_k(y|\boldsymbol{x};\boldsymbol{\theta}_k) = \phi\big(y;\mu(\boldsymbol{x};\boldsymbol{\beta}_k),\sigma_k^2\big)$ with parametric (non-)linear regression functions $\mu(\boldsymbol{x};\boldsymbol{\beta}_k)$
- parameter vector $\boldsymbol{\Psi} = (\mathbf{w}^T, \boldsymbol{\Psi}_1^T, \ldots, \boldsymbol{\Psi}_K^T)^T$

  $\hookrightarrow$ For a review, see Nguyen and Chamroukhi [2018] ▸ pdf available here

Illustration



Fitting the ME model

Maximum Likelihood Estimation via EM [Dempster et al., 1977, Jacobs et al., 1991]

- MLE : $\boldsymbol{\Psi}$ is commonly estimated by maximizing the observed-data log-likelihood : $\widehat{\boldsymbol{\Psi}}_n \in \arg\max_{\boldsymbol{\Psi}\in\Theta} L(\boldsymbol{\Psi})$ with $L(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k(\boldsymbol{x}_i; \boldsymbol{w}) f(\boldsymbol{y}_i|\boldsymbol{x}_i; \boldsymbol{\Psi}_k)$
  $\hookrightarrow$ the EM algorithm

Fitting the ME model

Maximum Likelihood Estimation via EM [Dempster et al., 1977, Jacobs et al., 1991]

- MLE : $\boldsymbol{\Psi}$ is commonly estimated by maximizing the observed-data log-likelihood : $\widehat{\boldsymbol{\Psi}}_n \in \arg\max_{\boldsymbol{\Psi}\in\Theta} L(\boldsymbol{\Psi})$ with $L(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k(\boldsymbol{x}_i; \boldsymbol{w}) f(\boldsymbol{y}_i|\boldsymbol{x}_i; \boldsymbol{\Psi}_k)$
  $\hookrightarrow$ the EM algorithm

$\hookrightarrow$ Consider a high-dimensional setting
$\hookrightarrow$ Looking for sparse models

Regularized MLE of the ME [Khalili, 2010] [Chamroukhi and Huynh, 2019]

$\boldsymbol{\Psi}$ is estimated by maximizing a penalized observed-data log-likelihood :

$$\widehat{\boldsymbol{\Psi}}_n \in \arg\max_{\boldsymbol{\Psi}\in\Theta} L(\boldsymbol{\Psi}) - \mathrm{Pen}_\lambda(\boldsymbol{\Psi})$$

- $\hookrightarrow$ $\mathrm{Pen}_\lambda(\boldsymbol{\Psi})$ LASSO penalties for experts and the gating network
- encourages sparse solutions
- performs parameter estimation and feature selection

$\hookrightarrow$ Doesn't apply (directly) to functional data (e.g functional predictors and/or responses)

Mixtures-of-Experts with functional predictors

- ME to relate functional predictors $\{X(t) \in \mathbb{R}; t \in \mathcal{T} \subset \mathbb{R}\}$ to a scalar response $Y \in \mathcal{Y} \subset \mathbb{R}$
- The inputs $\boldsymbol{X}(\cdot)$ are data continuously recorded from (multiple) subject' sensors for some time period

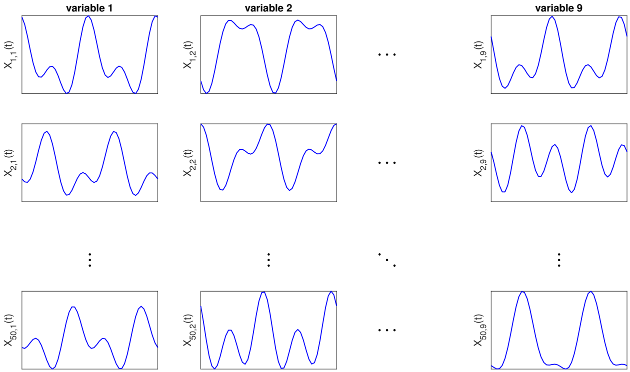


FIGURE – Functional predictors $X_{ij}(t)$ $t \in \mathcal{T}$, $i = 1, \cdots, n$ and $j = 1, \ldots, p$.

Mixtures-of-Experts with functional predictors

- ME to relate functional predictors $\{X(t) \in \mathbb{R}; t \in \mathcal{T} \subset \mathbb{R}\}$ to a scalar response $Y \in \mathcal{Y} \subset \mathbb{R}$
- The inputs $\boldsymbol{X}(\cdot)$ are data continuously recorded from (multiple) subject' sensors for some time period

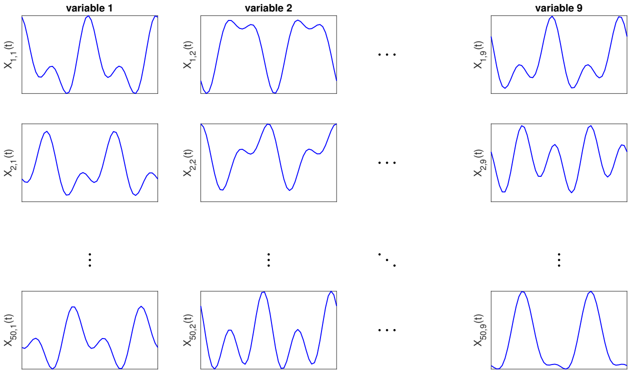


FIGURE – Functional predictors $X_{ij}(t)$ $t \in \mathcal{T}$, $i = 1, \cdots, n$ and $j = 1, \ldots, p$.

$\hookrightarrow$ We first consider univariate functional predictors ($p = 1$)

- Let $\{X_i(\cdot), Y_i\}_{i=1}^n$, be a random i.i.d sample from the pair $\{X(\cdot), Y)\}$

ME for functional predictors and a scalar response

Questioning

Regression, Clustering and classification of observations with functional predictors with three guidelines :

- (1) generative modeling : warranty for estimation and prediction
- (2) deal with high-dimensional setting (sparsity and feature selection)
- (3) User guideline : keep an interpretable fit

Proposed answering

(1) Mixture modeling (Mixture-of-Experts model) (2) regularization to encourage sparse solutions (3) Functional regression, classification and clustering

Main modeling guidelines

- Functional generalized linear models [James, 2002, Müller and Stadtmüller, 2005] (including FLR)
- Functional linear regression (FLR) (anf FGLM) that's interpretable FLiRTI [James et al., 2009]

Stochastic representation of the FunME model

Functional experts network

- The experts are formulated as functional regression models (see eg. James [2002])

$$Y_i = \beta_{z_i,0} + \int_{\mathcal{T}} X_i(t)\beta_{z_i}(t)dt + \varepsilon_i, \quad i = 1, \ldots, n, \tag{1}$$

$z_i \in [K]$ is the unknown expert label for $(X_i(.), Y_i)$

$\beta_{z_i,0} \in \mathbb{R}$ is an unknown intercept coefficient of functional LR z_i

$\{\beta_{z_i}(t) \in \mathbb{R}; t \in \mathcal{T}\}$ is the unknown function of parameters of functional expert z_i

$\varepsilon_i \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2_{z_i})$ with $\sigma^2_{z_i} \in \mathbb{R}^+$ the variance of expert z_i

Stochastic representation of the FunME model

Functional experts network

- The experts are formulated as functional regression models (see eg. James [2002])

$$Y_i = \beta_{z_i,0} + \int_{\mathcal{T}} X_i(t)\beta_{z_i}(t)dt + \varepsilon_i, \quad i = 1, \ldots, n, \tag{1}$$

$z_i \in [K]$ is the unknown expert label for $(X_i(.), Y_i)$

$\beta_{z_i,0} \in \mathbb{R}$ is an unknown intercept coefficient of functional LR z_i

$\{\beta_{z_i}(t) \in \mathbb{R}; t \in \mathcal{T}\}$ is the unknown function of parameters of functional expert z_i

$\varepsilon_i \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2_{z_i})$ with $\sigma^2_{z_i} \in \mathbb{R}^+$ the variance of expert z_i

Functional gating network

- Multinomial logistic (softmax) functional gated network : For $z = 1, \cdots, K-1$:

$$\begin{aligned} h_z(X(t), t \in \mathcal{T}) &= \log\left\{\frac{\mathbb{P}(Z = z|X(t), t \in \mathcal{T})}{\mathbb{P}(Z = K|X(t), t \in \mathcal{T})}\right\} = \alpha_{z,0} + \int_{\mathcal{T}} X(t)\alpha_z(t)dt \\ \mathbb{P}(Z = z|X(t), t \in \mathcal{T}) &= \frac{\exp\left(\alpha_{z,0} + \int_{\mathcal{T}} X(t)\alpha_z(t)dt\right)}{1 + \sum_{z'=1}^{K-1}\exp\left(\alpha_{z',0} + \int_{\mathcal{T}} X(t)\alpha_{z'}(t)dt\right)}, \end{aligned} \tag{2}$$

- $\alpha_{z,0} \in \mathbb{R}$ is an unknown intercept parameter
- $\{\alpha_z(t) \in \mathbb{R}; t \in \mathcal{T}\}$ is the unknown function of parameters of gating network z

Representation of the functional predictors

$$
\begin{aligned}
Y_i &= \beta_{z_i,0} + \int_{\mathcal{T}} X_i(t)\beta_{z_i}(t)dt + \varepsilon_i, \quad i = 1, \ldots, n, \\
h_z(X(t), t \in \mathcal{T}) &= \alpha_{z,0} + \int_{\mathcal{T}} X(t)\alpha_z(t)dt.
\end{aligned}
$$

- Estimating the coefficient functions $\alpha(.)$ and $\beta(.)$ is a high-dimensional problem $\hookrightarrow$ needs approximation for dimensionality reduction
- Two main approaches : i) basis representation ii) functional PCA (FPCA) [Ramsay and Silverman, 2005])

Representation of the functional predictors

$$
\begin{aligned}
Y_i &= \beta_{z_i,0} + \int_{\mathcal{T}} X_i(t)\beta_{z_i}(t)dt + \varepsilon_i, \quad i = 1, \ldots, n, \\
h_z(X(t), t \in \mathcal{T}) &= \alpha_{z,0} + \int_{\mathcal{T}} X(t)\alpha_z(t)dt.
\end{aligned}
$$

- Estimating the coefficient functions $\alpha(.)$ and $\beta(.)$ is a high-dimensional problem $\hookrightarrow$ needs approximation for dimensionality reduction
- Two main approaches : i) basis representation ii) functional PCA (FPCA) [Ramsay and Silverman, 2005])

$\hookrightarrow$ Here we represent the functional data by using a basis expansion :

$$
X_i(t) = \sum_{j=1}^{r} x_{ij} b_j(t) = \boldsymbol{x}_i^\top \mathbf{b}_r(t), \tag{3}
$$

- $\mathbf{b}_r(t) = (b_1(t), b_2(t), \ldots, b_r(t))^\top$ is an r-dimensional basis ((B-)spline, Wavelet,..)
- $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ir})^\top$ can be seen as the vector representation of $X_i(.)$

Representation of the functional predictors

$$
\begin{aligned}
Y_i &= \beta_{z_i,0} + \int_{\mathcal{T}} X_i(t)\beta_{z_i}(t)dt + \varepsilon_i, \quad i = 1,\ldots,n,\\
h_z(X(t), t\in\mathcal{T}) &= \alpha_{z,0} + \int_{\mathcal{T}} X(t)\alpha_z(t)dt.
\end{aligned}
$$

- Estimating the coefficient functions $\alpha(.)$ and $\beta(.)$ is a high-dimensional problem $\hookrightarrow$ needs approximation for dimensionality reduction
- Two main approaches : i) basis representation ii) functional PCA (FPCA) [Ramsay and Silverman, 2005])

$\hookrightarrow$ Here we represent the functional data by using a basis expansion :

$$X_i(t) = \sum_{j=1}^{r} x_{ij} b_j(t) = \boldsymbol{x}_i^\top \mathbf{b}_r(t), \tag{3}$$

- $\mathbf{b}_r(t) = (b_1(t), b_2(t), \ldots, b_r(t))^\top$ is an r-dimensional basis ((B-)spline, Wavelet,..)
- $\boldsymbol{x}_i = (x_{i1},\ldots,x_{ir})^\top$ can be seen as the vector representation of $X_i(.)$

Here the X's are directly observed. We later consider the case when they are not.
$\hookrightarrow$ The x_{ij}s' can be computed explicitly by $x_{ij} = \int_{\mathcal{T}} X_i(t)b_j(t)dt$ for $j = 1,\ldots,r$ and $\boldsymbol{x}_i = (x_{i1},\ldots,x_{ir})^\top$.

Representation of the functional gating network

Functional linear predictor for the gating network defined as :

$$h_z(X(t), t \in \mathcal{T}) = \log\left\{\frac{\mathbb{P}(Z = z|X(t), t \in \mathcal{T})}{\mathbb{P}(Z = K|X(t), t \in \mathcal{T})}\right\} = \alpha_{z,0} + \int_{\mathcal{T}} X(t)\alpha_z(t)dt$$

$\hookrightarrow$ The function $\alpha_z(t)$ is represented similarly as for X function by

$$\alpha_z(t) = \sum_{j=1}^{q} \zeta_{z,j} b_j(t) = \boldsymbol{\zeta}_z^\top \mathbf{b}_q(t) \quad (4)$$

where

- $\mathbf{b}_q(t) = (b_1(t), \ldots, b_q(t))^\top$ is a q-dimensional basis (of the same type as X).
- $\boldsymbol{\zeta}_z = (\xi_{z,1}, \xi_{z,2}, \ldots, \xi_{z,q})^\top$ is the vector of logistic regression coefficients

Representation of the functional gating network

Then the functional linear predictor $h_z(X_i)$ for $i = 1, \ldots, n$ is represented as

$$\begin{aligned} h_z(X_i(t), t \in \mathcal{T}; \boldsymbol{\alpha}) &= \alpha_{z_i,0} + \int_{\mathcal{T}} X_i(t)\alpha_{z_i}(t)dt = \alpha_{z_i,0} + \int_{\mathcal{T}} \boldsymbol{x}_i^\top \mathbf{b}_r(t)\mathbf{b}_q^\top(t)\boldsymbol{\zeta}_{z_i} dt \\ &= \alpha_{z_i,0} + \boldsymbol{x}_i^\top \left(\int_{\mathcal{T}} \mathbf{b}_r(t)\mathbf{b}_q^\top(t)dt \right) \boldsymbol{\zeta}_{z_i} \\ &= \alpha_{z_i,0} + \boldsymbol{\zeta}_{z_i}^\top \mathbf{r}_i, \end{aligned}$$

where

- $\boldsymbol{x}_i = (x_{i,1}, \ldots, x_{i,r})^\top$
- $\mathbf{r}_i = \left(\int_{\mathcal{T}} \mathbf{b}_r(t)\mathbf{b}_q(t)^\top dt\right)^\top \boldsymbol{x}_i$

Representation of the functional gating network

Then the functional linear predictor $h_z(X_i)$ for $i = 1, \ldots, n$ is represented as

$$\begin{aligned} h_z(X_i(t), t \in \mathcal{T}; \boldsymbol{\alpha}) &= \alpha_{z_i,0} + \int_{\mathcal{T}} X_i(t)\alpha_{z_i}(t)dt = \alpha_{z_i,0} + \int_{\mathcal{T}} \boldsymbol{x}_i^\top \mathbf{b}_r(t)\mathbf{b}_q^\top(t)\boldsymbol{\zeta}_{z_i} dt \\ &= \alpha_{z_i,0} + \boldsymbol{x}_i^\top \left(\int_{\mathcal{T}} \mathbf{b}_r(t)\mathbf{b}_q^\top(t)dt \right) \boldsymbol{\zeta}_{z_i} \\ &= \alpha_{z_i,0} + \boldsymbol{\zeta}_{z_i}^\top \mathbf{r}_i, \end{aligned}$$

where

- $\boldsymbol{x}_i = (x_{i,1}, \ldots, x_{i,r})^\top$
- $\mathbf{r}_i = \left(\int_{\mathcal{T}} \mathbf{b}_r(t)\mathbf{b}_q(t)^\top dt\right)^\top \boldsymbol{x}_i$

The FunME gating network (2) is then now phrased as

$$\begin{aligned} h_{z_i}(X_i; \boldsymbol{\xi}) &= \alpha_{z_i,0} + \boldsymbol{\zeta}_{z_i}^\top \mathbf{r}_i \\ \pi_k(\mathbf{r}_i; \boldsymbol{\xi}) &= \frac{\exp\{\alpha_{k,0} + \boldsymbol{\zeta}_k^\top \mathbf{r}_i\}}{1 + \sum_{k'=1}^{K-1} \exp\{\alpha_{k',0} + \boldsymbol{\zeta}_{k'}^\top \mathbf{r}_i\}} \end{aligned} \quad (5)$$

where $\boldsymbol{\xi} = ((\alpha_{1,0}, \boldsymbol{\zeta}_1^\top), \ldots, (\alpha_{K-1,0}, \boldsymbol{\zeta}_{K-1}^\top))^\top \in \mathbb{R}^{(K-1)\times(q+1)}$ is the unknown parameter vector of the gating network, to be estimated.

Representation of the functional experts

$$Y_i = \beta_{z_i,0} + \int_{\mathcal{T}} X_i(t)\beta_{z_i}(t)dt + \varepsilon_i, \quad i = 1, \ldots, n\cdot$$

- The coeffecient function $\beta_z(\cdot)$ is represented by the following expansion :

$$\beta_z(t) = \sum_{j=1}^{p} \eta_{z,j} b_j(t) + e(t) = \boldsymbol{\eta}_z^\top \mathbf{b}_p(t) + e(t) \tag{6}$$

- $\mathbf{b}_p(t) = (b_1(t), b_2(t), \ldots, b_p(t))^\top$ is a p-dimensional basis ((B-)spline, Wavelet,..)
- $\boldsymbol{\eta}_z = (\eta_{z,1}, \eta_{z,2}, \ldots, \eta_{z,p})^\top$ is the vector of regression coefficients
- $e(t) \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_e^2)$, $e(\cdot) \perp X_i$'s and represents the approximation error of $\beta_z(t)$ by linear projection $\mathbf{b}_p(t)^\top \boldsymbol{\eta}_z$.

Representation of the functional experts

The functional linear expert regressor z is then represented as :

$$
\begin{aligned}
Y_i &= \beta_{z_i,0} + \int_{\mathcal{T}} X_i(t)\beta_{z_i}(t)dt + \varepsilon_i = \beta_{z_i,0} + \int_{\mathcal{T}} \boldsymbol{x}_i^\top \mathbf{b}_r(t)\left(\mathbf{b}_p^\top(t)\boldsymbol{\eta}_{z_i} + e_i(t)\right)dt + \varepsilon_i \\
&= \beta_{z_i,0} + \boldsymbol{x}_i^\top \left(\int_{\mathcal{T}} \mathbf{b}_r(t)\mathbf{b}_p^\top(t)dt\right)\boldsymbol{\eta}_{z_i} + \int_{\mathcal{T}} X_i(t)e(t)dt + \varepsilon_i \\
&= \beta_{z_i,0} + \boldsymbol{\eta}_{z_i}^\top \mathbf{x}_i + \varepsilon_i + \int_{\mathcal{T}} X_i(t)e(t)dt
\end{aligned}
$$

where

- $\boldsymbol{x}_i = (x_{i,1}, \ldots, x_{i,r})^\top$
- $\mathbf{x}_i = \left(\int_{\mathcal{T}} \mathbf{b}_r(t)\mathbf{b}_p(t)^\top dt\right)^\top \boldsymbol{x}_i$
- $\varepsilon_i^\star = \varepsilon_i + \int_{\mathcal{T}} X_i(t)e(t)dt \sim \mathcal{N}(0, {\sigma_{z_i}^\star}^2)$.

Representation of the functional experts

The functional linear expert regressor z is then represented as :

$$\begin{aligned}
Y_i &= \beta_{z_i,0} + \int_{\mathcal{T}} X_i(t)\beta_{z_i}(t)dt + \varepsilon_i = \beta_{z_i,0} + \int_{\mathcal{T}} \boldsymbol{x}_i^\top \mathbf{b}_r(t)\left(\mathbf{b}_p^\top(t)\boldsymbol{\eta}_{z_i} + e_i(t)\right)dt + \varepsilon_i\\
&= \beta_{z_i,0} + \boldsymbol{x}_i^\top\left(\int_{\mathcal{T}} \mathbf{b}_r(t)\mathbf{b}_p^\top(t)dt\right)\boldsymbol{\eta}_{z_i} + \int_{\mathcal{T}} X_i(t)e(t)dt + \varepsilon_i\\
&= \beta_{z_i,0} + \boldsymbol{\eta}_{z_i}^\top \mathbf{x}_i + \varepsilon_i + \int_{\mathcal{T}} X_i(t)e(t)dt
\end{aligned}$$

where

- $\boldsymbol{x}_i = (x_{i,1}, \ldots, x_{i,r})^\top$
- $\mathbf{x}_i = \left(\int_{\mathcal{T}} \mathbf{b}_r(t)\mathbf{b}_p(t)^\top dt\right)^\top \boldsymbol{x}_i$
- $\varepsilon_i^\star = \varepsilon_i + \int_{\mathcal{T}} X_i(t)e(t)dt \sim \mathcal{N}(0, {\sigma_{z_i}^\star}^2)$.

The FunME expert (1) can thus be expressed as

$$Y_i \quad = \quad \beta_{z_i,0} + \boldsymbol{\eta}_{z_i}^\top \mathbf{x}_i + \varepsilon_i^\star, \quad i = 1, \ldots, n, \tag{7}$$

and we have $f(y_i|x_i(.), z_i = k; \boldsymbol{\theta}_k) = \phi(y_i; \beta_{k,0} + \boldsymbol{\eta}_k^\top \mathbf{x}_i, {\sigma_k^\star}^2)$ where $\boldsymbol{\theta}_k = (\beta_{k,0}, \boldsymbol{\eta}_k^\top, {\sigma_k^\star}^2)^\top \in \mathbb{R}^{p+2}$ is the unknown parameter vector of expert density k

FunME model

The Functional ME model

Combining (5) and (7), the resulting FunME distribution is defined by

$$f(y_i|X_i;\boldsymbol{\Psi}) = \sum_{k=1}^{K} \pi_k(\mathbf{r}_i;\boldsymbol{\xi})\phi(y_i;\beta_{k,0} + \boldsymbol{\eta}_k^\top \mathbf{x}_i, \sigma_k^{\star 2}) \quad (8)$$

where $\pi_k(\mathbf{r}_i;\boldsymbol{\xi}) = \exp\{\alpha_{k,0} + \boldsymbol{\zeta}_k^\top \mathbf{r}_i\} \Big/ 1 + \sum_{k'=1}^{K-1} \exp\{\alpha_{k',0} + \boldsymbol{\zeta}_{k'}^\top \mathbf{r}_i\}$ and $\boldsymbol{\Psi} = (\boldsymbol{\xi}^T, \boldsymbol{\theta}_1^T, \ldots, \boldsymbol{\theta}_K^T)^T$ the unknown parameter vector of the model

Model fitting

Since it is a mixture-of-experts model, then $\boldsymbol{\Psi}$ can be estimated by :

- ML via the EM algorithm [Jacobs et al., 1991, Dempster et al., 1977, McLachlan and Krishnan, 2008]
- Regularized ML to encourage sparsity (eg. lasso penalty [Tibshirani, 1996])
- Regularized ML (lasso-type regularization) on the derivatives of the $\alpha(\cdot)$ and $\beta(\cdot)$ function, by relying on the FLiRTI methodology [James et al., 2009]

1) FunME and MLE via the EM algorithm

Maximum-Likelihood Estimation

$$\widehat{\boldsymbol{\Psi}} \in \arg\max_{\boldsymbol{\Psi}} \log L(\boldsymbol{\Psi})$$

log-likelihood : $\log L(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k(X_i; \boldsymbol{\xi}) \phi(y_i; \beta_{k,0} + \boldsymbol{\eta}_k^\top \mathbf{x}_i, {\sigma_k^{\star}}^2)$

1) FunME and MLE via the EM algorithm

Maximum-Likelihood Estimation

$$\widehat{\boldsymbol{\Psi}} \in \arg\max_{\boldsymbol{\Psi}} \log L(\boldsymbol{\Psi})$$

log-likelihood : $\log L(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k(X_i;\boldsymbol{\xi})\phi(y_i;\beta_{k,0} + \boldsymbol{\eta}_k^\top \mathbf{x}_i, \sigma_k^{\star 2})$

The EM algorithm [Dempster et al., 1977]

$$\boldsymbol{\Psi}^{new} \in \arg\max_{\boldsymbol{\Psi}\in\Omega} \mathbb{E}[\log L_c(\boldsymbol{\Psi}) | \{X_i, Y_i\}_{i=1}^n, \boldsymbol{\Psi}^{old}]$$

complete log-likelihood :

$\log L_c(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \sum_{k=1}^{K} Z_{ik} \log\left[\pi_k(\mathbf{r}_i;\boldsymbol{\xi})\phi(y_i;\beta_{k,0} + \boldsymbol{\eta}_k^\top \mathbf{x}_i, \sigma_k^{\star 2})\right]$ where

$Z_{ik} = \mathbb{1}_{\{z_i = k\}}$, $k = 1, \ldots, K$

1) FunME and MLE via the EM algorithm

Maximum-Likelihood Estimation

$$\widehat{\boldsymbol{\Psi}} \in \arg\max_{\boldsymbol{\Psi}} \log L(\boldsymbol{\Psi})$$

log-likelihood : $\log L(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k(X_i; \boldsymbol{\xi})\phi(y_i; \beta_{k,0} + \boldsymbol{\eta}_k^\top \mathbf{x}_i, \sigma_k^{\star 2})$

The EM algorithm [Dempster et al., 1977]

$$\boldsymbol{\Psi}^{new} \in \arg\max_{\boldsymbol{\Psi}\in\Omega} \mathbb{E}[\log L_c(\boldsymbol{\Psi}) | \{X_i, Y_i\}_{i=1}^n, \boldsymbol{\Psi}^{old}]$$

complete log-likelihood :

$\log L_c(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \sum_{k=1}^{K} Z_{ik} \log \left[\pi_k(\mathbf{r}_i; \boldsymbol{\xi})\phi(y_i; \beta_{k,0} + \boldsymbol{\eta}_k^\top \mathbf{x}_i, \sigma_k^{\star 2})\right]$ where

$Z_{ik} = \mathbb{1}_{\{z_i = k\}}, \; k = 1, \ldots, K$

Clustering, Regression

- Expert label : $\widehat{z}_i = \arg\max_{1\leq k\leq K} \mathbb{E}(Z_{ik}|X_i; \widehat{\boldsymbol{\Psi}}), \quad (i = 1, \ldots, n)$
- Expert's mean function :
  $\widehat{y}_i | \{X_i, \widehat{z}_i = k\} = \widehat{\beta}_{k,0} + \widehat{\boldsymbol{\eta}}_k^\top \mathbf{x}_i, \quad (i = 1, \ldots, n; \; k = 1, \ldots, K)$
- FunME mean function : $\widehat{y}_i = \sum_{k=1}^{K} \pi_k(x_i(t); \widehat{\boldsymbol{\xi}})\{\widehat{\beta}_{k,0} + \widehat{\boldsymbol{\eta}}_k^\top \mathbf{x}_i\}, \quad (i = 1, \ldots, n)$

ML parameter estimation via EM (FunME-EM)

The E-Step

Compute the expectation of the complete-data log-likelihood, given the observed data $\{x_i(\cdot), y_i\}_{i=1}^n$, using the current parameter vector $\boldsymbol{\Psi}^{(s)}$:

$$
\begin{aligned}
Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(s)}) &= \mathbb{E}\left[\log L_c(\boldsymbol{\Psi})|\{x(\cdot),y\}_{i=1}^n;\boldsymbol{\Psi}^{(s)}\right]\\
&= \sum_{i=1}^n\sum_{k=1}^K \tau_{ik}^{(s)}\log\left[\pi_k(x_i(\cdot);\boldsymbol{\xi})\phi(y_i;\beta_{k,0}+\boldsymbol{\eta}_k^\top\mathbf{x}_i,\sigma_k^{\star 2})\right], \qquad (9)
\end{aligned}
$$

where $\tau_{ik}^{(s)} = \phi(y_i;\beta_{0,k}^{(s)}+\mathbf{x}_i^\top\boldsymbol{\eta}_k^{(s)},{\sigma^2}_k^{(s)})/f(y_i|\boldsymbol{x}_i;\boldsymbol{\Psi}^{(s)})$, is the probability that the pair $(x_i(t), y_i)$ is generated by the kth expert.

ML parameter estimation via EM (FunME-EM)

The E-Step

Compute the expectation of the complete-data log-likelihood, given the observed data $\{x_i(\cdot), y_i\}_{i=1}^n$, using the current parameter vector $\boldsymbol{\Psi}^{(s)}$:

$$
\begin{aligned}
Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(s)}) &= \mathbb{E}\left[\log L_c(\boldsymbol{\Psi})|\{x(\cdot), y\}_{i=1}^n;\boldsymbol{\Psi}^{(s)}\right]\\
&= \sum_{i=1}^{n}\sum_{k=1}^{K}\tau_{ik}^{(s)}\log\left[\pi_k(x_i(\cdot);\boldsymbol{\xi})\phi(y_i;\beta_{k,0}+\boldsymbol{\eta}_k^\top\mathbf{x}_i,\sigma_k^{\star 2})\right], \qquad (9)
\end{aligned}
$$

where $\tau_{ik}^{(s)} = \phi(y_i;\beta_{0,k}^{(s)}+\mathbf{x}_i^\top\boldsymbol{\eta}_k^{(s)},{\sigma^2}_k^{(s)})/f(y_i|\boldsymbol{x}_i;\boldsymbol{\Psi}^{(s)})$, is the probability that the pair $(x_i(t), y_i)$ is generated by the kth expert.

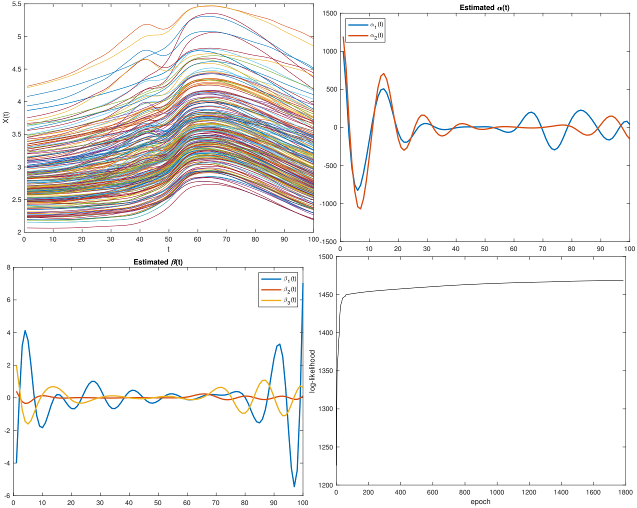
The M-Step

- Update the value of the parameter vector $\boldsymbol{\Psi}$ by $\boldsymbol{\Psi}^{(s+1)} = \arg\max_{\boldsymbol{\Psi}} Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(s)})$
- Separate maximizations w.r.t the gating network and the experts network

$$
\boldsymbol{\xi}^{(s+1)} = \arg\max_{\boldsymbol{\Psi}}\{Q(\boldsymbol{\xi};\boldsymbol{\Psi}^{(s)}) = \sum_{i=1}^{n}\sum_{k=1}^{K}\tau_{ik}^{(s)}\log\pi_k(x_i(\cdot);\boldsymbol{\xi})\} \qquad (10)
$$

$$
\boldsymbol{\theta}_k^{(s+1)} = \arg\max_{\boldsymbol{\Psi}}\{Q(\boldsymbol{\theta}_k;\boldsymbol{\Psi}^{(s)}) = \sum_{i=1}^{n}\tau_{ik}^{(s)}\log\phi(y_i;\beta_{k,0}+\boldsymbol{\eta}_k^\top\mathbf{x}_i,\sigma_k^{\star 2})\} (11)
$$

Example



X(t)
t
Estimated α(t)
α_1(t)
α_2(t)
Estimated β(t)
β_1(t)
β_2(t)
β_3(t)
log-likelihood
epoch

2) Regularized MLE via an EM-lasso algorithm

$\hookrightarrow p \gg n$ to ensure a good approximation of $\beta_z(t)$ by $\boldsymbol{\eta}_z^\top \mathbf{b}_p(t)$ (tradeoff between smoothness of the functional predictor and complexity of the estimation problem.)

Regularized Maximum-Likelihood Estimation

$$\widehat{\boldsymbol{\Psi}} \in \arg\max_{\boldsymbol{\Psi}} \log L(\boldsymbol{\Psi}) - \mathsf{Pen}_{\lambda,\chi}(\boldsymbol{\Psi})$$

log-likelihood : $\log L(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k(X_i;\boldsymbol{\xi})\phi(y_i;\beta_{k,0} + \boldsymbol{\eta}_k^\top \mathbf{x}_i, \sigma_k^{\star 2})$

2) Regularized MLE via an EM-lasso algorithm

$\hookrightarrow p \gg n$ to ensure a good approximation of $\beta_z(t)$ by $\boldsymbol{\eta}_z^\top \mathbf{b}_p(t)$ (tradeoff between smoothness of the functional predictor and complexity of the estimation problem.)

Regularized Maximum-Likelihood Estimation

$$\widehat{\boldsymbol{\Psi}} \in \arg\max_{\boldsymbol{\Psi}} \log L(\boldsymbol{\Psi}) - \text{Pen}_{\lambda,\chi}(\boldsymbol{\Psi})$$

log-likelihood : $\log L(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k(X_i; \boldsymbol{\xi}) \phi(y_i; \beta_{k,0} + \boldsymbol{\eta}_k^\top \mathbf{x}_i, \sigma_k^{\star 2})$

The EM-lasso algorithm

$$\boldsymbol{\Psi}^{new} \in \arg\max_{\boldsymbol{\Psi} \in \Omega} \mathbb{E}[\log L^c_{\lambda,\chi}(\boldsymbol{\Psi}) | \{X_i, Y_i\}_{i=1}^n, \boldsymbol{\Psi}^{old}]$$

complete log-likelihood :

$\log L^c_{\lambda,\chi}(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \sum_{k=1}^{K} Z_{ik} \log \left[\pi_k(X_i; \boldsymbol{\xi}) \phi(y_i; \beta_{k,0} + \boldsymbol{\eta}_k^\top \mathbf{x}_i, \sigma_k^{\star 2})\right] - \text{Pen}_{\lambda,\chi}(\boldsymbol{\Psi})$

Lasso regularization

$$\text{Pen}_{\lambda,\chi}(\boldsymbol{\Psi}) = \lambda \sum_{k=1}^{K} \|\boldsymbol{\eta}_k\|_1 + \chi \sum_{k=1}^{K-1} \|\boldsymbol{\xi}_k\|_1 \qquad (12)$$

where λ and χ are positive real values representing tuning parameters.

Regularized MLE via EM-lasso (FunME-EMlasso)

The EM-lasso algorithm for FunME

- E-Step : unchanged
- M-Step : $\boldsymbol{\Psi}^{(s+1)} = \arg\max_{\boldsymbol{\Psi}} \{Q_{\lambda,\chi}(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(s)}) = Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(s)}) - \mathsf{Pen}_{\lambda,\chi}(\boldsymbol{\Psi})\}$

Updating the expert' network parameters

$\boldsymbol{\theta}_k^{(s+1)} \in \arg\max_{\boldsymbol{\theta}_k} Q\lambda(\boldsymbol{\theta}_k;\boldsymbol{\Psi}^{(s)})$ with

$$Q_\lambda(\boldsymbol{\theta}_k;\boldsymbol{\Psi}^{(s)}) = \sum_{i=1}^{n} \tau_{ik}^{(s)} \log \phi(y_i;\beta_{k,0} + \boldsymbol{\eta}_k^\top \mathbf{x}_i, \sigma_k^{\star 2}) - \lambda \sum_{j=1}^{p} |\eta_{kj}|,$$

↪ A weighted LASSO problem for the $\boldsymbol{\eta}_k$'s

↪ Apply the LASSO machinery

↪ the update of $\sigma_k^{\star 2}$ is a weighted variant of the standard univariate Gaussian regression

Updating the gating network parameters

Updating the gating network parameters

$\boldsymbol{\xi}^{(s+1)} \in \arg\max_{\boldsymbol{\xi}} Q\chi(\boldsymbol{\xi}; \boldsymbol{\Psi}^{(s)})$ with

$$Q_\chi(\boldsymbol{\xi}; \boldsymbol{\Psi}^{(s)}) = \sum_{i=1}^{n}\sum_{k=1}^{K} \tau_{ik}^{(s)} \log \pi_k(\mathbf{r}_i; \boldsymbol{\xi}) - \chi \sum_{k=1}^{K-1}\sum_{j=1}^{q} |\xi_{kj}|$$

$$= \sum_{i=1}^{n}\left(\sum_{k=1}^{K-1} \tau_{ik}^{(s)} \left(\alpha_{k,0} + \boldsymbol{\zeta}_k^\top \mathbf{r}_i\right) - \log\left(1 + \sum_{k'=1}^{K-1} \exp\{\alpha_{k',0} + \boldsymbol{\zeta}_{k'}^\top \mathbf{r}_i\}\right)\right) - \chi \sum_{k=1}^{K-1}\sum_{j=1}^{q} |\xi_{kj}|,$$

$\hookrightarrow$ A weighted version of the regularized multinomial logistic problem (e.g [Mousavi and Sørensen, 2017])

- There is no closed-form solution
- we then use a Newton-Raphson with Coordinate Ascent updates of the gating network coefficients ξ_{kj}.

Coordinate Ascent for the gating network

For each expert k, for $j = 1, \ldots, p$:

$$\begin{aligned} \zeta_{k,j}^{(t+1)} &= \frac{\mathcal{S}\Big(\sum_{i=1}^{n} w_{ik} \mathbf{r}_{ij} (\tilde{h}_i^{(t)} - \tilde{z}_i^{(t)}); \chi \Big)}{\sum_{i=1}^{n} w_{ik} \mathbf{r}_{ij}^2} \\ &= \mathcal{S}\left(\mathbf{R}_j^T \mathbf{W}_k^{(t)} (\tilde{\boldsymbol{h}}_i^{(t)} - \tilde{\boldsymbol{z}}_i^{(t)}); \chi \right) / (\mathbf{R}_j^T \mathbf{W}_k^{(t)} \mathbf{R}_j) \end{aligned} \tag{13}$$

where

- $\tilde{h}_i^{(s)} = \alpha_{k,0}^{(s)} + \mathbf{r}_i^\top \boldsymbol{\zeta}_k + (\tau_{ik}^{(s)} - \pi_k(\mathbf{r}_i; \boldsymbol{\xi}^{(s)}))/w_{ik}$ is the working response
- $\tilde{z}_i^{(s)} = \alpha_{k,0}^{(s)} + \mathbf{r}_i^\top \boldsymbol{\zeta}_k - \mathbf{r}_{ij} \zeta_{k,j}^{(t+1)}$;fitted value excluding the contribution from $\zeta_{k,j}$
- $w_{ik} = \pi_k(\mathbf{r}_i; \boldsymbol{\xi}^{(t)})(1 - \pi_k(\mathbf{r}_i; \boldsymbol{\xi}^{(t)}))$
- $\mathbf{W}_k^{(t)} = \text{diag}(w_{ik}, \ldots, w_{nk})$ and $\mathbf{R}_j$ is the jth column of $\mathbf{R} = (\mathbf{r}_1, \ldots, \mathbf{r}_n)^\top$,
- $\mathcal{S}(\cdot)$ is a soft-thresholding operator defined by $\mathcal{S}(u, \chi) = \text{sign}(u)(|u| - \chi)_+$ and $(x)_+$ a shorthand for $\max\{x, 0\}$

For $\alpha_{k,0}$, the update is given by

$$\alpha_{k,0}^{(t+1)} = \frac{\sum_{i=1}^{n} w_{ik} (\tilde{h}_i^{(t)} - \mathbf{r}_i^\top \boldsymbol{\zeta}_k^{(t)})}{\sum_{i=1}^{n} w_{ik}} = \mathbf{W}_k^{(q)} (\tilde{\boldsymbol{h}}_i^{(t)} - \mathbf{R}\boldsymbol{\zeta}_i^{(t)}) / \text{trace}(\mathbf{W}_k^{(q)})$$

Coordinate Ascent for the expert network

For each expert k, for $j = 1, \ldots, p$:

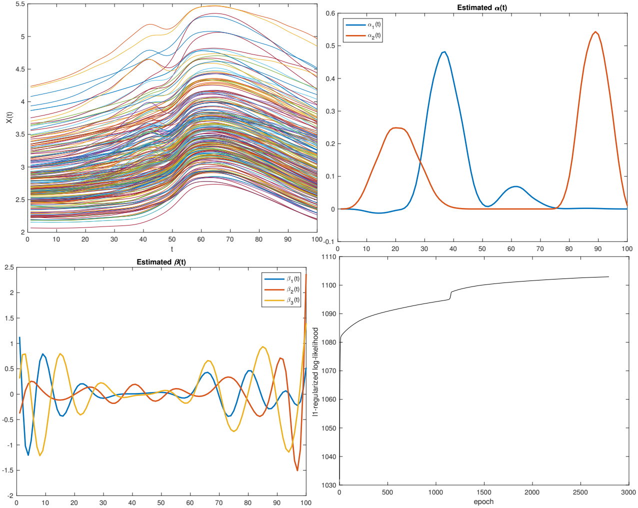
$$\begin{aligned} \eta_{kj}^{(q+1)} &= \mathcal{S}\left(\sum_{i=1}^{n} \tau_{ik}^{(s)}(y_i - \beta_{k0}^{(s)} - \mathbf{x}_i^\top \boldsymbol{\beta}_k^{(s)} + \eta_{kj}^{(q)} \mathbf{x}_{ij}); \lambda {\sigma_k^{(s)}}^2\right) / \sum_{i=1}^{n} \tau_{ik}^{(s)} \mathbf{x}_{ij}^2 \\ &= \mathcal{S}\left(\mathbf{X}_j^T \mathbf{W}_k^{(q)} \boldsymbol{r}_{kj}^{(q)}; \lambda {\sigma_k^{(s)}}^2\right) / (\mathbf{X}_j^T \mathbf{W}_k^{(q)} \mathbf{X}_j) \end{aligned} \quad (14)$$

where $\mathbf{X}_j$ is the jth column of the design matrix $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^\top$,
$\mathbf{W}_k^{(q)} = \text{diag}(\tau_{1k}^{(q)}, \ldots, \tau_{nk}^{(q)})$,
$\boldsymbol{r}_{kj}^{(q)} = \boldsymbol{y} - \beta_{k0}^{(q)} \mathbf{1}_n - \mathbf{X}\boldsymbol{\beta}_k^{(q)} + \beta_{kj}^{(q)} \mathbf{X}_j$ is the residual without the contribution of the jth coefficient
$\mathcal{S}(u, \eta) := \text{sign}(u)(|u| - \eta)_+$ is the soft-thresholding operator with $(.)_+ = \max\{., 0\}$.

$$\beta_{k,0}^{(s+1)} = \frac{\sum_{i=1}^n \tau_{ik}^{(s)} (y_i - \mathbf{x}_i^\top \boldsymbol{\eta}_k^{(s)})}{\sum_{i=1}^n \tau_{ik}^{(s)}} = \mathbf{W}_k^{(q)}(\boldsymbol{y} - \mathbf{X}\boldsymbol{\eta}_k^{(q)}) / \text{trace}(\mathbf{W}_k^{(q)}), \quad (15)$$

$$\begin{aligned} \sigma_k^{2(s+1)} &= \frac{\sum_{i=1}^n \tau_{ik}^{(s)} \left(y_i - \beta_{k,0}^{(s+1)} - \mathbf{x}_i^\top \boldsymbol{\eta}_k^{(s+1)}\right)^2}{\sum_{i=1}^n \tau_{ik}^{(s)}} \\ &= \|\sqrt{\mathbf{W}_k^{(s+1)}} \left(\boldsymbol{y} - \beta_{k0}^{(s+1)} \mathbf{1}_n - \mathbf{X}\boldsymbol{\eta}_k^{(s+1)}\right)\|_2^2 / \text{trace}(\mathbf{W}_k^{(q)}) \end{aligned} \quad (16)$$

Example



X(t)
t
Estimated α(t)
α_1(t)
α_2(t)
Estimated β(t)
β_1(t)
β_2(t)
β_3(t)
l1-regularized log-likelihood
epoch

3) FunME by regularizing functional derivatives

- For FunME-LASSO regularization described previously, there is no actually reason that the functions $\beta(.)$ and $\alpha(.)$ be sparse.
- So regularizing the parameter vectors representing these functions has no obvious interpretability

$\hookrightarrow$ FLiRTI methodology [James et al., 2009] offers an interpretable and sparse fit for functional linear regression

- Regularization is performed on the the derivatives of the coeffcient function, rather than on the paramters of the function

$\hookrightarrow$ We rely on FLiRTI methodology for the regression functions $\beta_{z_i}(t)$ (and $\alpha_{z_i}(t)$)

FLiRTI : determine whether the dth derivative of $\beta_{z_i}(t)$ is zero or not at each point t_j.

$\hookrightarrow$ can produce a highly interpretable estimate for $\beta_{z_i}(t)$ curves :
$\beta^{(0)}_{z_i}(t) = 0$ implies that $X(t)$ has no effect on Y at t
$\beta^{(1)}_{z_i}(t) = 0$ means that $\beta_{z_i}(t)$ is constant at t,
$\beta^{(0)}_{z_i}(t) = 1$ shows that $\beta_{z_i}(t)$ is a linear function of t, etc.

- Let D^d be the dth finite difference operator defined recursively as

$$D^1\mathbf{b}(t_j) = p[\mathbf{b}(t_j) - \mathbf{b}(t_{j-1})],$$
$$D^2\mathbf{b}(t_j) = D[D\mathbf{b}(t_j)] = p^2[\mathbf{b}(t_j) - 2\mathbf{b}(t_{j-1}) + \mathbf{b}(t_{j-2})],$$
$$D^d\mathbf{b}(t_j) = D[D^{d-1}\mathbf{b}(t_j)].$$

- $D^d\mathbf{b}(t_j)$ is an approximation for $\mathbf{b}^{(d)}(t_j) = [b_1^{(d)}(t_j), \ldots, b_p^{(d)}(t_j)]^\top$
- $\mathbf{A}_p = [D^d\mathbf{b}(t_1), D^d\mathbf{b}(t_2), \ldots, D^d\mathbf{b}(t_p)]^\top$ (the approximate derivative matrix)
- Let $\boldsymbol{\gamma}_{z_i} = \mathbf{A}_p\boldsymbol{\eta}_{z_i}$
- If $\beta_{z_i}^{(d)}(t) = 0$ over a large regions of t for some d, then $\boldsymbol{\gamma}_{z_i}$ is sparse.

$\hookrightarrow$ $\boldsymbol{\gamma}_{z_i} = [\gamma_{z_i,1}, \ldots, \gamma_{z_i,p}]^\top$ provides a sparse estimate for $[\beta_{z_i}^{(d)}(t_1), \ldots, \beta_{z_i}^{(d)}(t_p)]^\top$.

FLiRTI for the expert' network of FunME

$$\begin{aligned} Y_i &= \beta_{z_i,0} + \boldsymbol{\eta}_{z_i}{}^\top\mathbf{x}_i + \varepsilon_i^\star = \beta_{z_i,0} + (\mathbf{A}_p^{-1}\boldsymbol{\gamma}_{z_i})^\top\mathbf{x}_i + \varepsilon_i^\star \\ &= \beta_{z_i,0} + (\mathbf{x}_i^\top\mathbf{A}_p^{-1})\boldsymbol{\gamma}_{z_i} + \varepsilon_i^\star \\ &= \beta_{z_i,0} + \mathbf{v}_i{}^\top\boldsymbol{\gamma}_{z_i} + \varepsilon_i^\star. \end{aligned}$$

and we now have $\boldsymbol{\theta}_k = (\beta_{k,0}, \boldsymbol{\gamma}_k^\top, \sigma_k^{\star 2})^\top$ parameter vector of expert density k

FLiRTI for the gating network of FunME

- Let $\boldsymbol{\omega}_k = \mathbf{A}_q \boldsymbol{\zeta}_k$ where $\mathbf{A}_q = [D^d\mathbf{b}(t_1), D^d\mathbf{b}(t_2), \ldots, D^d\mathbf{b}(t_q)]^\top$
  $\hookrightarrow$ we get $\boldsymbol{\zeta}_k = \mathbf{A}_q^{-1}\boldsymbol{\omega}_k$.

The gating network probabilities become

$$\pi_k(\boldsymbol{v}_i;\mathbf{w}) = \frac{\exp\{\alpha_{k,0} + \boldsymbol{\zeta}_k{}^\top \mathbf{r}_i\}}{1+\sum_{k'=1}^{K-1}\exp\{\alpha_{k',0} + \boldsymbol{\zeta}_{k'}{}^\top \mathbf{r}_i\}} = \frac{\exp\{\alpha_{k,0} + \boldsymbol{v}_i^\top \boldsymbol{\omega}_k\}}{1+\sum_{k'=1}^{K-1}\exp\{\alpha_{k',0} + \boldsymbol{v}_i^\top \boldsymbol{\omega}_{k'}\}} \quad (17)$$

with $\boldsymbol{v}_i = \mathbf{r}_i^\top \mathbf{A}_q^{-1}$ is the new predictor and the new gating network parameter vector $\mathbf{w} = ((\alpha_{1,0}, \boldsymbol{\omega}_1^\top), \ldots, (\alpha_{K-1,0}, \boldsymbol{\omega}_{K-1}^\top))^\top$ and $(\alpha_{K-1,0}, \boldsymbol{\omega}_K^\top)^\top$ is a null vector.

The resulting FunME distribution and parameter estimation

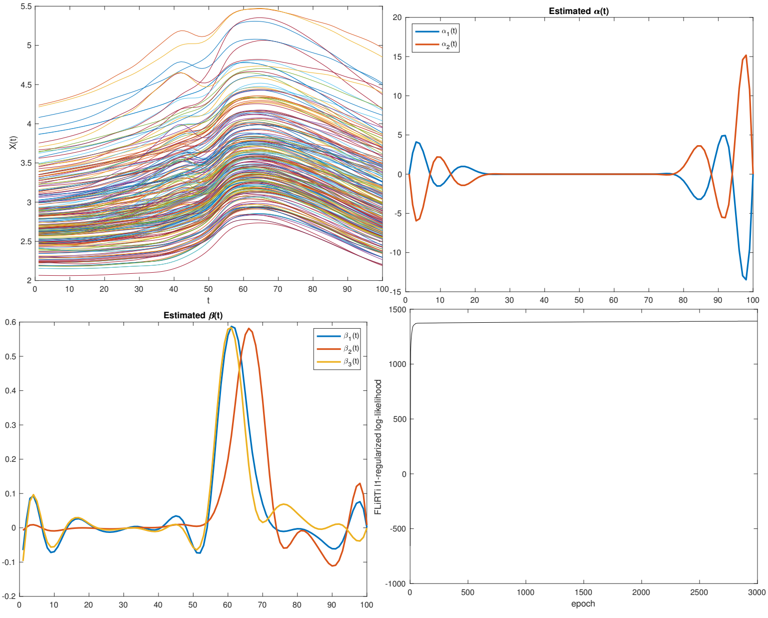
$$f(y_i|u_i(.);\boldsymbol{\Psi}) = \sum_{k=1}^{K} \pi_k(\boldsymbol{v}_i;\mathbf{w})\phi(y_i;\beta_{k,0} + \boldsymbol{\gamma}_k^\top \mathbf{v}_i, \sigma_k^{\star 2}) \quad (18)$$

where $\boldsymbol{\Psi} = (\mathbf{w}^T, \boldsymbol{\Psi}_1^T, \ldots, \boldsymbol{\Psi}_K^T)^T$ the unknown parameter vector of the model

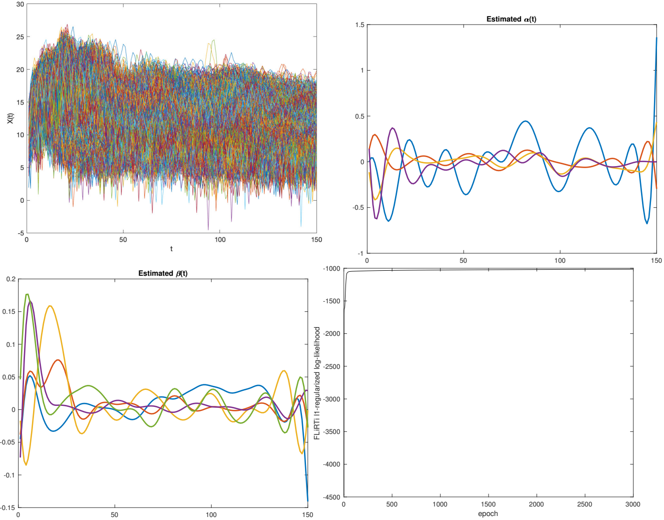
$\hookrightarrow$ Apply the EM-Lasso algorithm developed previously with :

- Predictors : $\mathbf{v}_i = \mathbf{x}_i^\top \mathbf{A}_p^{-1}$ and $\boldsymbol{v}_i = \mathbf{r}_i^\top \mathbf{A}_q^{-1}$
- Regularization : on $\boldsymbol{\omega}$'s and $\boldsymbol{\gamma}$'s : $\text{Pen}_{\lambda,\chi}(\boldsymbol{\Psi}) = \lambda\sum_{k=1}^{K}\|\boldsymbol{\gamma}_k\|_1 + \chi\sum_{k=1}^{K-1}\|\boldsymbol{\omega}_k\|_1$

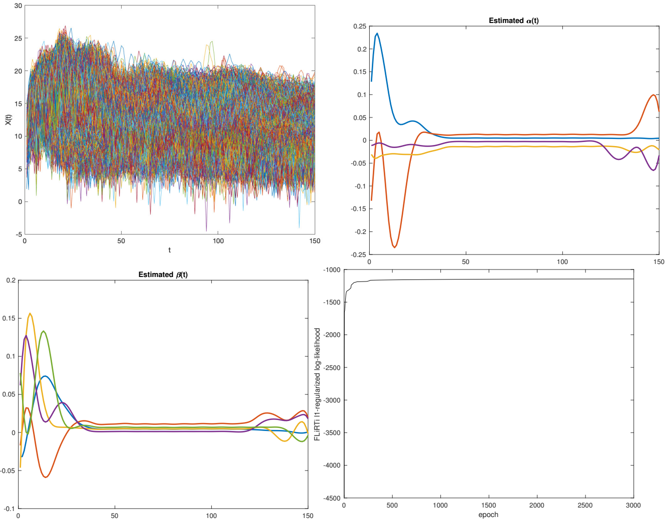
Example : Tecator data



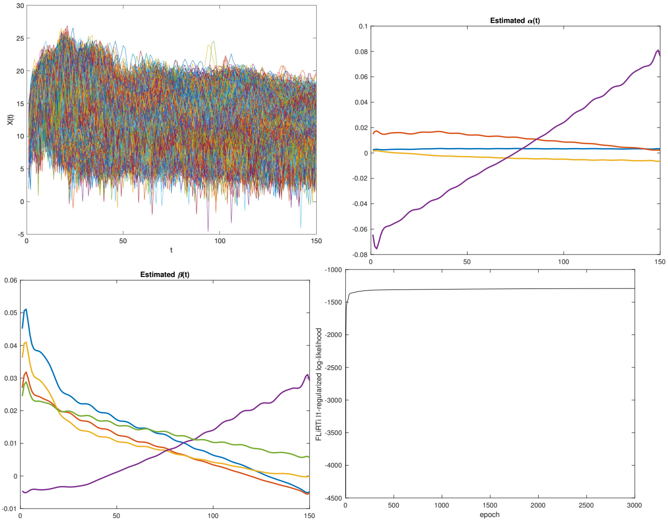
Example : Phonemes data (K=5), d=0



Example : Phonemes data (K=5), d=1



Example : Phonemes data (K=5), d=2



FunME for unobserved predictors

The functional predictors $X_i(t)$ are in general unobserved directly

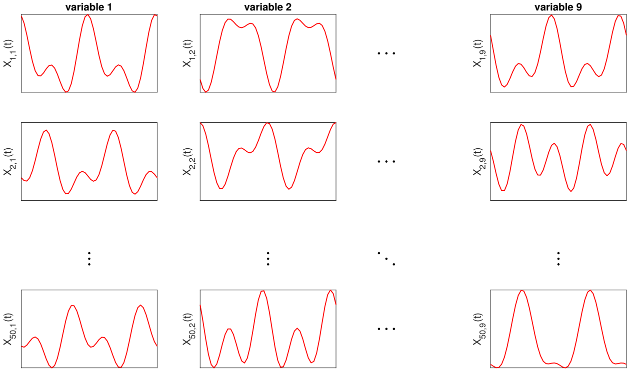


FIGURE – functional predictors $X_{ij}(t)$ $t \in \mathcal{T}$

FunME for unobserved predictors

We rather observe $U_i(t)$ a noisy version of $X_i(t)$

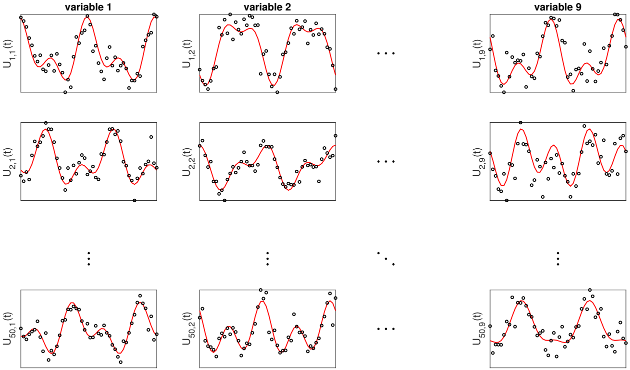


FIGURE – Noisy functional predictors $U_{ij}(t)$ $t \in \mathcal{T}$

Until now the functional predictors $X_i(t)$ are represented by basis expansion as

$$X_i(t) = \sum_{j=1}^{r} x_{ij} b_j(t) = \boldsymbol{x}_i^\top \mathbf{b}_r(t),$$

$\hookrightarrow$ the coefficients $x_{ij} = \int_{\mathcal{T}} X_i(t) b_j(t) dt$ are unknown since $X_i(t)$ is not observed

$\hookrightarrow$ We first model $U_i(t)$ (for a single variable) as

$U_i(t) = X_i(t) + \delta_i(t), \quad i = 1, \ldots, n, \quad \delta_i \sim \mathcal{N}(0, \sigma_\delta^2)$

We assume that the δ_i's are independent of the $X_i(\cdot)$'s and the Y_i's.

and propose an unbiased estimator of x_{ij} from $U_i(t)$ defined as

$$\widehat{x}_{ij} := \int_{\mathcal{T}} U_i(t) b_j(t) dt.$$

Indeed, we have $\mathbb{E}(\widehat{x}_{ij}) = \int_{\mathcal{T}} \mathbb{E}(U_i(t)) b_j(t) dt = \int_{\mathcal{T}} X_i(t) b_j(t) dt = x_{ij}$.

$\hookrightarrow$ Thus, an estimate $\widehat{X}_i(t)$ of $X_i(t)$ can be given as

$$\widehat{X}_i(t) = \widehat{\boldsymbol{x}}_i^\top \mathbf{b}_r(t), \quad i = 1, \ldots, n, \tag{19}$$

with $\widehat{\boldsymbol{x}}_i = (\widehat{x}_{i1}, \ldots, \widehat{x}_{ir})^\top$.

$\hookrightarrow$ The previous models/algorithms apply by replacing $\boldsymbol{x}_i$ by its estimate $\widehat{\boldsymbol{x}}_i$

FunME for classification

$Y \in [G]$ represents the known group label of the functional predictor $X(\cdot)$.

- Expert modeling : functional multinomial logistic distribution

$$\mathbb{P}(y_i|X_i(.), Z_i = k; \boldsymbol{\beta}) = \prod_{g=1}^{G} \left[\frac{\exp\left\{\beta_{kg,0} + \int_{\mathcal{T}} X(t)\beta_{kg}(t)dt\right\}}{1 + \sum_{g'=1}^{G-1} \exp\left\{\beta_{kg',0} + \int_{\mathcal{T}} X(t)\beta_{kg'}(t)dt\right\}} \right]^{\mathbb{I}(y_i=g)}$$

$\hookrightarrow$ use the same basis representation for the linear predictors $\beta_{kg,0} + \int_{\mathcal{T}} X(t)\beta_{kg}(t)dt$

- M-Step : Newton-Raphson with coordinate ascent

$$\boldsymbol{\theta}_k^{(t+1)} = \boldsymbol{\theta}_k^{(t)} - \left(\frac{\partial^2 Q_\lambda(\boldsymbol{\theta}_k; \boldsymbol{\Psi}^{(s)})}{\partial \boldsymbol{\theta}_k \partial \boldsymbol{\theta}_k^T}(\boldsymbol{\theta}_k^{(t)}) \right)^{-1} \frac{\partial Q_\lambda(\boldsymbol{\theta}_k; \boldsymbol{\Psi}^{(s)})}{\partial \boldsymbol{\theta}_k}(\boldsymbol{\theta}_k^{(t)})$$

with $\boldsymbol{\theta}_k = (\boldsymbol{\theta}_{k,1}^\top, \ldots, \boldsymbol{\theta}_{k,G-1}^\top)$ with $\boldsymbol{\theta}_{k,g} = (\beta_{kg,0}, \boldsymbol{\eta}_{kg}^\top)^\top \in \mathbb{R}^{p+1}$ for $g \in [G-1]$, be the unknown parameter vector of expert distribution k to be estimated.

- Bayes (Maximum A Posteriori) rule :

$$\widehat{y} = \arg \max_{1 \leq y \leq G} \mathbb{P}(Y = y|u; \boldsymbol{\Psi}) = \arg \max_{y=1}^{G} \sum_{k=1}^{K} \pi_k(\mathbf{r}; \boldsymbol{\xi}) p(y|\mathbf{x}; \boldsymbol{\theta}_k)$$

Concluding remarks

- A model for heterogeneous data with functional predictors
- The model inference can be performed by the EM algorithm
- Allows to perform feature selection
- Relying on FLiRTI methodology allows to keep the feature selection interpretable

Ongoing :

- BIC-based procedure for model selection
- Numerical experiments
- Package (currently codes ares written in Matlab and will be made public soon)
- Extension to the multivariate setting
- Extension to the case of functional predictors and functional responses

References I

Jeffrey D. Banfield and Adrian E. Raftery. Model-based Gaussian and non-Gaussian clustering. *Biometrics*, 49(3) :803–821, 1993.

C. Bouveyron and J. Jacques. Model-based clustering of time series in group-specific functional subspaces. *Adv. Data Analysis and Classification*, 5(4) :281–300, 2011.

C. Bouveyron, L. Bozzi, J. Jacques, and F.-X. Jollois. The functional latent block model for the co-clustering of electricity consumption curves. *Journal of the Royal Statistical Society, Series C*, 2018.

G. Celeux and G. Govaert. Gaussian Parsimonious Clustering Models. *Pattern Recognition*, 28(5) :781–793, 1995.

Gilles Celeux, Cathy Maugis-Rabusseau, and Mohammed Sedki. Variable selection in model-based clustering and discriminant analysis with a regularization approach. *Advances in Data Analysis and Classification*, 13(1) :259–278, Mar 2019. ISSN 1862-5355. doi: 10.1007/s11634-018-0322-5. URL `https://doi.org/10.1007/s11634-018-0322-5`.

Faicel Chamroukhi and Bao T. Huynh. Regularized Maximum Likelihood Estimation and Feature Selection in Mixtures-of-Experts Models. *Journal de la Société Française de Statistique*, 160(1) :57–85, March 2019. URL `https://chamroukhi.com/papers/Chamroukhi_Huynh_jsfds-published.pdf`.

Faicel Chamroukhi and Hien D. Nguyen. Model-based clustering and classification of functional data. *Wiley Interdisciplinary Reviews : Data Mining and Knowledge Discovery*, Dec 2018. URL `https://chamroukhi.com/papers/MBCC-FDA.pdf`. DOI : 10.1002/widm.1298.

A. Delaigle, P. Hall, and N. Bathia. Componentwise classification and clustering of functional data. *Biometrika*, 99(2) :299–313, 2012.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of The Royal Statistical Society, B*, 39(1) :1–38, 1977.

F. Ferraty and P. Vieu. Curves discrimination : a nonparametric functional approach. *Computational Statistics & Data Analysis*, 44(1-2) :161–173, 2003.

F. Ferraty and P. Vieu. *Nonparametric functional data analysis : theory and practice*. Springer series in statistics, 2006. ISBN 0-387-30369-3.

References II

R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1) : 79–87, 1991.

Julien Jacques and Cristian Preda. Functional data clustering : A survey. *Adv. Data Anal. Classif.*, 8(3) :231–255, September 2014. ISSN 1862-5347. doi: 10.1007/s11634-013-0158-y. URL http://dx.doi.org/10.1007/s11634-013-0158-y.

G. M. James and T. J. Hastie. Functional linear discriminant analysis for irregularly sampled curves. *Journal of the Royal Statistical Society Series B*, 63 :533–550, 2001.

G. M. James and C. Sugar. Clustering for sparsely sampled functional data. *Journal of the American Statistical Association*, 98 (462), 2003.

Gareth M. James. Generalized linear models with functional predictors. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 64(3) :411–432, 2002. ISSN 13697412, 14679868. URL http://www.jstor.org/stable/3088780.

Gareth M. James, Jing Wang, and Ji Zhu. Functional linear regression that's interpretable. *Annals of Statistics*, 37(5A) : 2083–2108, 2009.

M. I. Jordan and R. A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6 :181–214, 1994.

A. Khalili. New estimation and feature selection methods in mixture-of-experts models. *Canadian Journal of Statistics*, 38(4) : 519–539, 2010.

G. J. McLachlan and T. Krishnan. *The EM algorithm and extensions*. New York : Wiley, second edition, 2008.

G. J. McLachlan and D. Peel. *Finite mixture models*. New York : Wiley, 2000.

G. J. McLachlan, D. Peel, and R. W. Bean. Modelling high-dimensional data by mixtures of factor analyzers. *Computational Statistics & Data Analysis*, 41(3-4) :379–388, January 2003. URL http://www.sciencedirect.com/science/article/B6V8V-472JRC1-12/1/4d40244841bb6f7c8c454ca92e6cc347.

Seyed Nourollah Mousavi and Helle Sørensen. Multinomial functional regression with wavelets and lasso penalization. *Econometrics and Statistics*, 150–166, 2017. ISSN 2452-3062. doi: 10.1016/j.ecosta.2016.09.005.

Hans Müller and Ulrich Stadtmüller. Generalized functional linear models. *Ann. Statist*, pages 774–805, 2005.

References III

Hien D. Nguyen and Faicel Chamroukhi. Practical and theoretical aspects of mixture-of-experts modeling : An overview. *Wiley Interdisciplinary Reviews : Data Mining and Knowledge Discovery*, pages e1246–n/a, Feb 2018. ISSN 1942-4795. doi: 10.1002/widm.1246. URL http://dx.doi.org/10.1002/widm.1246.

Trung Tin Nguyen, Hien D Nguyen, Faicel Chamroukhi, and Geoffrey J McLachlan. Approximation by finite mixtures of continuous density functions that vanish at infinity. *Submitted*, March 2019. URL https://arxiv.org/pdf/1903.00147.pdf. Submitted.

Wei Pan and Xiaotong Shen. Penalized model-based clustering with application to variable selection. *Journal of Machine Learning Research*, 8(May) :1145–1164, 2007.

Xinghao Qiao, Shaojun Guo, and Gareth M. James. Functional graphical models. *Journal of the American Statistical Association*, 0(0) :1–12, 2018. doi: 10.1080/01621459.2017.1390466.

J. O. Ramsay and B. W. Silverman. *Functional Data Analysis*. Springer Series in Statistics. Springer, June 2005.

J.O. Ramsay, T.O. Ramsay, and L.M. Sangalli. Spatial functional data analysis. In F. Ferraty, editor, *Recent Advances in Functional Data Analysis and Related Topics*, pages 269–275. Springer, 2011.

A. Samé, F. Chamroukhi, G. Govaert, and P. Aknin. Model-based clustering and segmentation of time series with changes in regime. *Advances in Data Analysis and Classification*, pages 1–21, 2011. ISSN 1862-5347.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58(1) : 267–288, 1996.

Thank you for your attention !