Modèles de régression linéaire fonctionnelle pour données hétérogènes : Application à la prédiction de la QoE du streaming video et de la VoIP

Jean Steve TAMO TCHOMGUI

Thèse présentée par **Jean Steve Tamo Tchomgui**

Soutenue le **02 décembre 2024**

En vue de l'obtention du titre de Docteur de l'Université Lumière Lyon 2

Discipline : **Mathématiques appliquées**
Spécialité : **Statistiques**

Modèles de régression linéaire fonctionnelle pour données hétérogènes : Application à la prédiction de la QoE du streaming video et de la VoIP

Composition du Jury :

Sophie Dabo-Niang	Professeur, Université de Lille	Rapporteur
Faïcel Chamroukhi	Professeur, IRT SystemX, Université de Caen	Rapporteur
Emilie Devijver	Chargée de Recherche CNRS, Université de Grenoble	Examinateur
Sofiene Jelassi	Maître de Conférences, Université de Rennes 1	Examinateur
Julien Jacques	Professeur, Université de Lyon 2	Directeur de thèse
Stéphane Chretien	Professeur, Université de Lyon 2	Co-directeur de Thèse
Guillaume Fraysse	Orange Innovation, Châtillon	Invité
Vincent Barriac	Orange Innovation, Lannion	Invité

Thèse préparée au sein l'équipe DMD du laboratoire ERIC de l'école doctorale ED512 InfoMaths.
5 Avenue Mendes France, 69500 Bron

Résumé

L'expansion rapide des moyens de communication via internet ces dernières décennies et la course à la compétitivité des entreprises dans ce secteur ont renforcé l'importance de la qualité d'expérience, Quality of Experience (QoE) en anglais, dans l'évaluation des performances et de la qualité d'un réseau. La QoE est une mesure subjective qui reflète la perception globale de la qualité d'un service par l'utilisateur. Elle est influencée par une myriade de facteurs dont les plus importantes sont les conditions du réseau, les caractéristiques du contenu et les préférences individuelles de l'utilisateur. Pour des services telles que la voix sur IP, Voice Over IP (VoIP) en anglais, ou la lecture en continu des vidéos (streaming) par exemple, mesurer la QoE nécessite de disposer d'informations au niveau des applications (résolution, qualité de diffusion, temps d'interruption, écho, qualité vocale, ...) qui ne sont généralement pas à la disposition des opérateurs réseaux à moins qu'ils soient également fournisseurs de contenus ou aient des accords avec ces derniers. En l'absence de tels accords, une mesure de la QoE devient difficile compte tenu de sa nature subjective. De nombreux travaux de recherches antérieures ont exploré diverses approches de la prédiction de la QoE, notamment via des études subjectives sur les utilisateurs. Ces derniers ont l'inconvénient d'être coûteuses, lourdes en temps et difficile à automatiser. D'autres travaux faisant intervenir des modèles statistiques ont pallié ces insuffisances en tirant parti de la puissance des données et des techniques d'apprentissage automatique. Toutefois les modèles traditionnels de prédiction de la QoE, qui reposent souvent sur des observations discrètes, ne parviennent souvent pas à saisir la nature complexe, multidimensionnelle et dynamique des facteurs influençant la QoE. Cette thèse vise à répondre à ces limitations en employant de nouvelles méthodologies statistiques dans le cadre innovant et en plein expansion de l'Analyse des Données Fonctionnelles (ADF). L'ADF aussi connue sous le terme de Functional Data Analysis (FDA) en anglais, est un cadre méthodologique destiné à traiter et analyser des données qui sont de nature fonctionnelle, c'est-à-dire des données qui sont observées sur un ensemble continu comme le temps, l'espace, ou d'autres dimensions. Dans ce cadre, les observations ne sont pas simplement des points discrets comme réalisations de variables aléatoire réelles, mais des fonctions entières comme réalisations d'une fonction aléatoire. L'un des principaux défis de notre problématique réside dans le fait que la QoE en tant que variable réponse et les facteurs qui l'influencent (c'est-à-dire les paramètres des réseaux en tant que covariables) sont enregistrés au fil du temps. Dans ce contexte, nous sommes dans le cadre d'une régression d'une variable réponse fonctionnelle impliquant des covariables fonctionnelles. En utilisant la convention que le premier terme désigne la nature de la variable réponse et le second celle des covariables, on distingue en ADF les régression scalaire-sur-fonction, fonction-sur-scalaire et fonction-sur-fonction. Par rapport aux deux premiers types, le troisième que nous abordons ici est le moins développé dans la littérature. Pour étudier la relation entre la QoE notée $\mathrm{Y}(t)$ et les covariables notées $\mathrm{X}(t) = \big(\mathrm{X}^1(t),\ \ldots,\ \mathrm{X}^p(t)\big)$, une première approche va être de considérer deux types de modèles

régression linéaire abordés dans les travaux de Ramsay and Silverman (2005, Chapter 12):

- Le modèle **concurrent**:

$$\mathrm{Y}_i(t) \quad = \quad \mathrm{X}_i(t)^\top \beta(t) + \varepsilon_i(t), \tag{0.0.1}$$

avec $\beta(t) = \big(\beta_0(t), \quad \beta_1(t), \quad \ldots, \quad \beta_p(t)\big)^\top$, $\mathrm{X}_i(t) = \big(1, \quad \mathrm{X}_i^1(t), \quad \ldots, \quad \mathrm{X}_i^p(t)\big)^\top$ et où $\mathbb{E}(\varepsilon_i(t)) = 0$, $\mathbb{V}\mathrm{ar}(\varepsilon_i(t)) = \sigma_i$.

- Le modèle **intégral**:

$$\mathrm{Y}_i(t) \quad = \quad \gamma_0(t) + \int_0^t \mathrm{X}_i(s)^\top \gamma(s,t)\,\mathrm{d}s + \varepsilon_i(t), \tag{0.0.2}$$

avec $\gamma(s,t) = \big(\gamma_1(s,t) \quad \gamma_2(s,t) \quad \ldots \quad \gamma_p(s,t)\big)^\top$ et $\mathrm{X}_i(s) = \big(\mathrm{X}_i^1(s) \quad \mathrm{X}_i^2(s) \quad \ldots \quad \mathrm{X}_i^p(s)\big)^\top$. où $\mathbb{E}(\varepsilon_i(t)) = 0$, $\mathbb{V}\mathrm{ar}(\varepsilon_i(t)) = \sigma_i$.

Les paramètres fonctionnels $\beta(t)$ dans le modèle concurrent (0.0.1) et $\big(\gamma_0(t), \gamma(s,t)\big)$ dans le modèle intégral (0.0.2) sont supposés de carré intégrable. En pratique, la différence entre ces deux modèles résident dans l'interprétation de la variable réponse $\mathrm{Y}(t)$ qui est à effet instantané dans le modèle (0.0.1) ou à effet cumulatif des effets passés jusqu'au temps présent t dans le modèle (0.0.2).

Ces deux modèles, utilisés pour prendre en compte la nature dynamique de la QoE, sont influencés par une multitude de facteurs. Ces derniers peuvent varier considérablement d'une situation à une autre. Par exemple, l'impact de la latence sur la QoE peut dépendre de la résolution du contenu ou du type d'encodage utilisé. Dans ce cas un modèle unique pourrait avoir du mal à capturer de manière efficace cette spécificité. Le cadre des modèles de mélanges qui permet de partitionner l'espace des données en sous espaces plus homogènes est adapté pour pallier au problème d'hétérogénéité. Le contexte de modélisation prédictive dans lequel nous sommes nous conduit au mélange d'expert (Mixture-of-Experts (MoE) en anglais) où chaque "expert" se spécialise pour prédire la QoE dans une région homogène de l'espace des données et une fonction d'activation (gating) détermine quel expert est le plus pertinent pour une donnée spécifique. En combinant l'ADF et le mélange d'experts, nous obtenons un modèle robuste avec une grande flexibilité permettant une prédiction plus fiable de la QoE défini, pour un mélange à K classes avec la variable d'appartenance à une classe $\mathrm{Z} = (z_1, \ldots, z_\mathrm{K})$, par

$$\mathrm{MoE}\Big(\mathrm{Y}(t) \,|\, \mathrm{X}(t)\Big) \quad = \quad \sum_{k=1}^{\mathrm{K}} \pi_k\Big(\mathrm{X}(t), \alpha_k(t)\Big)\, \mathbb{E}\Big[\mathrm{Y}(t) \,|\, \mathrm{X}(t), z_k = 1\Big] \tag{0.0.3}$$

Où $\pi_k\Big(\mathrm{X}(t), \alpha_k(t)\Big)$ est la fonction d'activation, et $\mathbb{E}\Big[\mathrm{Y}(t) \,|\, \mathrm{X}(t), z_k = 1\Big]$ l'expert défini par le modèle concurrent (0.0.1)

Abstract

The rapid expansion of internet-based communication over the past decades, and the competitiveness between companies in this sector, has heightened the importance of Quality of Experience (QoE) in evaluating network performance and service quality. QoE is a subjective measure that reflects the user's overall perception of a service's quality, influenced by a myriad of factors, including network conditions, content characteristics and individual user preferences. For services such as Voice over IP (VoIP) or video streaming, accurately measuring QoE requires access to application-level information such as resolution, streaming quality, interruption times, echo, voice quality, etc... which is often not available to network operators unless they are also content providers or have agreements with them. In the absence of such agreements, measuring QoE becomes challenging due to its subjective nature. Numerous previous research efforts have explored various approaches to predicting QoE, including subjective user studies. However, these studies are often costly, time-consuming, and difficult to automate. Other research leveraging statistical models has addressed these shortcomings by harnessing the power of data and machine learning techniques. Nonetheless, traditional QoE prediction models, which frequently rely on discrete observations, often fail to capture the complex, multidimensional, and dynamic nature of the factors influencing QoE. This thesis aims to address these limitations by employing novel statistical methodologies within the innovative and rapidly expanding framework of Functional Data Analysis (FDA). FDA is a methodological framework designed to process and analyze data that are inherently functional in nature—data observed over a continuous domain, such as time, space, or other dimensions. In this framework, observations are not merely discrete points, like realizations of real-valued random variables, but entire functions, representing realizations of a random function. One of the main challenges in our problem is that both QoE, as the response variable, and the factors influencing it (i.e., network parameters as covariates) are recorded over time. In this context, we are dealing with a regression problem involving a functional response variable and functional covariates. Using the convention where the first term indicates the nature of the response variable and the second term describes the covariates, FDA distinguishes between scalar-on-function, function-on-scalar, and function-on-function regressions. Among these, the function-on-function regression, which we address here, is the least developed in the literature. To study the relationship between QoE, denoted by $\mathrm{Y}(t)$, and the covariates, denoted by $\mathrm{X}(t) = \big(\mathrm{X}^1(t),\ \ldots,\ \mathrm{X}^p(t)\big)$, our initial approach involves considering two types of linear regression models explored in previous work of Ramsay and Silverman (2005, Chapter 12) given in (0.0.1) and (0.0.2). These two models are employed to account for the dynamic nature of QoE, which is influenced by a multitude of factors that can vary significantly from one situation to another. For example, the impact of latency on QoE may depend on the content resolution or the type of codec used. In such cases, a single model may struggle to effectively capture these specificities. The mixture model framework, which partitions the data space into more

homogeneous subspaces, is well-suited to address the problem of heterogeneity. In the predictive modeling context, this leads us to the Mixture-of-Experts (MoE) approach, where each "expert" specializes in predicting QoE within a homogeneous region of the data space, and a gating function determines which expert is most relevant for a specific data point. By combining FDA with the MoE framework, we obtain a robust and flexible model that enables more reliable QoE predictions. For a K components mixture model with $Z = (z_1, \ldots, z_K)$ as class membership variable, the MoE is defined by the (0.0.3) formula.

Remerciements

Au moment de faire le bilan de ces trois années thèse, je me rends compte à quel point elles ont été riches et intenses tant professionnellement que personnellement. Pendant ce parcous, j'ai croisé le chemin de beaucoup de personnes et elles ont contribué de façon plus ou moins active à ma thèse.

C'est une aventure qui a en réalité débuté en 6 mois avant le début de la thèse (en avril 2021), lors de mon stage de fin d'études d'ingénieur et de M2 chez Orange à Lannion. Une période marquée par les restrictions sanitaires et où, résidant à Rennes, je ne devais me rendre à Lannion qu'une seule fois par semaine. Lors de ma première visite, je me souviens être parti de Rennes à 6h du matin, et à mon arrivée autour de 8h, j'ai été accueilli par Vincent sur le quai de la gare à cette heure très matinale. Ce geste très attentionné m'a immédiatement fait comprendre que j'étais entre de bonnes mains.

J'ai donc eu la chance d'être encadré par des personnes certes très occupées, mais toujours disponibles pour m'accompagner. C'est dans ce contexte que je tiens tout d'abord à les remercier ainsi que pour la confiance accordée. Julien qui m'a appris être pragmatique, que l'essentiel n'est pas d'être capable de faire des choses compliquées mais compréhensibles et utiles à répondre une problématique. Stéphane qui a toujours plein d'idées lumineuses, peu importe le problème posé. Merci pour ta capacité à prendre du recul face à mes multiples questions pour me proposer des solutions intéressantes. Guillaume, celui avec qui j'ai passé le plus de temps et qui a dû me supporter, merci pour ta bienveillance. Tu incarnes le type de chercheur que j'aspire à devenir. Vincent dont les qualités humaines et la disponibilité à m'orienter n'ont jamais fait de doutes. A tous les quatre, merci pour cette riche expérience.

Je tiens ensuite à exprimer ma gratitude aux membres de mon comité de suivi de thèse Jairo et Matthieu, pour vos observations constructives et votre regard critique, qui ont grandement contribué à améliorer la qualité de ce travail. Je remercie également Sophie et Faïcel pour avoir accepter de rapporter ma thèse. C'est un immense honneur ! Merci pour votre relecture attentive et les remarques constructives qui m'ont aidé à préparer ma soutenance et à prendre du recul sur mon travail. Je remercie enfin Emilie et Sofiene d'avoir accepté de faire parti de mon jury.

Je tiens également à exprimer toute ma gratitude à ma famille sans qui rien n'aura été possible. Tout d'abord papa parti trop tôt. Je sais que d'où tu es tu veilles toujours sur moi. Maman et papa Gaston pour m'avoir appris à être tenace face aux difficultés de la vie. A mes frères et sœurs Dolores, Séréna, Carmen, Manelle, Luther, Evelyne, Emeraude dont le seul sourire suffit à rendre ma journée agréable. A toute la grande famille, merci. Enfin, Gwladys sur qui je me repose au

quotidien. Un grand merci d'être là.

Merci également à mes ami.e.s et camarades. D'abord, Hassan et Daouda, lorsque nous avons pris la décision de faire une thèse alors qu'une carrière d'ingénieur s'ouvrait à nous on ne savait clairement pas ce qui nous attendais. Mais je suis heureux de l'avoir accompli. Aux amis de longue date, Baudoin, Aurélien, Dieudonné, Wilson malgré la distance et le temps qui passe nous sommes restés toujours aussi proches. Sans oublier les camarades data scientists dont le père de scientisttools Duvérier, Caleb, Rahimatou, Cabrelle, Lucile, Franklin, Danis et tous ceux que je ne cite pas.

Mes remerciements vont également aux collègues doctorants du labo ERIC avec qui nous avons partageons et surmontons les mêmes difficultés au quotidien. Martial (qui a déjà soutenu), Francesco, Eliz, Noé, vu nous partageons le même directeur de thèse, je vous comprend particulièrement. Sinon les autres Simon, Rémi, Irina, Floribert et tous ceux que j'oublie. Je garde un excellent souvenir des déjeuners, des soirées et des conférences passés ensemble.

À vous tous, je dédie cette thèse avec gratitude et reconnaissance.

Contents

Glossary

AI Artificial Intelligence. 1

AR AutoRegressive. 83, 87

DL Deep Learning. 82

FDA Functional Data Analysis. iii, v, vi, 2–4, 6, 11, 18, 50, 78, 79, 82, 83, 87, 89, 90, 92, 93, 101, 102, 109, 110

FFMoE Function-on-Function Mixture-of-Experts. 15, 49, 57, 59, 61–65, 67, 69–72, 74–76, 91–93, 102–107, 109

i.i.d. independent and identically distributed. 8, 11–13, 19, 53

IoT Internet of Things. 1

LTE Long-Term Evolution. 83

ML Machine Learning. 18, 20, 78–80, 93, 96

MNO Mobile Network Operator. 1, 78, 92, 93, 96

MoE Mixture-of-Experts. iv, vi, 14, 49, 51, 55, 59, 60, 70, 76, 90, 92, 93, 102, 103, 106, 109, 110

MOS Mean Opinion Score. 4, 84, 87, 92, 94–96, 101

NN Neural Network. 87, 96

NWDAF Network Data Analytics Function. 93

PenFFR Penalized Function-on-Function Regression. 33, 77–79, 87–89, 91, 103, 106

QoE Quality of Experience. iii–vi, 1, 2, 4, 15, 18, 77–79, 82–93, 96, 103, 105–107, 109, 110, 143, 154

QoS Quality of Service. 1, 4, 18, 78, 86, 92, 96

RF Random Forest. 82, 87–89

VoIP Voice Over IP. iii, v, 2, 4, 15, 91–93, 96, 103, 106, 110

Chapter 1

Introduction

1.1 Scientific and Industrial Context

In the continuous evolution of telecommunications networks, the exponential growth of the internet traffic and the fast development of network services have led to an increasing demand of high network qualities. Mobile Network Operator (MNO) need to differentiate from their competitors by giving particular attention to delivering an optimal end-user experience. This means switching from simply "objective" measurements of network performance or Quality of Service (QoS) parameters to "subjective" measurements of user experience, perception, preference or satisfaction levels simply referred to as Quality of Experience (QoE) parameters. QoE is often expressed in terms of Mean Opinion score (MOS), that is a scalar values ranging from 1 (for very bad) to 5 (for excellent), originally resulting from an average of individual scores in a formal quality subjective test, and today often computed by means of objective models using audio or video signals as inputs. We will widely use this acronym MOS in this document whenever QoE will be addressed. In this context, the ability to accurately predict QoE has become a challenge for network operators. According to the previsions made in 2020 by Cisco Visual Networking Index (Cisco, 2019), they have projected that nearly two-thirds of the global population will have Internet access by 2023 and the number of devices connected to IP networks will be more than three times the global population. With this billions of users and devices connected to the network; faster speeds, connectivity and the multiple technological advancements in the next-generation architecture applications will led to extremely complex requirements and will become more than the norm. Then ensuring an optimal QoE of a service in this complex ecosystem poses multiple challenges. In an environment where user fidelity is closely linked to the perceived QoS, the ability to deliver an optimal QoE confers a distinct competitive advantage. MNO are challenged to continuously refine their QoE prediction models, leveraging advances in Artificial Intelligence (AI), data analytics and network orchestration techniques to remain competitive. With the advent of 5th generation telecommunications networks, edge computing and the Internet of Things (IoT) are making QoE prediction even more relevant. These new technologies introduce new use cases, new performance requirements and new user expectations requiring QoE prediction models that can adapt to various network scenarios. One of the primary challenges in the QoE's prediction framework is the fact that for some use cases, the user experience (i.e. QoE as response variable) and the factors that influence them (i.e. networks parameters as covariates) are recorded over time so we need new methods to handle these data.

Faced this ever increasing volume of data, new tools for exploiting and analyzing them are developed during the past few decades. Functional Data Analysis (FDA) have then become very popular in a constantly growing number of industrial, societal and medical applications. FDA is branch of statistics that deals with data that can be represented as functions. Indeed, data is no longer collected in its traditional form, i.e. a response $Y \in \mathbb{R}$ described by a finite number of covariates $X = (X^1, \ldots, X^p) \in \mathbb{R}^p$. It flexibility in handling complex, high-dimensional, and structured data makes it applicable to a broad range of scientific and practical problems, providing insights that traditional data analysis methods may not be able to unveil. Most notable recent applications encompass, in particular, Healthcare and Medicine (monitoring patient health over time, FMRI data), Environmental Science (temperature or precipitation trends over time), Economics and Finance (evolution of stocks or commodities, modelling consumer behaviour over time), Sports Science (Analyzing athletes' performance data over time or during an event to optimize training and performance), Meteorology (analyzing weather patterns and trends to improve forecasting models), Chemometrics (analyzing spectroscopy data to identify and quantify chemical substances), Genomics and Bioinformatics (analyzing gene expression data over time), Traffic Analysis and Urban Planning. In this context, extension of linear regression to the functional setting has therefore naturally become a major area of research in FDA. The main state-of-the art references for FDA are Ramsay and Silverman (2005), Ramsay et al. (2009), Horváth and Kokoszka (2012), Kokoszka and Reimherr (2017), which provide excellent introduction to FDA. Also Goldsmith et al. (2011) and Morris (2014) provide a broad overview of the methods of functional linear regression. In the functional setting, different types of functional linear regression have been considered, depending on the functional nature of the response and/or at least one of the covariates. Thus, using the convention that first term denotes response-type and second term denotes covariate-type, the following regression models are all the possible options to consider: function-on-scalar, scalar-on-function and function-on-function.

In the context of the time-varying QoE's prediction for streaming video or VoIP based on time-varying networks parameters, we are in the extension of linear regression of a functional response involving functional covariates. Traditional multivariate analysis methods are often inadequate for such data due to the inherent smoothness and continuity of functional data. Therefore, FDA provides a suite of methods and tools tailored to handle this complexity, by leveraging basis expansions and dimension reduction techniques.

This PhD thesis aims to contribute to the field of FDA by proposing solutions to the prediction of functional responses from functional covariates using linear regression, addressing key methodological challenges and proposing innovative solutions. The proposed approaches use the basis function expansion, penalized regression techniques and adapted methods for heterogeneous data. We first present in Section 1.2 a few examples of functional data, then define in Section 1.3 the notion of random function. Extensions of the expectation and covariance concepts is give in Section 1.4. After that we properly describe in Section 1.5 the functional basis expansion with a focus on cubic B-splines and some practical illustrations. Section 1.6 gives a complete state of art on regression with functional data and finally give an overview of our contributions in Section 1.7.

1.2 Examples of Functional Data

Functional data are realizations of random function and involve where each observation is a function rather than a scalar or a vector. Here are some examples of functional data we can present and that we will use in our applications in Chapters 2, 3, 4 and 5 in order to give a clear and precise

idea to people new to the field. In dimension 1, an observation of functional data is a curve or trajectory, in 2-dimensional space, it is an image or ransom surface and in superior dimension, it is more complicated objects. But in this thesis, we only work with curves. Below is the list of functional data used along the thesis

Canadian weather data

Figure 1.1 shows the temperature measurements at weather stations in Canada. There are 35 observations representing stations located in various places in Canada. Each data point represents the temperature recorded by a weather station at a day of year, averaging over the years between 1961 to 1994. The colors indicates the geographic climates of the stations. A classic problem with this dataset is the prediction of the level of daily precipitation at a given weather station by observing these daily temperatures. This is typically the type of problem that FDA seeks to solve.

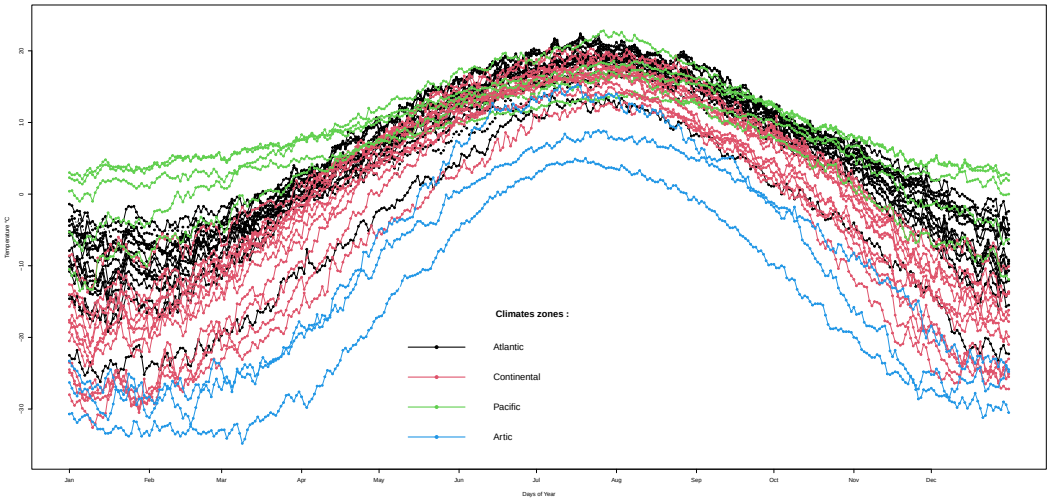
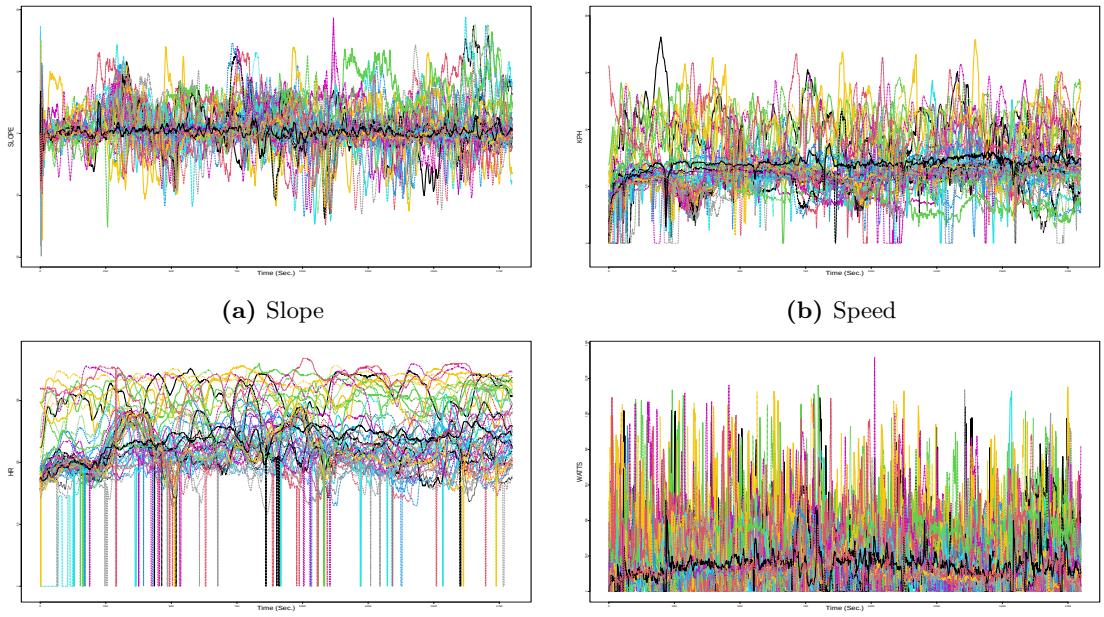


Figure 1.1: Raw temperature profiles of the 35 Canadian weather stations.

Cycling data

Figure 1.2 presents the Cycling data set, produced for Jacques and Samardzic (2022) study and contains the measurements of several parameters during 216 cycling sessions of 30 minutes. The sampling rate is one measure per second. The goal in this study is to predict the developed power according to the three parameters known to have an impact such as heart rate (in beats per minute), the speed (in km/h) and the slope (in percentage).



(a) Slope **(b)** Speed

(c) Heart Rate **(d)** Developped Power

Figure 1.2: Raw curves in cycling data set

Video streaming data

Figure 1.3 displays the continuous Mean Opinion Score (MOS) curves of the LIVE-NFLX-II dataset Bampis et al. (2018b) made at The University of Texas at Austin's LIVE subjective testing lab. The dataset is designed to predict the QoE (through MOS) based on QoS parameters of the Netflix streaming video service in realistic adaptive streaming pipeline model. They show for 420 samples of videos and for each frame (along the x-axis, from the first frame of the video to the last) the perceived MOS on the y-axis. We noticed that all the curves do not have the same length which can be problematic for some models. FDA can perfectly handle this type of data for a problem of prediction of MOS curves.

Voice over IP data

Figure 1.4 presents the MOS curves of the VoIP service given in BigQoE Schwarzmann et al. (2022) dataset. As with LIVE-NFLX-II dataset, the aim is to predict the QoE (through MOS) based on QoS parameters. It is a simulated dataset that generated 50 seconds communications using OMNeT++ library Varga (2001) for mobile users within a 4G network cell under several conditions. They show for 15 samples of conversations the perceived MOS at some time. We noted that the time where we have recorded values depend on the conversation.

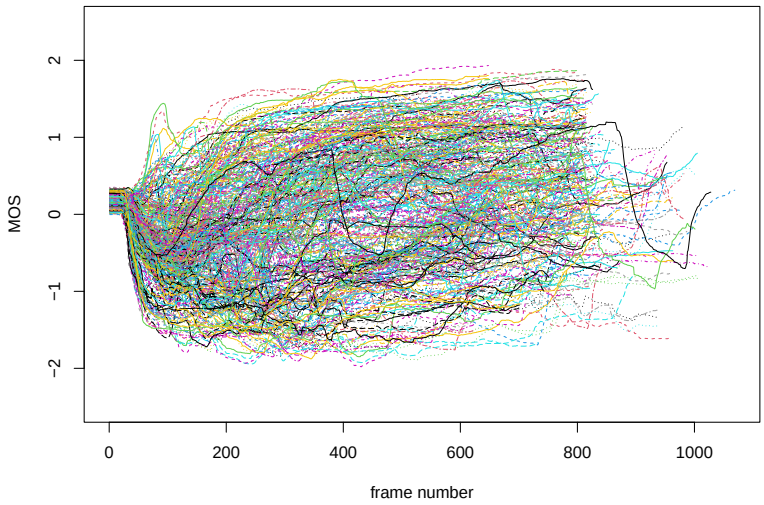


Figure 1.3: Mean Opinion Score (MOS) curves streaming video service

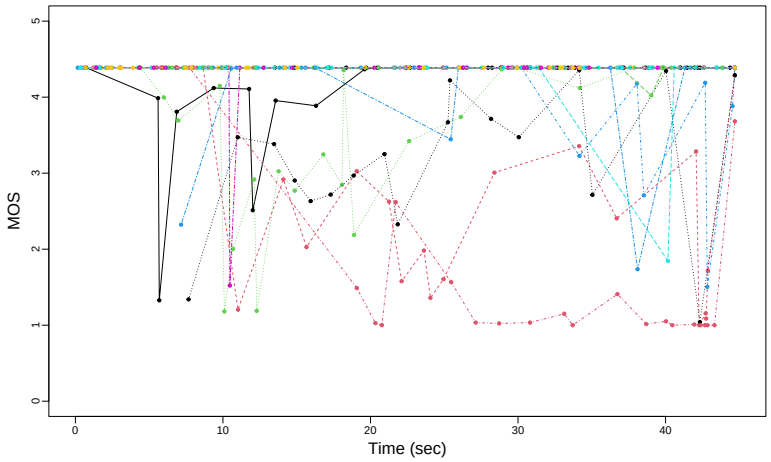


Figure 1.4: Mean Opinion Score (MOS) curves for VoIP service

1.3 Random Functions

Definition 1. *Let $(\Omega, \mathcal{A}, \mathbb{P})$ be a probability space where Ω is a non empty sample space, $\mathcal{A}$ is a σ-algebra of events and $\mathbb{P}$ a probability measure on $\mathcal{A}$. Let also $\mathcal{F}$ be a function space.*
*A **random function** Y is a map defined on $(\Omega, \mathcal{A}, \mathbb{P})$ where for any event $\omega \in \Omega$, $\mathrm{Y}(\omega) \in \mathcal{F}$. i.e. $\mathrm{Y}(\omega)$ is a deterministic function. We consider $\mathcal{F}$ to be the space function with a support $\mathcal{T}$ taking values in $\mathbb{K}$:*

- *If $\mathbb{K} \in \mathbb{R}$, Y is a random curve ;*
- *If $\mathbb{K} \in \mathbb{R}^d$ with $d > 1$, Y is a more complex random process, random surface for example when $d = 2$.*

To sum up, Y can be viewed as a map : $\Omega \times \mathcal{T} \to \mathbb{K}$, such that:

- for all $\omega \in \Omega$ fixed, $\mathrm{Y}(\omega, .) : \mathcal{T} \to \mathbb{K}$ is a trajectory of Y ;
- for all $t \in \mathcal{T}$ fixed, $\mathrm{Y}(., t) : \Omega \to \mathbb{K}$ is a real random variable.

Therefore, what we will call functional data in the following will be a trajectory of the random function. Performing FDA on observations collected in series form means:

- considering each series as functions,
- the existence of an underlying random process of which these observations are realisations which is the fundamental assumption in FDA.

1.4 Extension of expectation and covariance concepts

A crucial step of inference problems is the approximation of the conditional expectation and particularly the regression. In the following, we assume that Y is a random function on $(\Omega, \mathcal{A}, \mathbb{P})$ and all their realizations $\mathrm{Y}(\omega)$ are elements of infinite dimensional space $\mathcal{L}^2([0, \mathrm{T}])$ i.e. the space of all functions whose (absolute) second moment is integrable on $[0, \mathrm{T}]$ taking values in $\mathbb{R}$.
The choice of $\mathcal{L}^2([0, \mathrm{T}])$ is based on the fact that, with its scalar product, it is a separable Hilbert space. A space in which we can state some very useful results.

Definition 2 (Expectation)**.** *Using the Bochner integral, an extension of the Lebesgue integral to functions, the Expectation of Y can be define as :*

$$\mathbb{E}(\mathrm{Y}) := \int_{\Omega} \mathrm{Y}(\omega) d\mathbb{P}(\omega).$$

For all $t \in \mathcal{T}$ fixed, we know that $\mathrm{Y}(., t) := \mathrm{Y}(t)$ is a random variable. So we can also define a less restrictive expectation (pointwisely) as

$$\Big(\mathbb{E}(\mathrm{Y})\Big)(t) = \mathbb{E}(\mathrm{Y}(t)) := \int_{\Omega} \mathrm{Y}(t, \omega) d\mathbb{P}(\omega)$$

The infinite dimension in which we are working requires the definition of an operator called the Covariance Operator for the case of covariance.

Definition 3 (Covariance)**.** *In the context of the Hilbert space* $\mathcal{L}^2\Big([0,\mathrm{T}]\Big)$*, the* ***Covariance operator*** *is defined by*

$$\Gamma : f \in \mathcal{L}^2(\mathrm{T}) \longmapsto \Gamma f = \mathbb{E}\Big[\langle\, \mathrm{Y} - \mathbb{E}[\mathrm{Y}], f\,\rangle\, (\mathrm{Y} - \mathbb{E}[\mathrm{Y}])\Big],$$

where $\langle ., . \rangle$ is the inner product in $\mathcal{L}^2$ and $\|.\|$ the associate norm.

Proposition 1.1. *Let* Y *be a random variable in a separable Hilbert space* H *such that* $\mathbb{E}\Big[\|\mathrm{Y}\|^2\Big] < \infty$*. Then the covariance operator is*

- ***self-adjoint*** *:* $\forall\, f, g \in \mathrm{H}, \langle\, \Gamma f, g\,\rangle = \langle\, f, \Gamma g\,\rangle$ *;*
- ***Hilbert-Schmidt :*** *there exists a hilbertian basis* $(\varphi_j)_{j\in\mathbb{N}}$ *such that* $\sum_{j\in\mathbb{N}} |\Gamma\varphi_j|^2 < \infty$

Proof.

1. Γ **is self-adjoint :**
   Let $f,\, g \in \mathcal{L}^2(\mathrm{T})$ and show that $< \Gamma f,\, g >=< f,\, \Gamma g >$

$$\begin{aligned}
\langle\, \Gamma f,\, g\,\rangle &= \int_{\mathrm{T}} \Gamma f(t)\, g(t) dt = \int_{\mathrm{T}} \Big(\int_{\mathrm{T}} \mathbb{C}_{\mathrm{Y}}(s,t)\, f(s)\, ds \Big)\, g(t) dt \\
&= \int_{\mathrm{T}} \int_{\mathrm{T}} \mathbb{C}_{\mathrm{Y}}(s,t)\, f(s)\, g(t)\, ds\, dt = \int_{\mathrm{T}} \int_{\mathrm{T}} \mathbb{C}_{\mathrm{Y}}(s,t)\, f(s)\, g(t)\, dt\, ds \quad \textit{(Fubini)} \\
&= \int_{\mathrm{T}} f(s) \underbrace{\Big(\int_{\mathrm{T}} \mathbb{C}_{\mathrm{Y}}(s,t)\, g(t)\, dt \Big)}_{\Gamma g(s)}\, ds = \int_{\mathrm{T}} f(s) \Gamma g(s)\, ds = \langle\, f,\, \Gamma g\,\rangle.
\end{aligned}$$

2. Γ **is of Hilbert-Schmidt :**
   We have to find a Hilbertian basis $\{\psi_j\}_{j\in\mathbb{N}}$ of $\mathcal{L}^2(\mathrm{T})$ such that $\sum_{j=1}^{\infty} \|\Gamma\psi_j\|^2 < \infty$

$$\begin{aligned}
\sum_{j=1}^{\infty} \|\Gamma\psi_j\|^2 &= \sum_{j=1}^{\infty} < \Gamma\psi_j,\, \Gamma\psi_j >= \sum_{j=1}^{\infty} \int_{\mathrm{T}} \Big(\Gamma\psi_j(t)\Big)^2\, dt \\
&= \sum_{j=1}^{\infty} \int_{\mathrm{T}} \left(\int_{\mathrm{T}} \mathbb{C}_{\mathrm{Y}}(t,s)\, \psi_j(s)\, ds \right)^2 dt = \sum_{j=1}^{\infty} \int_{\mathrm{T}} \Big(< \mathbb{C}_{\mathrm{Y}}(t,.), \psi_j > \Big)^2\, dt \\
&= \int_{\mathrm{T}} \sum_{j=1}^{\infty} < \mathbb{C}_{\mathrm{Y}}(t,.), \psi_j >^2\, dt = \int_{\mathrm{T}} \|\mathbb{C}_{\mathrm{Y}}(t,.)\|^2\, dt \\
&= \int_{\mathrm{T}} \int_{\mathrm{T}} \mathbb{C}^2_{\mathrm{Y}}(t,s) ds\, dt < \infty
\end{aligned}$$

□

The covariance function of the process Y is the map:

$$\mathbb{C}_{\mathrm{Y}} : (s,t) \in [0,\mathrm{T}]^2 \longmapsto \mathbb{C}_{\mathrm{Y}}(s,t) = \mathbb{E}\left[\Big(\mathrm{Y}(s) - \mathbb{E}[\mathrm{Y}(s)]\Big)\Big(\mathrm{Y}(t) - \mathbb{E}[\mathrm{Y}(t)]\Big)\right]$$

The relation between covariance operator and covariance function is given by :

$$\Gamma f(t) = \int_0^{\mathrm{T}} \mathbb{C}_{\mathrm{Y}}(s,t)\, f(s)\, ds$$

i.e. Γ is an operator of kernel $\mathbb{C}_{\mathrm{Y}}$.

We have thus defined and presented important properties related to the extensions of the notions of expectation and covariance in the case of a random function. However, as we have previously mentioned, the main object Y that we are dealing with here is a process that takes values in an infinite-dimensional space. A prerequisite for the application of this theory is the reduction of dimensionality, which is the subject of the next section.

1.5 Functional basis expansion of functional data

In practice, we do not properly observe continuous curve (trajectory) for each realization of a random function X. We only have access to a set of (noisy) observations at timestamps. As a result, the functional data can be presented as a vector that does not carry all the information on the temporal dynamic. In order to recover the continuous form, which generally belongs to an infinite dimensional space (e.g. Hilbert separable space $L^2([0,T])$), one efficient way to proceed is to expand the considered function in a functional basis via an infinite sum. In this way, we have the advantage that by truncating the series at a given level q, we obtain an approximation of the function $\mathrm{X}_i(t)$ in a q dimensional space. In mathematical terms, given a basis $\{\mathrm{B}_j(t)\}_{j\geq 1}$, the function $\mathrm{X}_i(t)$ will be expressed as:

$$\mathrm{X}_i(t) = \sum_{j=1}^{q} x_{ij}\, \mathrm{B}_j(t),$$

with x_{ij} the basis coefficients. Assume that we have n realizations $\big(\mathrm{X}_i(t)\big)_{1\leq i\leq n}$ whose values on an observation grid are given by:

$$\Big\{(\mathrm{X}_{i1}, t_{i1}),\, (\mathrm{X}_{i2}, t_{i2}),\, \ldots, (\mathrm{X}_{im_i}, t_{im_i})\Big\}_{1\leq i\leq n},$$

where X_{ij} is the value of the curve $\mathrm{X}_i(t)$ at timestamp t_{ij}.

Our main goal is to recover for each i, the trajectory $\mathrm{X}_i(.)$ based on the assumption that: $\mathrm{x}_{ij} = \mathrm{X}_i(t_{ij}) + \varepsilon_{ij}$, with ε_{ij} an unobserved independent and identically distributed (i.i.d.) gaussian noise.

In the particular case of the cubic spline basis (Ramsay and Silverman, 2005), the above expression in functional basis $\{\mathrm{B}_j(t)\}_{j\geq 1}$ truncated at q becomes:

$$\mathrm{X}_i(t) \;=\; \sum_{j=1}^{q} x_{ij}\, \mathrm{B}_j(t) \;=\; x_{i1} + x_{i2}\, t + x_{i3}\, t^2 + x_{i4}\, t^3 + \sum_{j=1}^{q-4} x_{i(4+j)}\, (t-\tau_j)_+^3\,, \qquad (1.5.1)$$

where τ_j are the nodes, $f(t)_+ = \max(f(t), 0)$ the positive part of $f(t)$ and the parameters $(x_{ij})_{0\leq j\leq q-1}$ with respect to the truncated cubic spline basis $\left\{1,\, t,\, t^2,\, t^3,\, (t-\tau_1)_+^3,\, \ldots,\, (t-\tau_{q-4})_+^3\right\}$. Given for any realization i the m_i observations on the discrete grid $t_{i1}, \ldots, t_{im_i}$, we can write:

$$\begin{pmatrix} \mathrm{X}_i(t_{i1}) \\ \mathrm{X}_i(t_{i2}) \\ \vdots \\ \mathrm{X}_i(t_{im_i}) \end{pmatrix} = \begin{pmatrix} x_{i1} + x_{i2}\, t_{i1} + x_{i3}\, t_{i1}^2 + x_{i4}\, t_{i1}^3 + \sum_{l=1}^{q-4} x_{i(4+l)}\, (t_{i1} - \tau_l)_+^3 \\ x_{i1} + x_{i2}\, t_{i2} + x_{i3}\, t_{i2}^2 + x_{i4}\, t_{i2}^3 + \sum_{l=1}^{q-4} x_{i(4+l)}\, (t_{i2} - \tau_l)_+^3 \\ \vdots \\ x_{i1} + x_{i2}\, t_{im_i} + x_{i3}\, t_{im_i}^2 + x_{i4}\, t_{im_i}^3 + \sum_{l=1}^{q-4} x_{i(4+l)}\, (t_{im_i} - \tau_l)_+^3 \end{pmatrix}$$

$$\begin{pmatrix} \mathrm{X}_i(t_{i1}) \\ \mathrm{X}_i(t_{i2}) \\ \vdots \\ \mathrm{X}_i(t_{im_i}) \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & t_{i1} & t_{i1}^2 & t_{i1}^{l^3} & (t_{i1} - \tau_1)_+^3 & (t_{i1} - \tau_2)_+^3 & \ldots & (t_{i1} - \tau_{q-4})_+^3 \\ 1 & t_{i2} & t_{i2}^2 & t_{i2}^{l^3} & (t_{i2} - \tau_1)_+^3 & (t_{i2} - \tau_2)_+^3 & \ldots & (t_{i2}^l - \tau_{q_l-4})_+^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ldots & \vdots \\ 1 & t_{im_i} & t_{im_i}^2 & t_{im_i}^3 & (t_{im}i - \tau_1)_+^3 & (t_{im_i} - \tau_2)_+^3 & \ldots & (t_{im_i} - \tau_{q-4})_+^3 \end{pmatrix}}_{\mathrm{T}_i} \begin{pmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{iq} \end{pmatrix}$$

which we simply write in matrix/vector form as:

$$\mathrm{X}_i = \mathrm{T}_i\, x_i.$$

Therefore, the problem is to recover the functional basis coefficients. In the particular case of cubic splines, it is equivalent to solving the minimization problem:

$$\min_{x_i} \|\mathrm{X}_i - \mathrm{T}_i\, x_i\|^2.$$

One of the main challenges addressed in this section is the one of estimating q, which represents the number of nodes, and their location as well. This problem was previously addressed in Li and Ruppert (2008) and then in Ruppert (2002), where it is shown that:

(i) it is sufficient to choose a large enough number of these nodes to capture the full complexity of the problem,

(ii) imposing a regularity penalty on the optimisation problem limits overfitting.

In Ruppert (2002), it is even further shown that if one chooses a sufficient number of nodes, the order of the spline is no longer crucial in the approximation error.

The regularity penalty that we will impose on the coefficients $(x_{ij})_{j>4}$ at the different nodes will take be of one of the three following possible forms:

$$\textbf{(C1)}\ \max_{5\leq j\leq q} |x_{ij}| < \mathrm{C} \qquad \textbf{(C2)}\ \sum_{j=5}^{q} |x_{ij}| < \mathrm{C} \qquad \textbf{(C3)}\ \sum_{j=5}^{q} x_{ij}^2 < \mathrm{C}.$$

In the present work, we choose Constraint **(C3)**, which is the simplest to implement since it only requires defining the matrix:

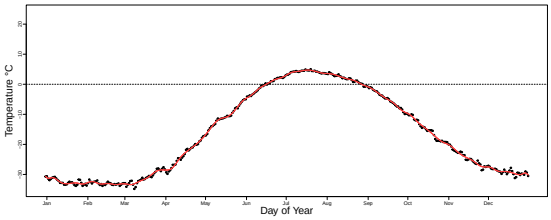
$$\mathrm{D} = \begin{pmatrix} 0_{4\times 4} & 0_{4\times(q-4)} \\ 0_{(q-4)\times 4} & \mathrm{I}_{(q-4)\times(q-4)} \end{pmatrix},$$

where $0_{k\times l}$ is a matrix of size $k \times l$ composed of 0 and $\mathrm{I}_{k\times l}$ the identity matrix. The problem to be solved therefore becomes:

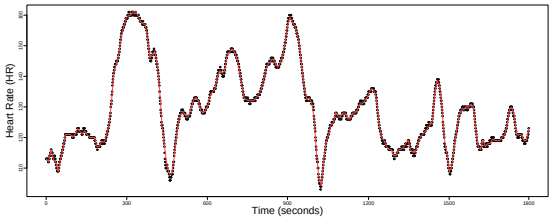
$$\begin{cases} \min\limits_{x_i} \|\mathrm{X}_i - \mathrm{T}_i\, x_i\|^2 \\ \text{s.c.} \quad x_i^\top \mathrm{D}\, x_i \leq \mathrm{C} \end{cases} \iff \min_{x_i} \|\mathrm{X}_i - \mathrm{T}_i\, x_i\|^2 + \lambda_l\, x_i^\top \mathrm{D}\, x_i \qquad (\lambda_l \geq 0), \tag{1.5.2}$$

where the positive integer λ_l is the penalty parameter which controls the amount of regularity of the considered function. The solution to this constrained optimisation problem can be found using the so-called Lagrange multipliers method. The minimization of the least squares criterion (when $\lambda_l = 0$) only ensures that the recovered curve fits the observed data. In practice selecting an appropriate value for the parameter λ_l is of paramount importance for regularity and interpretability of the recovered solution.

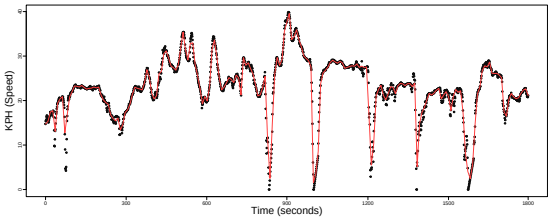
Now we show in Figure 1.5 the results of the smoothing process (red curves) based on raw data (black dots) on datasets presented in Section 1.2. Figure 1.5a give for a Canadian weather station the smoothing temperature curve with $\mathrm{L}_\beta = 150$ basis functions; Figure 1.5b and Figure 1.5c give, for the Cycling dataset, the basis expansion for the heart rate and the speed resp. of a cyclist using $\mathrm{L}_\beta = 200$ B-splines basis functions. And the last one, Figure 1.5d, gives the expansion of a video for the Spatio Temporal - Reduced Reference Entropic Differencing (ST-RRED) scores in the LIVE-NFLX-II dataset. Unless we have more than thousand raw observations, B-splines basis expansion is able to capture the trend of the dynamic evolution.



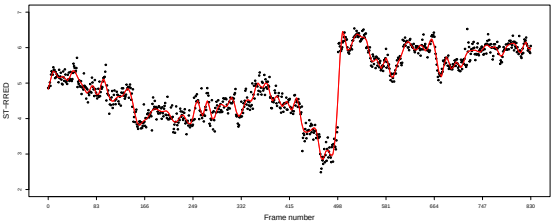
(a) Temperature profile, Canadian weather dataset



(b) Heart rate, Cycling dataset



(c) Speed, Cycling dataset



(d) ST-RRED, LIVE-NFLX-II dataset

Figure 1.5: Raw data (black dots) and functional expansion (red line) of some observations in the presented datasets.

1.6 State of the art of regression with functional data

Functional data are inherently infinite-dimensional, and each observation is represented as a continuous function. The primary goal of FDA is to analyze these functional observations and uncover relationships between them or between functional and scalar covariates. As a result, functional regression has become the most thoroughly researched topic within the broader literature in this framework. Functional regression models extend classical regression approaches to account for the infinite-dimensional nature of functional data. There are several types of functional regression models, depending on the nature of response and predictor variables as given in Ramsay and Silverman (2005), the book that introduced the philosophical thinking behind FDA. We firstly refers readers to Morris (2014) and Wang et al. (2016) for a broad overview of FDA and functional regression.

1.6.1 Scalar-on-function regression

The scalar-on-function regression is the regression where the response variable is scalar and (at least one of) the covariates are functional. The basic form of the model is given by

$$\mathrm{y}_i = \alpha + \int_{\mathcal{I}} \beta(t)\mathrm{X}_i(t)\,dt + \varepsilon_i, \tag{1.6.1}$$

where α is the scalar intercept, $\beta(t)$ the functional parameter, ε_i the i.i.d. errors with mean zero and constant variance σ^2, and $\mathcal{I}$ the domain of the function X.

Estimation of scalar-on-function has been developed in the various domain including Chemometrics Ferraty et al. (2010), cardiology Ratcliffe et al. (2002), climatology Ferraty et al. (2005), etc. The standard techniques include Functional Principal Components Regression (FPCR), (penalized) basis functions based methods, Partial Least Squares (PLS), Bayesian methods and nonparametric approachs. FPCR based methods correspond to reduce the dimensionality of functional predictors by representing them using functional principal components (fPCs). In this decomposition, the covariance structure is analyzed to extract eigenfunctions or Principal Components (PCs). Ramsay and Silverman (2005) use Functional PCA to reduce the functional predictors into finite number of components then traditional linear regression can be performed using these scores as the predictors. Extensions of this method includes regularization techniques (Ridge (Reiss and Ogden, 2007), LASSO (Zhao et al., 2012), smoothing (Silverman, 1996)) to prevent overfitting and impose smoothness on the estimated coefficient function. Kalogridis and Van Aelst (2019) propose a two-step estimation procedure that combines robust functional principal components and robust linear regression.

When the functional predictors are approximated using basis functions, Ramsay and Silverman (2005) introduce an estimator that minimizes a penalized least-squares criterion. Marx and Eilers (1999) proposed an estimator based on penalized splines. Goldsmith et al. (2012) extends the generalized linear mixed model by handling functional predictors; this method is computationally feasible and is applicable when the functional predictors are measured densely, sparsely or with error. For PLS methods, Saricam et al. (2022) propose, to avoid the time-consuming particularity of PLS method on large datasets, two modified functional partial least-squares methods to efficiently estimate the regression coefficients. Gurer et al. (2024) proposed method involves computing the functional partial least squares components through sparse partial robust M-regression Huber and Ronchetti (2011), facilitating robust and locally sparse estimations of the regression coefficient function.

For Bayesian methods, we can mention Crainiceanu and Goldsmith (2010) which have developed tools for functional generalised linear models using WinBUGS. Goldsmith et al. (2014) propose a fast and scalable inferential procedure for estimate the functional coefficient by combining an Ising prior distribution and an intrinsic Gaussian Markov random field. Grollemund et al. (2016) has developed a parsimonious and adaptive decomposition of the coefficient function as a step function and recover periods of time which influence the most the outcome.

Ferraty and Nagy (2021) proposed a method that advocate local linear regression based on a projection as a nonparametric approach to this problem. The asymptotic results demonstrate that functional local linear regression outperforms its functional local constant counterpart. The work in Reiss et al. (2017) reviews the different provided methods for estimating scalar-on-function regression and categorising them in linear, non-linear and non-parametric methods. Goldsmith and Scheipl (2014) proposed a tool to facilitate the comparison and combination of many scalar-on-function estimation methods based on minimizing the cross-validated prediction error of the final estimator. Finally, Cardot et al. (2003) interested in testing the null hypothesis with two test statistics based on the norm of the empirical cross-covariance operator.

1.6.2 Function-on-scalar regression

The function-on-scalar regression is the regression where the response variable is functional and all of the covariates are scalar. The basic form of the model is given by

$$\mathrm{Y}_i(t) = \alpha(t) + \beta(t)\mathrm{x}_i + \varepsilon_i(t), \tag{1.6.2}$$

where $\alpha(t)$ is the scalar intercept function, $\beta(t)$ the parameter for the scalar covariate and $\varepsilon_i(t)$ the i.i.d. errors with mean zero and constant variance σ^2.

Several techniques have been proposed to estimate the time-varying coefficients. These methods often involve smoothing or dimension reduction to address the infinite-dimensional nature of functional data. Greven and Scheipl (2017) proposes a comprehensive framework for additive (mixed) models based on the guiding principle of reframing functional regression in terms of corresponding models for scalar data, allowing the adaptation of a large body of existing methods for these novel tasks. Ramsay and Silverman (2005) proposed to use basis functions, with quadratic roughness penalties applied to avoid overfitting, for fit the model with a penalized ordinary least squares (P-OLS) estimator of the coefficient functions. Reiss et al. (2010) recast the Ramsay and Silverman (2005) estimator as a generalized ridge regression estimator, and present a penalized generalized least squares (P-GLS) alternative. Challenges arise in function-on-scalar regression, particularly regarding the computational complexity of estimating high-dimensional models. Issues like multicolinearity, overfitting or noisy data complicate estimation. To address these, various smoothing techniques and dimensionality reduction methods are employed. Cai et al. (2022) develop a robust variable selection procedure that simultaneously selects relevant predictors and provides estimates for the functional coefficients based on exponential squared loss combined with the group smoothly clipped absolute deviation regularization method. Miao and Wang (2024) propose two types of regularized robust estimation methods: (i) The first based of reproducing kernel Hilbert space, least absolute deviation and group Lasso techniques (ii) The second applies the pre-whitening technique and estimates the error covariance function by using functional principal component analysis. We can also mention the work of Huang et al. (2020) that propose a robust way by incorporating a spatial autoregressive parameter and a spatial weight matrix into the scalar-on-function regression

to accommodate spatial dependencies among individuals and assume a t-distribution assumption for the error terms.

1.6.3 Function-on-function regression

The function-on-function regression is the functional regression that set to our use cases and are the less studied model in the literature compare to the two previous functional regression models. It correspond to the regression where both the response variable and (at least one of) the covariates are functional. The general form of the model is given by

$$\mathrm{Y}_i(t) = \alpha(t) + \int_{\mathcal{I}} \beta(s,t)\mathrm{X}_i(s)\,ds + \varepsilon_i(t), \tag{1.6.3}$$

where $\alpha(t)$ is the functional intercept, $\beta(s,t)$ the functional effect of the covariate at time s to the response at time t; and $\varepsilon(t)$ the i.i.d. errors with mean zero and constant variance σ^2. $\mathcal{I}$ is an interval (window) and can have various form and we can mention three of them as:

- $\mathcal{I} = t$, leads to the concurrent function-on-function model where the response at time t depend on the covariates at the same time t. In this case the covariate effect becomes an univariate function and the model is given by

  $$\mathrm{Y}_i(t) = \alpha(t) + \beta(t)\mathrm{X}_i(t) + \varepsilon_i(t), \tag{1.6.4}$$

  This model is interesting because as shown in Hastie and Tibshirani (1993), also called the varying coefficient model, any functional model can be reduced to this form ;

- $\mathcal{I} = [0, \mathrm{T}]$ correspond to fully integral functional regression model where the response $\mathrm{Y}(t)$ at time t is modeled as a linear function of the entire predictor curve $\mathrm{X}(s)$, for s over the whole domain ;

- $\mathcal{I} = [0, t]$ leads to the historical integral model which assumes that the response at time t depends not only on the predictor at time t but also on the entire past values of the predictor function up to time t. This model is more realistic than the fully functional because we can not future values of covariates that have impact on the past values of response variable.

Ramsay and Silverman (2005) presented Model 1.6.3 and discussed on inference methods using basis functions. Besse and Cardot (1996); Kokoszka and Reimherr (2017) developed penalized approaches with basis functions expansions. Yao et al. (2005a) proposed a fPC based methods by modeling the functional variables using fPC decompositions with i.i.d. measurement errors. Yao et al. (2005b) also proposed the Principal component Analysis through Conditional Expectation (PACE) method to perform fPC analysis for sparse longitudinal data. As mentioned by Crainiceanu et al. (2009), the shape and interpretation of the functional parameter can change dramatically when one includes one or two additional PCs. Instead of PC scores of predictors, Chen and Wang (2011) proposed to used noisy observations on the weighted least squares criterion to account for the correlation within the functions in estimating the regression coefficients. One of issues by using PC scores is the selection of the number components. Ivanescu et al. (2015) extend the penalized functional regression designed by Goldsmith et al. (2011) to the function-on-function framework using PC expansions of the functional predictors while keeping many PCs, and regularizing the functional coefficient using penalized splines. The smoothness of the functional coefficients is controlled by

smoothing parameters, which are estimated using restricted maximum likelihood (REML) in an associated mixed model. Scheipl and Greven (2016) pursues by discussing on the identifiability issues of this model. In the case of historical integral model, i.e. when $s \leq t$ or $\beta(s,t)$ is lower triangular, Malfait and Ramsay (2003) proposed a custom basis functions that are tent-like piecewise linear functions on some grid to ensure the $s \leq t$ constraint. Regularization by imposing roughness penalty is developed by James et al. (2009) that lead to interpretable parameters. Harezlak et al. (2007) considered a variety of regularization techniques for linear B-spline basis functions, including basis truncation, roughness penalties, and sparsity penalties (LASSO). Centofanti et al. (2022) proposed a Smooth plus LASSO (S-LASSO) estimator that is locally sparse (i.e., is zero on the null region) and, at the same time, smooth on the non-null region. They have shown that it is able to increase the interpretability of the model, by better locating regions where the coefficient function is zero, and to smoothly estimate non-zero values of the coefficient function. Finally,Meyer et al. (2015) propose a general Bayesian framework for multi-level functional data fitting by using Monte-Carlo Markov Chain (MCMC) procedure.

The field of function-on-function linear regression is still developing. Despite its utility, the literature on heterogeneous functional data and robust confidence intervals in this context remains sparse, especially when addressing the challenges posed by real-world. In this work, we provide a methods to address these issues by emerging solutions such as conformal prediction for robust confident intervals and MoE for heterogeneous data.

1.7 Outline and contributions of the Thesis

We now present our contributions that following the organization of the manuscript:

- Chapter 2 presents an estimation method of two function-on-function linear regression models using B-splines basis expansion both for functional parameters and covariates. A penalized version of the method is also proposed that used a ridge type penalty on second derivatives of parameters. The second contribution of this chapter is the proposed confident interval generated by a functional conformalized quantile regression that combined a soft version of functional quantile regression and conformalization method that computed functional quantiles using optimal transport. It has been submitted for publication in a journal and has been the subject of several scientific communications. We can mention the short paper presented in *les 53es Journées de la Statistique (jds) de la Société Française de Statistique (SFdS)* Tamo Tchomgui et al. (2022b), May 2022. A Poster presented in spring school *Statlearn: challenging problems in statistical learning* Tamo Tchomgui et al. (2022c), April 2022. A paper accepted as mini-conf in the *20th International Conference on Network and Service Management (CNSM)* Tamo Tchomgui et al. (2024a), October 2024. Finally, the submitted journal paper Tamo Tchomgui et al. (2023a).

- Chapter 3 proposed an extension of MoE regression in a function-on-function linear regression setup that is well suited in a predictive modelling for handling heterogeneous data. An estimation scheme is proposed for concurrent linear function-on-function regression using B-splines basis expansion for gated and experts parameters. We also provide a penalized version of the method that leads to interpretable estimated parameters. It has been published in *Statistics and computing* journal, Vol 34, num 154 Tamo Tchomgui et al. (2024c), June 2024. This work was also the subject of some scientific communications such as the *15th International*

Conference of the ERCIM WG on Computational and Methodological Statistics. 16th International Conference on Computational and Financial Econometrics Tamo Tchomgui et al. (2022a), December 2022. A presentation in the *fda-Lille: Functional Data Analysis Workshop* Tamo Tchomgui et al. (2024b), March 2024. The short paper presented in *les 54es Journées de la Statistique (jds) de la Société Française de Statistique (SFdS)*, Tamo Tchomgui et al. (2023c), July 2023. The presentation in *Royal Statistical Society 2024 International Conference*Jacques and Tamo Tchomgui (2024), September 2024.

- Chapter 4 is an application of our first proposed method presented in Chapter 2 to the video streaming use cases' QoE prediction. The experimental results show that the performance of our prediction is on par for one metric with Deep-Learning-based methods and even better for another metric, while preserving the explicability of the model. This work is published as mini-conf in the *20th International Conference on Network and Service Management (CNSM)* Tamo Tchomgui et al. (2024a), October 2024.

- Chapter 5 is also an application of the second proposed method presented in Chapter 3 to the QoE's prediction of VoIP service. It is not so far a submitted or published work but the obtained result gives satisfactory explanation of the QoE dynamic.

- Appendix A associated to the Chapter 2 gives the values of parameters used in the simulation study (Appendix A.1); the maximum likelihood estimation of mixed model derived from the functional model (Appendix A.2); The visualization of the estimated parameters in the simulated study (Appendix A.3) ; The representation of the estimated parameter gives by the concurrent model on Hawaii ocean Data (Appendix A.4); Finally, Appendix A.5 show the visualization of the prediction gives by the concurrent model on Canadian weather dataset.

- Appendix B associated to the Chapter 3 firstly describe the EM algorithm for the Function-on-Function Mixture-of-Experts (FFMoE) (Appendix B.1); for the penalized version PenFFMoE (Appendix B.2); After that we give graphical visualization of estimated parameter on simulated data (Appendix B.3); Canadian weather data (Appendix B.4); and Cycling data (Appendix B.5).

Chapter 2

A Penalized Spline Estimator for Functional Linear Regression with Functional Response

This chapter is a submitted work in a statistical journal. We have reproduced the article as submitted with few changes in introduction to enhance the context of the manuscript. For this reason, some concepts may have been repeated from the previous chapter such as state of the art. The paper addresses the problem of linear function-on-function regression for concurrent and (historical) integral model in a predictive modelling. For this purpose, we proposed an inference method based on basis expansion of functional covariates and parameters to transform the functional model into a linear mixed model. We provide after that robust confidence interval of our predictions through a conformalized quantile prediction. It is a combination between conformal prediction theory and functional quantile regression.

Abstract: Many scientific studies in recent years have been collecting data at a high frequency and can be considered as functional data. We provide a novel and easy-to-implement method addressing function-on-function linear modelling and obtain interpretable parameters when both the response variable to be modelled and the covariates are functions. Two main types of models are considered: (i) the concurrent model which explains the response curve $Y_i(t)$ at time t from the values at same time t of the covariates $X_i^l(t)$; (ii) the (feed-forward) integral model which explains $Y_i(t)$ based on the values of covariate curves $X_i^l(s)$ observed at any times $s \leq t$. A regularized inference approach is proposed, which accurately selects an appropriate set of basis functions that can be used for functional data reconstruction and at the same time provides smooth and interpretable functional parameters. A functional confidence interval procedure is also proposed which uses the conformalization framework. Numerical studies on simulated data with different scenarios illustrate the good performance of our method to capture the relationship between covariates and response. The method is finally applied to well-known data and compared to a baseline. Our method shows significant improvements on prediction error: on Canadian weather data with the problem of predicting precipitations from temperature measurements and on Hawaii ocean data for predicting ocean salinity from temperature, oxygen, chloropigments and density measurements.
Keywords: Functional Data Analysis, function-on-function regression, penalized splines, Canadian weather data, Hawaii ocean data.

2.1 Introduction

The final goal of this thesis is to provide a Machine Learning (ML) model able to predict the user experience, perception, preference or satisfaction levels simply referred to as Quality of Experience (QoE) of a given service based on "objective" measurements of network performance or QoS parameters. The data for the targeted services are such that we are led into the framework of FDA, more specifically, to models where both covariates and response are functional. As mentioned in Morris (2014), functional regression is the area of FDA hat has received the most attention in applications and methodological development. It is therefore a good approach for exploring the relationship between functional response and functional covariates. Extension of linear regression to the functional setting has therefore naturally become a major area of research in FDA. While the literature is too vast to cover here, the recommended references for this field are Ramsay and Silverman (2005), Ramsay et al. (2009), Horváth and Kokoszka (2012), Kokoszka and Reimherr (2017), which provide excellent introductions to FDA. Moreover Goldsmith et al. (2011) and Morris (2014) provide a broad overview of the methods of functional linear regression. In the functional setting, different types of functional linear regression have been considered, depending on the functional nature of the response and/or at least one of the covariates. Thus, using the convention that first term denotes response-type and second term denotes covariate-type, the following regression models are all the possible options to consider: function-on-scalar, scalar-on-function and function-on-function. The scalar-on-function linear regression models is the most thoroughly studied model among the three models in the current literature. Some references include Cardot et al. (1999) and Hastie and Tibshirani (1993).

Most of the inference approaches for these models rely on a basis expansion assumption. For instance Besse and Cardot (1996) and Ramsay and Silverman (2005) proposed spline-type approximations of the functional covariates and then performed the estimation step by minimizing

a least squares criterion. Among other useful references, Antoch et al. (2010) uses B-spline expansions for both the functional parameters and the functional covariates. The issue of possible non-identifiability was pursued in Scheipl and Greven (2016). In theses approaches, the functional regression models become equivalent to a multivariate model on the basis expansion coefficients. An alternative way is to consider Functional Principal Components Analysis (FPCA, Ramsay and Silverman (2005)), possibly using smoothness promoting penalization (Besse et al., 1997; Silverman, 1996). Possible issues in determining the number of components to account for that seem to be still open. Indeed, it was shown in Crainiceanu et al. (2009) that the shape of the functional parameters can drastically change as one or two additional principal components are included, making the process quite unstable and relatively difficult to interpret.

In comparison with scalar-on-function problems, function-on-function models, that we address here, have been much less studied in the literature. For instance, Ivanescu et al. (2015) proposes via the pffr method to estimate a function-on-function regression model using a penalized mixed model. In this setting as well, the main issue faced is not only the problem of accurately selecting the number of basis functions and the location of the knots (Li and Ruppert, 2008), but also the possible interpretability of the obtained estimators (James et al., 2009). Signal compression approach (*wSigcomp*) designed by Luo et al. (2016) which is another way to address function-on-function models firstly apply wavelets transformation to covariates and with the functional response and the obtained multivariate covariates, proposed a method to estimate the functional bivariate parameter by characterize it as the solution of a generalized functional eigenvalue problem. The Optimal Penalized Function-on-Function Regression (OPFFR) proposed by (Sun et al., 2018), produce an estimator of the 2D functional parameter as optimizer of a form of penalized least squares where the penalty enforces a certain level of smoothness.

In mathematical terms, the problem considered in the present chapter is the one of estimating a linear relationship between functional covariates and functional response based on the n-sample

$$\left\{ \mathrm{Y}_i(t),\ \mathrm{X}_i(t) = \left(\mathrm{X}_i^1(t), \ldots, \mathrm{X}_i^p(t)\right)^\top,\ t \in [0, \mathrm{T}] \right\}$$

$i = 1, \ldots, n$, where the output variable $\mathrm{Y}(t)$ and the p input variables $(\mathrm{X}^l(t))_{1 \leq l \leq p}$ are assumed to belong to the separable Hilbert $\mathrm{L}^2\big([0;\mathrm{T}]\big)$. In the sequel, we focus in particular on the following two functional linear models:

$$\mathrm{Y}_i(t) \;=\; \beta_0(t) + \sum_{l=1}^{p} \beta_l(t)\, \mathrm{X}_i^l(t) + \varepsilon_i(t) \;=\; \big(1,\, \mathrm{X}_i(t)\big)^\top \beta(t) + \varepsilon_i(t)\ , \tag{2.1.1}$$

$$\mathrm{Y}_i(t) \;=\; \gamma_0(t) + \sum_{l=1}^{p} \int_0^t \gamma_l(s,t)\, \mathrm{X}_i^l(s)\, \mathrm{d}s + \varepsilon_i(t) = \gamma_0(t) + \int_0^t \mathrm{X}_i(s)^\top \gamma(s,t)\, \mathrm{d}s + \varepsilon_i(t) \tag{2.1.2}$$

where $\beta(t) = \big(\beta_0(t),\quad \beta_1(t),\quad \ldots,\quad \beta_p(t)\big)^\top$, $\gamma(s,t) = \big(\gamma_0(t),\quad \gamma_1(s,t),\quad \gamma_2(s,t),\quad \ldots,\quad \gamma_p(s,t)\big)^\top$ are the unknown functional parameters and are assumed to be square integrable; $\varepsilon_i(t)$ is the model error and is a sample of centered random variables with variance σ_i^2, specific to the i^{th} individual (see Ramsay and Silverman (2005, Chapter 13)); $\varepsilon_i(t)$ and $\mathrm{X}_i(t)$ are assumed to be uncorrelated. The noise functions $\varepsilon_i(t)$ can be rigorously defined using white noise theory as presented in Hida et al. (1993). In our context, we will only use the fact that when sampled at various times from a finite set $\mathcal{T}$, the vector $(\varepsilon_i(t))_{t \in \mathcal{T}}$ can be expressed as a sum of a vector with i.i.d. components and a vector with prescribed covariance matrix, i.e. a vector with constant components in the simplest case.

Model (2.1.1), known as the "concurrent model" , assumes that the response function at time t, $Y_i(t)$, is explained by covariate functions $X_i^l(t)$, at exactly the same time t, the functional parameters being allowed to vary with t as well. The second model (2.1.2), called the "integral model", represents $Y_i(t)$ using the values of the covariates curves $X_i^l(s)$ for all the observed times $s \leq t$. Clearly, Model (2.1.2) is more general and richer than Model (2.1.1). Exploring the "concurrent model" further at the first step is of great interest because, as mentioned in Hastie and Tibshirani (1993), any functional linear model can be reduced to this form.

In this chapter, we firstly develop an efficient approach named PenFFR (or FFR for the non penalized version) to estimate the functional parameters $\beta(t)$ of the concurrent model (2.1.1) and $\gamma(s,t)$ of the integral model (2.1.2). For this purpose, we use cubic B-spline basis expansion for both functional covariates and functional parameters. We propose a penalized estimator of the corresponding functional basis coefficients. As will be shown below, our approach allows to simply choose equispaced knots and a sufficient number of basis functions to capture the main features of the covariates. Overfitting will be naturally avoided by penalizing roughness via controlling the second derivatives of the functional parameters which are being maximized. Secondly, we propose a method to build confidence intervals of the predictions gives by the method. To achieve this, we draw a functional quantile regression by perturbing the standard linear regression and compute functional quantiles by optimal transport. This functional quantile regression is then combined to the conformalized method to build a functional conformalized quantiles regression. Indeed, confidence intervals are one way to guarantee how good are the predictions we made and it is acknowledged by several authors in the literature. They also help for assess statistical significance, model validation or decision making. That's why in some practical situations the final inference goal is generally the estimation of a confidence interval of the predictions. To build confidence intervals with desirable properties such as distribution-freeness or non-asymptoticity, conformal prediction is paramount in modern ML for providing reliable measures of prediction uncertainty. A comprehensive development of this topic is originally presented in Vovk et al. (2005) and Lei et al. (2016). Angelopoulos and Bates (2023) proposes a comprehensive overview, Tibshirani et al. (2019) present a weighted version which can be used for problems where the test and training covariate distributions differ.

Outline of the chapter. The chapter is organized as follows: Section 2.2 shows how, under the assumption of functional basis expansion, the concurrent and integral models are transformed into mixed matrix models. Section 2.3 details two-step estimation scheme. The first one recovers the functional nature of the covariates, by approximating them into a functional basis. The second step computes the penalized estimation of the functional regression coefficients, which are themselves decomposed in another functional basis. Section 2.4 develops the functional quantile regression model by perturbing linear regression in order to provide conformal predictions. Section 2.5 contains a simulation-based exploration of the method which confirms the efficiency of the proposed approach. Section 2.6 presents an illustration of the method on two real data sets. The first one is the well-known Canadian weather data set, in which the goal is to explain the precipitation as a function of the temperatures in different Canadian cities. The second one is the Hawaii ocean data set in which salinity is explained as a function of four functional covariates. Finally, Section 2.7 concludes the chapter and presents some future works.

2.2 Linear models for function-on-function regression

This section shows how the functional models (2.1.1) and (2.1.2) can be, under the basis expansion assumption of covariates and parameters, reduced to a linear mixed model onto the discrete observations of the functional response and functional covariates.

2.2.1 Functional concurrent model

Linear regression for a functional response involving one or more functional covariates in the concurrent model is a well-known problem. The main issue is to estimate an infinite dimensional parameter $\beta(t)$ through a finite sample of observations. As shown in Hastie and Tibshirani (1993), Model (2.1.1), also called the varying coefficient model, is interesting because any functional model can be reduced to this form. Chapter 14 in Ramsay and Silverman (2005) describes how this model can be fitted by minimizing an unweighted least squares criterion. The method proposed in this chapter addresses the estimation problem using a penalized function-on-function regression as proposed in Ivanescu et al. (2015), where the problem is represented as a mixed model. Nevertheless, our work differs by the choice of the penalization criterion enforced on the functional parameter. The parameter $\beta(t)$ is expanded in functional basis using q_β basis functions to get back to a classical mixed model for which the estimations of the parameters are well known. Furthermore, we allow to choose the number of basis functions q_β to be large enough to capture any desired variations of $\beta(t)$, and we add a roughness penalty term to get a smooth solution for the parameter at the end. As a first step of our modelling, we recover the underlying functional process, by using penalized cubic B-splines expansion for all the functional covariates.

Functional basis expansion of covariates and model parameters

In practice, we do not properly observe a continuous curve for each realization of both the response variable $\mathrm{Y}_i(t)$ and the covariate variables $\left(\mathrm{X}_i^l(t)\right)_{1\leq l\leq p}$. Indeed, as opposed to the ideal observation setting, we only have access to a set of noisy observations at a finite number of points on a grid. As a result, the functional data can be presented as a numerical vector. In order to recover the continuous form, which generally belongs to an infinite dimensional space (e.g. Hilbert separable space $L^2([0,T])$), one efficient way to proceed is by expanding the considered functions in a functional basis. The functional response, which is assumed in model (2.1.1) even in model (2.1.2) to be written as a linear combination of these predictors, is not necessary to be pre-processed. The advantage of this approach is that by truncating the series at a given level q_l, we obtain an approximation of the covariate function $\mathrm{X}_i^l(t)$ in a q_l dimensional space.

So for all the p covariates $\mathrm{X}^l(t)$, we can therefore recover a representation in cubic B-splines functional basis. As indicated by Li and Ruppert (2008), the choice of the number of knots depends on the complexity of the variable and should be large enough to capture the patterns of the variable. It is reasonable to suppose that this number and, thus, the number of basis functions depends on the covariate. So to distinguish the basis functions of each covariate, although they just differ by their number, we will adopt in the rest of this article the system $\left\{\mathrm{B}_1^l(t),\ \mathrm{B}_2^l(t),\ \ldots,\ \mathrm{B}_{q_{\mathrm{X}^l}}^l(t)\right\}$ as the basis function of $\mathrm{X}^l(t)$. Then, any functional covariate can be written as:

$$\mathrm{X}_i^l(t) \quad = \quad \sum_{j=1}^{q_{\mathrm{x}^l}} x_{ij}^l\, \mathrm{B}_j^l(t) = \mathrm{B}^l(t)^\top\, x_i^l \qquad \text{with } 1\leq l\leq p. \tag{2.2.1}$$

Knowing the basis functions $\mathrm{B}_j^l(t)$, the estimation of coefficients x_{ij}^l is done as a preliminary step Li and Ruppert (2008); Ramsay and Silverman (2005); Ruppert (2002).

Similarly as for functional covariates, we expand all the functional parameters $\big(\beta_l(t)\big)_l$ of the concurrent model in functional basis. The number of basis functions $q_{_{\beta^l}}$ must be chosen as sufficiently large to capture the patterns of any $\beta_l(t)$:

$$\beta_l(t) = \sum_{j=1}^{q_{\beta^l}} b_j^l \, \phi_j^l(t) = \phi^l(t)^\top \, b^l \qquad \text{with } \; 0 \leq l \leq p. \tag{2.2.2}$$

Using the expressions (2.2.1) and (2.2.2), the components in Model **(2.1.1)** become:

$$\beta(t) = \begin{pmatrix} \beta_0(t) \\ \beta_1(t) \\ \vdots \\ \beta_p(t) \end{pmatrix} = \begin{pmatrix} \phi^0(t)^\top \, b^0 \\ \phi^1(t)^\top \, b^1 \\ \vdots \\ \phi^p(t)^\top \, b^p \end{pmatrix} = \underbrace{\begin{pmatrix} \phi^0(t)^\top & 0 & \dots & 0 \\ 0 & \phi^1(t)^\top & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \phi^p(t)^\top \end{pmatrix}}_{(p+1,\, \sum_l q_{\beta^l}) \text{ - matrix}} \underbrace{\begin{pmatrix} b^0 \\ b^1 \\ \vdots \\ b^p \end{pmatrix}}_{\sum_l q_{\beta^l} \text{ - vect.}} = \Phi(t)\, b \,,$$

and

$$\mathrm{X}_i(t) = \begin{pmatrix} 1 \\ \mathrm{X}_i^1(t) \\ \vdots \\ \mathrm{X}_i^p(t) \end{pmatrix} = \begin{pmatrix} 1 \\ \mathrm{B}^1(t)^\top \, x_i^1 \\ \vdots \\ \mathrm{B}^p(t)^\top \, x_i^p \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & \mathrm{B}^1(t)^\top & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathrm{B}^p(t)^\top \end{pmatrix}}_{(p+1,\, \sum_l q_{\mathrm{X}^l}) \text{ - matrix}} \underbrace{\begin{pmatrix} 1 \\ x_i^1 \\ \vdots \\ x_i^p \end{pmatrix}}_{\sum_l q_{\mathrm{X}^l} \text{ - vect.}} = \mathrm{B}(t)\, x_i.$$

By plugging-in these expressions into Model (2.1.1), we get:

$$\mathrm{Y}_i(t) \quad = \quad x_i^\top \mathrm{B}(t)^\top \Phi(t) \; b + \varepsilon_i(t) = \mathrm{R}_i(t)^\top b + \varepsilon_i(t) \tag{2.2.3}$$

with $\mathrm{R}_i(t) = \Phi(t)^\top \, \mathrm{B}(t)\, x_i$ which is used as design matrix and b the unknown parameters to be estimated.

Functional concurrent model on the observations

The concurrent model implicitly assumes that the functional covariates and the functional response are observed at the same timestamps. The observation grid will consist of m points $\{t_1, \dots, t_m\}$. In mathematical terms we have:

$$\mathrm{Y}_i(t_j) = \mathrm{R}_i(t_j)^\top b + \varepsilon_i(t_j) \quad \text{ with } \; 1 \leq i \leq n \text{ and } 1 \leq j \leq m. \tag{2.2.4}$$

One very specific issue to take care of is that the successive values of the observation noise $\varepsilon_i(t_1), \dots, \varepsilon_i(t_m)$ can not be assumed independent. As the functional response is measured at m different times for the same individual, the correlation between these different measures can no longer be neglected, which means that the standard assumptions of linear regression cannot be optimal here. Indeed, suppose we, in the simplest case, ignore this correlation and consider each

point of this type of data as an independent observation. In that case, the model has the advantage of being simple to explain, but the main drawbacks are that it is highly dependent on the training sample while our interest is in the whole population.

One way to address the question of dependency is to use a Linear Mixed Model (LMM) (Wood, 2006). We thus assume that the model error can be decomposed as $\varepsilon_i(t_j) = \mathrm{U}_i + \eta_{ij}$, with η_{ij} a Gaussian white noise and U_i a random variable which takes into account the random effect in each individual $i = 1, \ldots, n$. To summarize, our model consists of a LMM with fixed effects b and random effect U_i. In matrix form we get:

$$\mathrm{Y} \quad = \mathrm{R}^\top b + \mathrm{Z}\,\mathrm{U} + \eta, \tag{2.2.5}$$

where $\mathrm{Y} = \Big(\mathrm{Y}_1(t_1), \ldots, \mathrm{Y}_1(t_m), \mathrm{Y}_2(t_1), \ldots, \mathrm{Y}_n(t_m)\Big)^\top$, $\mathrm{R} = \Big(\mathrm{R}_i(t_j)\Big)_{i,j}$ the design matrix of dimension $q_\beta \times nm$ with $q_\beta = \sum_l q_{\beta^l}$, $\mathrm{U} = \Big(\mathrm{U}_1, \mathrm{U}_2, \cdots, \mathrm{U}_n\Big)^\top \sim \mathcal{N}(\mathbf{0}, \Gamma)$, $\eta = (\eta_{ij})_{i,j} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbb{I}_{nm})$ and

$$\mathrm{Z} = \underbrace{\begin{pmatrix} 1_{m\times 1} & 0_{m\times 1} & \ldots & 0_{m\times 1} \\ 0_{m\times 1} & 1_{m\times 1} & \ldots & 0_{m\times 1} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{m\times 1} & 0_{m\times 1} & \ldots & 1_{m\times 1} \end{pmatrix}}_{(nm\times n)\text{ - matrix}}.$$

The specific notations we used are the matrices $0_{k\times l}$ and $1_{k\times l}$ of size $k \times l$, which are composed of zeros and ones, respectively; The notations $\mathbf{0}$ refers to the corresponding null vector and Γ the unknown covariance matrix of the random effects.
The parameters are then the fixed effects vectors b and the variance components σ^2 and Γ. We describe how to perform the inference in Section 2.3.

2.2.2 Functional integral model

The integral Model (2.1.2) assumes cumulative effects of covariates. More clearly, the model we propose uses observations of covariates until time t to predict the response at time t. It is important to note that in most models found in the literature Horváth and Kokoszka (2012); Ramsay and Silverman (2005), when both covariates and response have the same domain, consider that the response at any time t depends on the influence of the covariates on the whole domain. Such model implicitly assumes that the covariates at any time $(t+s)$ can influence the response variable at time t. However, in the integral model, the functional parameters are bivariate functions $\gamma_l(s,t)$, except for the constant of the model, which remains univariate. In this section, we start by expanding the parameters in a finite-dimensional functional basis and then plug this expression into the model.

Functional basis expansion of covariates and model parameters

The functional parameters are therefore expanded in a bivariate basis which may or may not have the same number of basis functions on each of the two dimensions. Without loss of generality and for the sake of simplicity, we assume that the number of basis functions is the same in the two

dimensions. This leads to the following expression:

$$\gamma_l(t,s) = \sum_{j,k=1}^{q_{\gamma^l}} a^l_{jk}\, \mathrm{B}^l_{1j}(t)\, \mathrm{B}^l_{2k}(s) \tag{2.2.6}$$

where $\left\{\mathrm{B}^l_{1j}(t)\right\}_{1\leq j\leq q_{\gamma^l}}$ and $\left\{\mathrm{B}^l_{2j}(t)\right\}_{1\leq j\leq q_{\gamma^l}}$ are the basis functions and $(a^l_{jk})_{1\leq j,k\leq q_{\gamma^l}}$ the unknown basis coefficients to be estimated. We can rewrite this expression in matrix form by:

$$\gamma_l(t,s) \quad = \quad {a^l}^\top\, \mathbf{B}^l_1(t)\, \mathrm{B}^l_2(s) \tag{2.2.7}$$

with

$$\begin{aligned}
a^l &= \left(a^l_{11} \quad \dots \quad a^l_{1q_{\gamma^l}} \quad a^l_{21} \quad \dots \quad \dots \quad a^l_{q_{\gamma^l}1} \quad \dots \quad a^l_{q_{\gamma^l}q_{\gamma^l}}\right)^\top,\\
\mathbf{B}^l_1(t) &= \mathbf{diag}\left(\mathrm{B}^l_{11}(t), \dots, \mathrm{B}^l_{1q_{\gamma^l}}(t), \dots, \dots, \mathrm{B}^l_{11}(t), \dots, \mathrm{B}^l_{1q_{\gamma^l}}(t)\right),\\
\mathrm{B}^l_2(s) &= \left(\mathrm{B}^l_{21}(s) \quad \dots \quad \mathrm{B}^l_{21}(s) \quad \dots \quad \dots \quad \mathrm{B}^l_{2q_{\gamma^l}}(s) \quad \dots \quad \mathrm{B}^l_{2q_{\gamma^l}}(s)\right)^\top.
\end{aligned}$$

The functional constant being univariate, it can thus be written as in (2.2.2) in the form:

$$\gamma_0(t) = \sum_{j=1}^{q_{\gamma^0}} a^0_j\, \mathrm{B}^0_j(t) = \mathrm{B}^0(t)^\top a^0.$$

Functional integral model on the observations

By plugging covariates and parameters functional basis expansion in the integral Model (2.1.2), we get:

$$\begin{aligned}
\mathrm{Y}_i(t) &= \gamma_0(t) + \sum_{l=1}^{p} \int_0^t {x^l_i}^\top \mathrm{B}^l(s)\, \mathrm{B}^l_2(s)^\top\, \mathbf{B}^l_1(t)^\top\, a^l\, \mathrm{d}s + \varepsilon_i(t)\\
&= \gamma_0(t) + \sum_{l=1}^{p} {x^l_i}^\top \underbrace{\left(\int_0^t \mathrm{B}^l(s)\, \mathrm{B}^l_2(s)^\top\, \mathrm{d}s\right)}_{\mathbf{B}^l_2(t)} \mathbf{B}^l_1(t)^\top\, a^l + \varepsilon_i(t)\\
&= \gamma_0(t) + \sum_{l=1}^{p} {x^l_i}^\top\, \mathbf{B}^l_2(t)\, \mathbf{B}^l_1(t)\, a^l + \varepsilon_i(t)\\
&= \mathrm{B}^0(t)^\top a^0 + \sum_{l=1}^{p} \mathrm{Q}^l_i(t)^\top a^l + \varepsilon_i(t)\,,
\end{aligned}$$

with $\mathrm{Q}^l_i(t) = \mathbf{B}^l_1(t)^\top\, \mathbf{B}^l_2(t)^\top\, x^l_i$. Finally we obtain:

$$\mathrm{Y}_i(t) \quad = \quad \mathrm{Q}_i(t)^\top a + \varepsilon_i(t) \tag{2.2.8}$$

with $a = (a^0, a^1, a^2, \dots, a^p)^\top$ and $\mathrm{Q}_i(t) = \left(\mathrm{B}_0(t)^\top, \mathrm{Q}^1_i(t)^\top, \mathrm{Q}^2_i(t)^\top, \dots, \mathrm{Q}^p_i(t)^\top\right)^\top$ two vectors of length $q_\gamma = q_{\gamma^0} + \sum_{l=1}^{p} q^2_{\gamma^l}$.

Once again, we are faced with the problem of lack of independence of the different measured values for the same individual. We will proceed exactly in the same way as with the concurrent model using a linear mixed model with fixed effects given by the vector a and random effects given by the random vector $\mathrm{U} = (\mathrm{U}_i)_i$. The model will therefore be written as a LMM given by:

$$\mathrm{Y} \quad = \mathrm{Q}^\top a + \mathrm{Z}\,\mathrm{U} + \eta, \tag{2.2.9}$$

with Z, U and η define similarly to (2.2.5). $\mathrm{Q} = \Big(\mathrm{Q}_i(t_j)\Big)_{i,j}$ the design matrix of dimension $q_\gamma \times nm$. As in the concurrent model, the parameters we need to estimate are the fixed effects vectors a and the variance components σ^2 and Γ. The inference scheme is described in Section 2.3.

2.3 B-spline-based penalized estimator

In both the concurrent and the integral models presented in Section 2.2.1 and Section 2.2.2 respectively, we have used the decomposition of the infinite-dimensional functional covariates and parameters into a truncated functional basis depending on the chosen number of basis functions. These values naturally needed to be correctly selected in order to avoid over- or under-fitting. Nevertheless, precise adjustment of these values often induces a high computational effort. In the case of the B-spline basis, even more parameters have to be properly tuned such as the choice of the spline order and the location of the knots. In order to reduce the expected cost of such a computationally demanding procedure, we made the choice of choosing a sufficiently large a priori value for q_β (or q_γ) and then apply a roughness penalty. This approach brings the benefit of reducing the overall computational cost, and of possibly improving the interpretability of the estimated functional coefficients. This last point is very interesting in the case of the linear model because as we already know, the interpretation of the predictors-response relationship becomes more difficult when the shape of the functional parameter β (or γ) does not have a simple structure.

Various approaches to regularize the parameter shape have been proposed in the literature. In our setting of interest, the main idea is to enhance the model performance and interpretability by adding a roughness penalty. The work of Leurgans et al. (1993) is among the first to explore the functional penalization and shows that the obtained estimator $\hat{\beta}(t)$ (resp. $\hat{\gamma}(s,t)$) becomes less sensitive to the rather subjective choice of the number of basis functions q_β (resp. q_γ). More recently, James et al. (2009) proposed a method called Functional Linear Regression That is Interpretable (FLiRTI) which addresses the issue of choosing relevant penalties. Based on variable selection ideas such as the Lasso penalty, they produce accurate, flexible and highly interpretable estimates of the functional parameters. The main idea in James et al. (2009) is, instead of enforcing sparsity on the function themselves, to enforce sparsity of the derivatives instead. Using the notation $\beta^{(l)}(t)$ for the l^{th} derivative of $\beta(t)$, we may deduce that $\beta^{(0)}(t) = 0$ guarantees $\mathrm{X}(t)$ has no effect on $\mathrm{Y}(t)$ at t; $\beta^{(1)}(t) = 0$ implies that $\beta(t)$ is constant at t; $\beta^{(2)}(t) = 0$ means that $\beta(t)$ is linear at t and so on. The FLiRTI approach also combine sparsity enforcing penalties for more than one derivative at a time, which can be useful for smooth parameters that may even vanish on some intervals.

Instead of the Lasso penalty applied in the FLiRTI method, where choosing the derivatives remains a difficult computational issue, our approach uses a Ridge penalty on the second derivative of the functional parameters. The choice of penalizing the second derivative is mainly motivated by the desire to obtain a possibly locally linear relationship if needed. Moreover, the use of the Ridge penalty is motivated by the lack of exact sparsity observed in real problems and the clear benefits of getting a closed form formula for the estimators.

2.3.1 Penalized estimator for the concurrent model

Let us first consider the concurrent model in the classical mixed model form as in (2.2.5). In order to obtain an interpretable estimator, we use a roughness penalty in the form of a Ridge-type penalty on the second derivatives, as advocated for in the previous paragraph. The objective function is the penalized log-likelihood function given by

$$\mathcal{L}_{pen}(b,\ \Gamma) \quad = \quad -2\,\mathcal{L}(b,\ \Gamma\,|\,\mathrm{Y}) \ + \ \sum_{l=0}^{p} \mathrm{Pen}(\beta_l)\,; \tag{2.3.1}$$

with,

$$\mathcal{L}(b,\ \Gamma\,|\,\mathrm{Y}) \quad = \quad nm\log(2\pi) + \log|\mathrm{V}| + (\mathrm{Y}-\mathrm{R}^\top b)^\top \mathrm{V}^{-1}(\mathrm{Y}-\mathrm{R}^\top b) \tag{2.3.2}$$

using that $\mathrm{V} = \mathrm{Var}(\mathrm{ZU}+\eta)$ and the penalty

$$\begin{aligned} \mathrm{Pen}(\beta_l) \quad &= \quad \lambda_l \int \beta_l^{''}(t)^2\,dt = \lambda_l \int \left[\sum_{j=1}^{q_{\beta^l}} b_j^l \phi_j^{l"}(t)\right]^2 dt = \lambda_l \sum_{s,k=1}^{q_{\beta^l}} b_s^l b_k^l \Phi_{sk}^l, \\ &\text{with} \quad \Phi_{sk}^l = \int \phi_s^{l"}(t)\,\phi_k^{l"}(t)\,dt. \end{aligned}$$

When λ_l is too large, the estimation of $\beta_l(.)$ is too smooth, and we are not able to account for the possible variations of the regression coefficients. When, instead, λ_l is too small, the estimators might become too rough and overfitting might occur.
For a given value λ_l, the estimation of $\big(\beta_l(t)\big)_{0\le l\le p}$ is obtained by solving:

$$\begin{aligned} \min_{b,\,\Gamma} \mathcal{L}_{pen}(b,\ \Gamma) \quad &= \quad \min_{b,\,\Gamma} -2\,\mathcal{L}(b,\ \Gamma\,|\,\mathrm{Y}) \ + \ \sum_{l=0}^{p} \lambda_l \sum_{s,k=1}^{q_{\beta^l}} b_s^l b_k^l \Phi_{sk}^l \\ &= \quad \min_{b,\,\Gamma} -2\,\mathcal{L}(b,\ \Gamma\,|\,\mathrm{Y}) \ + \ b^\top(\lambda\mathrm{P})\,b\,, \end{aligned} \tag{2.3.3}$$

where $\lambda\mathrm{P} \in \mathbb{R}^{q_\beta \times q_\beta}$ is given by:

$$\lambda\mathrm{P} = \begin{pmatrix} \lambda_0\Psi^0 & 0_{q_{\beta^0}\times q_{\beta^1}} & \dots & 0_{q_{\beta^0}\times q_{\beta^p}} \\ 0_{q_{\beta^1}\times q_{\beta^0}} & \lambda_1\Psi^1 & \dots & 0_{q_{\beta^1}\times q_{\beta^p}} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{q_{\beta^p}\times q_{\beta^0}} & 0_{q_{\beta^p}\times q_{\beta^1}} & \dots & \lambda_p\Psi^p \end{pmatrix} \quad \text{with } \Psi^l = \begin{pmatrix} \Phi_{11}^l & \Phi_{12}^l & \dots & \Phi_{1q_{\beta^l}}^l \\ \Phi_{21}^l & \Phi_{22}^l & \dots & \Phi_{2q_{\beta^l}}^l \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{q_{\beta^l}1}^l & \Phi_{q_{\beta^l}2}^l & \dots & \Phi_{q_{\beta^l}q_{\beta^l}}^l \end{pmatrix}.$$

Here, $0_{q_1\times q_2}$ is the standard notation for the null matrix of size $q_1 \times q_2$. As Ψ^l is a symmetric positive-definite matrix for any $0 \le l \le p$, we can easily find its Cholesky decomposition, which can be efficiently leveraged in the implementation.

We first rewrite Model (2.2.5) in the form :

$$\mathrm{Y} = \mathrm{R}^\top b + \varepsilon^*,$$

with $\varepsilon^* = \mathrm{Z}\,\mathrm{U} + \eta$ and $\mathrm{V} = \mathrm{Var}(\varepsilon^*) = \mathrm{Z}\,\Gamma\,\mathrm{Z}^\top + \sigma^2\mathrm{I}$. By setting the partial derivatives with respect to b and V to 0 and then solving the resulting linear system, we get:

$$\widehat{b}_\lambda \quad = \quad \left(\mathrm{R}^\top \hat{\mathrm{V}}^{-1}\mathrm{R} + \lambda\mathrm{P}\right)^{-1}\mathrm{R}^\top\hat{\mathrm{V}}^{-1}\mathrm{Y}. \tag{2.3.4}$$

(see Appendix A.2 for more details).
Let us now address the problem of choosing the smoothing parameters $\lambda = (\lambda_l)_{0\le l\le p}$. The correct choice will make great use of the observed accuracy of the prediction. For this purpose, for a fixed value of λ, we resort to a leave-one-out cross-validation type approach and compute $\widehat{b}_\lambda^{(-i)}$ based on the sample except for the i^{th} observation. We then compute the prediction $\widehat{\mathrm{Y}}_\lambda^{(-i)}$ at observation i. Finally, we can compute the prediction error or cross-validation score associated with the parameter λ as

$$\mathcal{V}_n(\lambda) = \frac{1}{n}\sum_{i=1}^{n}\left(\mathrm{Y}_i - \widehat{\mathrm{Y}}_\lambda^{(-i)}\right)^2.$$

The value of λ that achieves the lowest estimated risk will be selected.

2.3.2 Penalized estimator for the integral model

For the integral model, we can also optimize the penalized log-likelihood function as in (2.3.1). However, the main difference lies in the specific form of the penalty. The log-likelihood of the model will thus have the expression:

$$\mathcal{L}(a, \Gamma\,|\,\mathrm{Y}) \quad = \quad nm\log(2\pi) + \log|\mathrm{V}| + (\mathrm{Y} - \mathrm{Q}^\top a)^\top\mathrm{V}^{-1}(\mathrm{Y} - \mathrm{Q}^\top a) \tag{2.3.5}$$

using $\mathrm{V} = \mathrm{Var}(\mathrm{ZU} + \eta)$, and the penalized log-likelihood:

$$\mathcal{L}_{pen}(a,\ \Gamma\,|\,\mathrm{Y}) \quad = \quad -2\,\mathcal{L}(a,\ \Gamma\,|\,\mathrm{Y}) \ +\ \mathrm{Pen}(\gamma_0) \ +\ \sum_{l=1}^{p}\mathrm{Pen}(\gamma_l),; \tag{2.3.6}$$

For this model, the parameters γ_l will be bivariate functions except for γ_0, which is univariate. The penalties for bivariate parameters will take the following expression:

$$\mathrm{Pen}(\gamma_l) = \lambda_l\int\int\left\|\mathbf{H}_{\gamma_l}(t,s)\right\|^2 ds\,dt = \lambda_l\int\int\left\|\begin{bmatrix}\frac{\partial^2\gamma_l(t,s)}{\partial t^2} & \frac{\partial^2\gamma_l(t,s)}{\partial t\,\partial s}\\ \frac{\partial^2\gamma_l(t,s)}{\partial s\,\partial t} & \frac{\partial^2\gamma_l(t,s)}{\partial s^2}\end{bmatrix}\right\|^2 ds\,dt.$$

Here $\mathbf{H}_f(t,s)$ denotes the Hessian matrix of the bivariate function f and $\|\cdot\|$ is the standard Frobenius norm. To simplify expressions we will use the notation: $\dfrac{\partial^2\gamma_l(t,s)}{\partial t\,\partial s} \equiv \gamma_l^{ts}(t,s)$, and then we have

$$\mathrm{Pen}(\gamma_l) \quad = \quad \lambda_l\int\int\left(\gamma_l^{tt}(t,s)^2 + 2\,\gamma_l^{ts}(t,s)^2 + \gamma_l^{ss}(t,s)^2\right)ds\,dt. \tag{2.3.7}$$

We know from (2.2.7) that

$$\begin{aligned}\gamma_l(t,s)^2 &= \Big(a^{l\top}\mathbf{B}_1(t)\mathrm{B}_2(s)\Big)^2 = \left(\sum_{i,j=1}^{q_{\gamma^l}} a_{ij}^l \mathbf{B}_{1i}^l(t)\mathrm{B}_{2j}^l(s)\right)^2 \\ &= \sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l \mathbf{B}_{1i}^l(t)\mathbf{B}_{1k}^l(t)\,\mathrm{B}_{2j}^l(s)\mathrm{B}_{2m}^l(s),\end{aligned}$$

so we then have the following expressions for the partial derivatives:

$$\begin{cases}\displaystyle\int\int \gamma_l^{tt}(t,s)^2\,ds\,dt &= \displaystyle\sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l \int \mathbf{B}_{1i}^{l''}(t)\mathbf{B}_{1k}^{l''}(t)\,dt \int \mathrm{B}_{2j}^l(s)\mathrm{B}_{2m}^l(s)\,ds \\ &= \displaystyle\sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l\,\Phi_{1,ik}^{l''}\,\Phi_{2,jm}^l\,; \\ \displaystyle\int\int \gamma_l^{ts}(t,s)^2\,ds\,dt &= \displaystyle\sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l \int \mathbf{B}_{1i}^{l'}(t)\mathbf{B}_{1k}^{l'}(t)\,dt \int \mathrm{B}_{2j}^{l'}(s)\mathrm{B}_{2m}^{l'}(s)\,ds \\ &= \displaystyle\sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l\,\Phi_{1,ik}^{l'}\,\Phi_{2,jm}^{l'}\,; \\ \displaystyle\int\int \gamma_l^{ss}(t,s)^2\,ds\,dt &= \displaystyle\sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l \int \mathbf{B}_{1i}^{l}(t)\mathbf{B}_{1k}^{l}(t)\,dt \int \mathrm{B}_{2j}^{l''}(s)\mathrm{B}_{2m}^{l''}(s)\,ds \\ &= \displaystyle\sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l\,\Phi_{1,ik}^{l}\,\Phi_{2,jm}^{l''}\,.\end{cases}$$

with the notation $\Phi_{u,sk}^{l'} = \displaystyle\int \mathrm{B}_{us}^{l'}(t)\mathrm{B}_{uk}^{l'}(t)\,dt$.

Back to the problem of minimizing the penalized log-likelihood (2.3.6), for fixed values $(\lambda_l)_{0\le l\le p}$, the estimation of $\big(\gamma_0(t), \gamma_1(t,s),\ \ldots,\ \gamma_p(t,s)\big)$ is obtained by solving the problem:

$$\begin{aligned}\min_{a,\ \Gamma} \mathcal{L}_{pen}(a,\ \Gamma) &= \min_{a,\ \Gamma} -2\,\mathcal{L}(a,\ \Gamma\,|\,\mathrm{Y}) + \lambda_0 \sum_{s,k=1}^{q_{\gamma^0}} a_s^0 a_k^0 \Phi_{sk}^0 + \\ &\qquad \sum_{l=1}^{p}\lambda_l \sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l \Big(\Phi_{1,ik}^{l''}\,\Phi_{2,jm}^l + 2\,\Phi_{1,ik}^{l'}\,\Phi_{2,jm}^{l'} + \Phi_{1,ik}^{l}\,\Phi_{2,jm}^{l''}\Big).\end{aligned}$$

The penalty term for any bivariate parameter $\gamma_l(t,s)$ can be seen as the tensor product between the three following terms:

(i) the $(a_{ij})_{1\le i,j\le q_{\gamma^l}}$ matrix, the 4^{th} order square tensor of dimension q_{γ^l} ;

(ii) the matrix $(a_{km})_{1\le k,m\le q_{\gamma^l}}$. We can rearrange this tensor product as a matrix product by flattening the matrix to a vector ;

(iii) the 4^{th} order tensor of basis functions flatten to a matrix.

So we get a matrix product between the row vector of length $q_{\gamma^l}^2$, the square matrix of dimension $q_{\gamma^l}^2 \times q_{\gamma^l}^2$ and the column matrix of length $q_{\gamma^l}^2$. The minimization problem can be written in matrix form:

$$\min_{a,\,\Gamma} \mathcal{L}_{pen}(a,\,\Gamma) \quad = \quad \min_{a,\,\Gamma} \text{-}\,2\,\mathcal{L}(a,\,\Gamma\,|\,\mathrm{Y}) \,+\, a^\top(\lambda \mathrm{P})\,a\,, \tag{2.3.8}$$

where λP the matrix of dimension $q_\gamma \times q_\gamma$ with $q_\gamma = q_{\gamma^0} + \sum_{l=1}^{p} q_{\gamma^l}^2$ defined as in (2.3.3). The main difference lies in the expression of the block matrix Ψ^l for $l > 0$ given by:

$$\Psi^l = \left(\Phi_{1,ik}^{l''}\,\Phi_{2,jm}^{l} + 2\,\Phi_{1,ik}^{l'}\,\Phi_{2,jm}^{l'} + \Phi_{1,ik}^{l}\,\Phi_{2,jm}^{l''} \right)_{1 \le i,j,k,m \le q_{\gamma^l}}.$$

With this expression, we proceed in the same way as in the concurrent model to obtain the penalized estimator of a and then $\left(\hat{\gamma}_l(t,s)\right)_l$.

2.4 Conformal prediction

Conformal prediction (Vovk et al. (2005)) is a framework that provides valid measures of confidence for individual predictions. It aims to quantify the uncertainty associated to predictions made by machine learning models, allowing users to assess the reliability of the model's predictions. The key idea of the method is to construct prediction regions around individual predictions such that, with a certain level of confidence, the true value of the response falls within these regions. The resulting interval has desirable properties such as distribution freeness and non-asymptoticity. Despite these good properties, classical conformal methods leads to intervals of constant or weakly varying length across the input space. Romano et al. (2019) proposed a fully adaptive method to heteroscedasticity named Conformalized Quantile Regression (CQR) that combines conformal prediction to quantile regression inheriting the advantages of the both methods. We adapt the CQR method to function-on-function regression.

2.4.1 Functional conformalized quantile regression

In its original formulation, conformal prediction works with methods of point prediction such as linear regression but also decision trees, support vector, boosting, neural networks (Angelopoulos and Bates (2023) for a nice recent overview). Based on the prediction given by the model, the method works by compute a non-conformity measure, which measures how unusual an example looks relative to previous examples, and the conformal prediction algorithm turns this nonconformity measure into prediction regions. The CQR extension constructs prediction intervals around the quantile estimates, allowing users to assess the reliability of the model's predictions at different quantile levels. Quantile regression attempts to learn the τ-conditional quantile of $Y(t)|X(t) = x(t)$ for any possible value of $x(t)$. Since $Y(t)|X(t) = x(t) < q_{0.05}(x(t))$ with probability 5% and $Y(t)|X(t) = x(t) > q_{0.95}(x(t))$ with probability 5%, then $[q_{0.05}(x(t))\,;\, q_{0.95}(x(t))]$ is a valid interval with 90% coverage.

For a given labelled dataset $\left\{(\mathrm{X}_i(t_j), \mathrm{Y}_i(t_j)), 1 \le i \le n \;\; 1 \le j \le m\right\}$, CQR works as follows:

1. fit two conditional quantiles regression functions $\widehat{\mathrm{q}}_{\tau_0}\Big(\mathrm{X}_i(t_j)\Big)$ and $\widehat{\mathrm{q}}_{\tau_1}\Big(\mathrm{X}_i(t_j)\Big)$ with $\tau_0 < \tau_1$ on a proper training set $\mathcal{A}_1$.

2. On the calibration set $\mathcal{A}_2$, compute conformity scores E_i given by:

$$E_i := \left\{ \max\Big(\widehat{\mathrm{q}}_{\tau_0}\big(\mathrm{X}_i(t_j)\big) - \mathrm{Y}_i(t_j)\,;\, \mathrm{Y}_i(t_j) - \widehat{\mathrm{q}}_{\tau_1}\big(\mathrm{X}_i(t_j)\big)\Big) \right\}_{1\leq j\leq m} \in \mathbb{R}^m \qquad \text{for } i \in \mathcal{A}_2$$

3. Finally, the prediction interval of a new observation $\widehat{\mathrm{Y}}_{n+1}(t_j)$ with input $\mathrm{X}_{n+1}(t_j)$ is given by

$$C\Big(\mathrm{X}_{n+1}(t_j)\Big) \quad = \quad \Big[\widehat{\mathrm{q}}_{\tau_0}\Big(\mathrm{X}_{n+1}(t_j)\Big) - Q_{1-\tau}(\mathcal{A}_2)\,;\, \widehat{\mathrm{q}}_{\tau_1}\Big(\mathrm{X}_{n+1}(t_j)\Big) + Q_{1-\tau}(\mathcal{A}_2)\Big] \tag{2.4.1}$$

Where $Q_{1-\tau}(\mathcal{A}_2)$ is the $\Big((1-\tau)(1+\frac{1}{|\mathcal{A}_2|})\Big)^{\text{th}}$ quantile of the multivariate errors E_i.

Romano et al. (2019) establish that if the elements of the sample $\{\big(\mathrm{X}_i(t_j), \mathrm{Y}_i(t_j)\big), 1 \leq i \leq n+1\}$ are interchangeable and conformity scores E_i are almost surely distinct, the predictive interval (2.4.1) satisfies

$$1-\tau \leq \mathbb{P}\Big\{\widehat{\mathrm{Y}}_{n+1}(t_j) \in C\Big(\mathrm{X}_{n+1}(t_j)\Big)\Big\} \leq 1-\tau+\frac{1}{|\mathcal{A}_2|+1}.$$

As described in the steps of the method above, we need to build (i) the functional quantile regression $\widehat{q}_\tau(.)$ and (ii) the quantile of multivariate conformity scores E_i. This process is described in the two next subsections describe.

2.4.2 Multivariate quantiles

Multivariate quantiles extend the concept of univariate quantiles to multivariate data, allowing for the characterization of the joint distribution of several random variables. But unlike in the 1-dimensional space $\mathbb{R}$, the d-dimensional space $\mathbb{R}^d$ (with $d \geq 2$) is not canonically ordered. So the strong order-related notions of quantile and rank, signs and distribution are not quite natural and still an open problem. Hallin et al. (2021) proposed a center-outward definition of multivariate distribution and quantile functions based on measure transportation introduced by Chernozhukov et al. (2017) for the so-called Monge-Kantorovitch problem. These definitions are made under the assumption of nonvanishing density support and the main advantage is the fact that it satisfying desirable properties as distribution-freeness and the maximal invariance property that make univariate quantile be successful tools for statistical inference.
We consider two distributions: a discrete n-sample (empirical) distribution μ represented by a set of points $Z_1^{(n)}, \ldots, Z_n^{(n)}$ in $\mathbb{R}^d$ and a augmented uniform grid distribution ν. The goal of the optimal transport is to find the map $\mathbf{T}$ between the sample and to the uniform grid that minimizes the total transportation cost. It is also the most efficient way to redistribute mass from the sample points to the grid points while minimizing the L^2 distance. Mathematically, the optimal L^2 transport map can be defined as:

$$\mathbf{T} := \underset{T\in\mathcal{T}(\mu,\nu)}{\arg\min} \int \|x - T(x)\|^2 d\mu(x) \tag{2.4.2}$$

where $\mathcal{T}(\mu, \nu)$ is the set of all transports mapping μ to ν.
Optimal transport can be used to estimate multivariate quantiles in a robust manner, particularly in the presence of heterogeneous data. In the context of multivariate distributions, quantiles represent values that partition the distribution into segments of equal probability mass. In a similar way, optimal transport can be used to estimate multivariate quantiles by minimizing the transportation cost required to redistribute mass from the distribution tails to the quantile levels. To define multivariate quantiles using optimal transport, we firstly consider a reference distribution ν. The common choice is the uniform distribution on the unit ball. We also consider the empirical distribution μ derived from the data. Now denoted by $\mathbf{T}$ the optimal transport that pushes μ forward to ν : $\mathbf{T}\#\mu = \nu$. The second step is to reduce the radius of the unit ball with the aim to exclude a τ-probability to be in the crown delimited by the unit ball and the ball of radius r. The third step is dedicated to defined the τ-quantile by simply choose to point with lower rank among the data points excludes by the reduction of the radius of the unit ball.

Step 1: Construct the uniform grid on the unit ball

Since most of the volume of a unit ball is concentrated in a thin layer near its surface, it is not so trivial to generate uniformly in the unit ball. For this, it is prior to ensure that the resulting distribution must match with volume of crowns. In d-dimensional space, the volumes of the ball of radius r and the unit ball (radius 1) are given respectfully by

$$\mathrm{V}_{d,r} = \frac{\pi^{d/2}}{\Gamma(\frac{d}{2}+1)}\, r^d \quad \text{and} \quad \mathrm{V}_{d,1} = \frac{\pi^{d/2}}{\Gamma(\frac{d}{2}+1)}.$$

So the volume of the crown will be : $\mathrm{V}_{\mathcal{C}_r} = \mathrm{V}_{d,1} - \mathrm{V}_{d,r} = \mathrm{V}_{d,1}(1 - r^d)$.

The probability of being in this crown, for the generated distribution, are

$$p_{\mathcal{C}_r} = \frac{\mathcal{V}_r}{\mathrm{V}_{d,1}} = 1 - r^d.$$

With this result, we can derive the probability distribution function (and the density function) of the random variable R of the radii of balls required to be in the crown:

$$\mathrm{F}_{\mathrm{R}}(r) = 1 - \mathbb{P}(R > r) = 1 - (1 - r^d) = r^d \quad \text{and} \quad f_{\mathrm{R}}(r) = d\, r^{d\text{-}1}$$

So, we propose the following algorithm to generate a uniform grid on the unit ball

1. simulate a set of d-dimensional points according to the normal distribution
2. normalize the simulated points
3. simulate as many radii r as points generated to the distribution F_{R}. Then $r = \exp\left(\frac{\log(u)}{d}\right)$ with $u \sim \mathcal{U}([0;1])$.
4. Finally, each normalized point is multiplied by a simulated radius

With this process, we ensure that the generated uniform distribution will respect the geometry and the distribution of volumes in high dimensional spaces.

Step 2: Find the radius r that exclude τ-probability points

We answer in this part the question: "By how much should the radius of the unit ball be reduced so that the excluded volume represents τ% of the total volume ?"
In this context, τ represents the ratio of the volume of the crown formed by the unit ball and the ball of radius r to the volume of the unit ball.

$$\tau \quad = \quad \frac{\mathrm{V}_{\mathcal{C}_r}}{\mathrm{V}_{d,1}} = 1 - r^d \quad \Longleftrightarrow \quad r = \exp\Big(\frac{log(1-\tau)}{d}\Big) \tag{2.4.3}$$

Then to exclude τ% of the total volume, we can reduce the radius of the unit ball from 1 to r.

Step 3: Define the multivariate quantile

Once we have understood how to properly construct an uniform grid on the d-dimensional unit ball and how to reduce the radius of this ball to extract a well-defined volume, the final step is to determine the empirical multivariate τ-quantile of the set of points $Z_1^{(n)}, \ldots, Z_n^{(n)}$ in $\mathbb{R}^d$. The empirical τ-quantile is obtained via an optimal transport $\mathbf{T}$ of the sample distribution to the deterministic uniform grid on the unit ball. By reducing the radius of that ball from 1 to r according to formula (2.4.3), the τ-quantile given by the convex enclosure of the observations excluded by radius reduction.

2.4.3 Quantiles function-on-function regression by perturbation

Quantile regression introduced by Koenker and Bassett (1978) offers a convincing solution by estimating conditional quantiles, which provides a more complete understanding of the distribution of the response variable. In FDA, the main difficulty relies on the fact that it is hard to design an equivalent to the pinball loss for functional output variables. In the scalar-on-function setup, we can mention works from Ferraty et al. (2005) and Cardot et al. (2005) which estimate conditional quantiles using nonparametric estimation and spline estimators respectively. For function-on-scalar and function-on-function setups, Liu et al. (2020) proposed a Bayesian framework by developing a scalable Gibbs sampler for an asymmetric Laplace likelihood distribution for the functional basis coefficients. Beyaztas et al. (2024) designed a pointwise pinball loss and a quadratic roughness penalty to estimate the model. Rather than using these techniques, we propose to use an alternative technique which does not require the assumption of continuity.

We developed a function-on-function quantile regression by perturbing linear function-on-function regression repeatedly. As any resampling methods such as bootstrap, jacknife and so on, we generate artificial samples of predictions of the linear regression. Our resulting quantile regression is simply the multivariate quantiles of that perturbing predictions. Originally, the use of perturbation in statistics can be found at least back to Gauss (1809) who used them to linearize a non linear problem. Later Stewart (1990)'s work examined the application of matrix perturbation theory for solving least squares regression problems, when the design matrix is contaminated by random errors. Our work is mostly inspired by Minnier et al. (2011) in which an interesting innovative and robust perturbation approach for regularized regression estimation is proposed and is particularly useful for high dimensional data. The idea of the method is to create series of perturbed datasets using them to estimate the variability of the regression coefficients. Specifically, for a given dataset $\big(\mathrm{X}(t), \mathrm{Y}(t)\big)$, create perturbed datasets $\big(\mathrm{X}(t), \mathrm{Y}(t) + \varepsilon\big)$, where ε is a small random perturbation drawn from a specified distribution (e.g., normal with mean 0 and small

variance or exponential with parameter 1).
Let N_p be the number of perturbed datasets. The perturbed response is defined as:

$$\mathrm{Y}_i^{(\ell)}(t_j) \quad = \quad \mathrm{Y}_i(t_j) + \varepsilon_{ij}^{(\ell)} \qquad \text{with } \varepsilon_{ij}^{(\ell)} \sim \mathcal{E}(1) \text{ and } 1 \leq \ell \leq N_p$$

With these new response variable, we can apply the PenFFR algorithm for any of models (2.1.1) or (2.1.2) depending to the initial model we work with. Based on the predictions given by each perturbed regression, the computation of multivariate quantiles lead to the functional quantile regression.

2.5 Simulation study

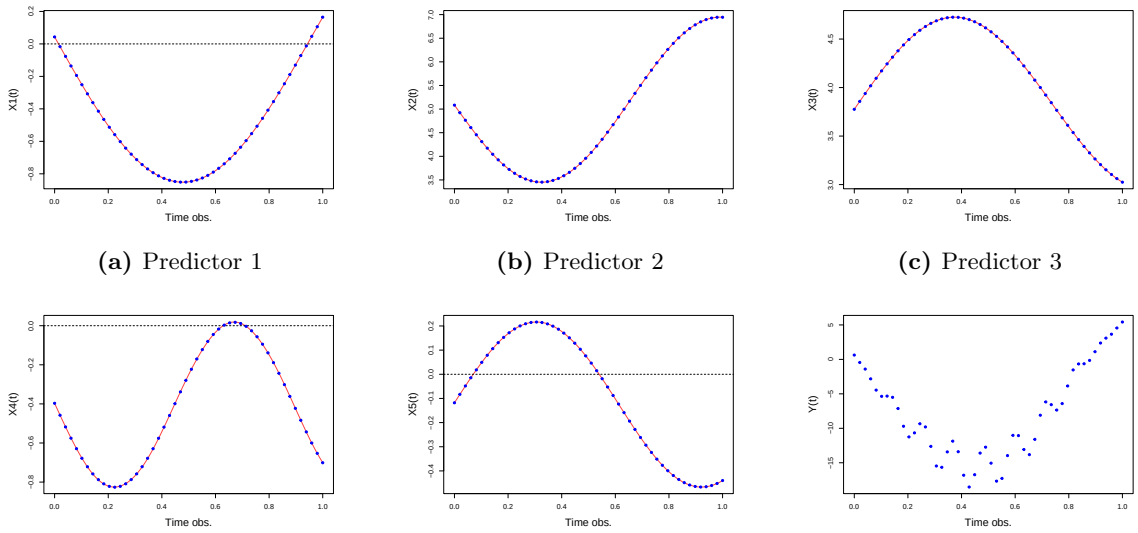
The aim of this section is firstly dedicated to illustrate and validate the estimation and predictive capabilities of the proposed Penalized Function-on-Function Regression (PenFFR) algorithm described in Sections 2.3 and 2.4 through a series of Monte Carlo simulations in the framework of "perfectly controlled" data, i.e. in the set-up where the assumptions about the distribution are the ones underlying our theory. Secondly, we will compare on the simulated datasets, the performance of our PenFFR method to pffr. The main properties of interest are accuracy, interpretability and smoothness of the estimated parameters, prediction quality and coverage of confidence intervals. Only the concurrent model is considered in this section, but similar results have been obtained for the integral model.

2.5.1 Data generation process

We will first simulate the covariates and then use them as input to our regression model in order to simulate the corresponding response. The $p = 5$ functional covariates are simulated at $m = 50$ equidistant viewpoints $(t_j)_j$ over the domain $\mathrm{T} = [0, 1]$ according the following procedure :

$$\mathrm{X}_i^l(t_j) = \quad \xi_{i,1}^l + \Big(\log(10 + t_j)\Big)^{\xi_{i,2}^l} + \xi_{i,3}^l \sin\left(\frac{2\pi t_j}{\xi_{i,4}^l}\right) \tag{2.5.1}$$

where $\xi_{i,r}^l$ is drawn from $\mathcal{U}\big([\text{-}2, 2]\big)$ $(1 \leq r \leq 4)$. We can see this data in Figures (2.1a)-(2.1e) inside Figure 2.1 for one randomly chosen individual (blue dots).



(a) Predictor 1 **(b)** Predictor 2 **(c)** Predictor 3

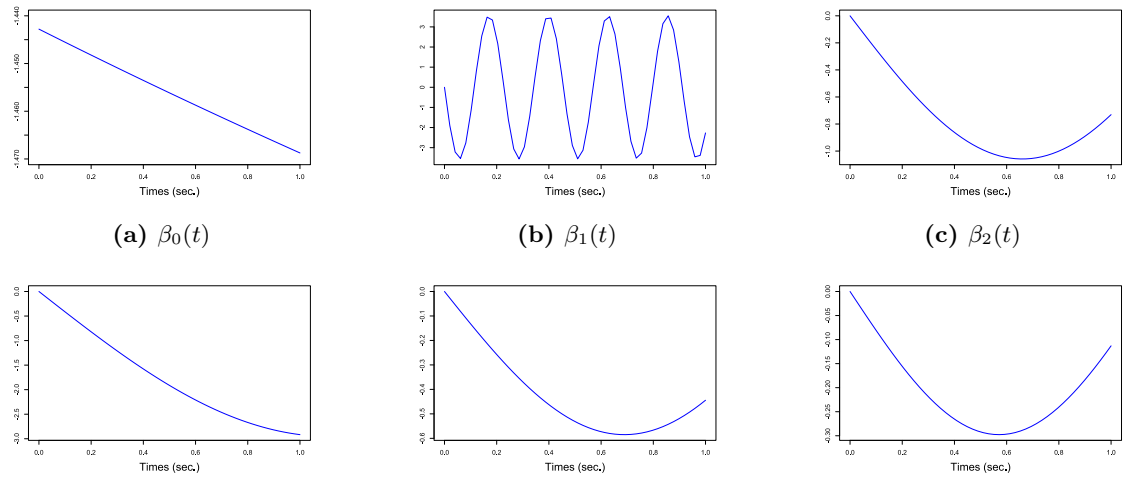
(d) Predictor 4 **(e)** Predictor 5 **(f)** Response

Figure 2.1: Simulated predictors (blue dots for raw data and red lines for functional recovery) and functional (concurrent) response for a randomly chosen individual.

Scenarios	number of observations: n	variance : σ^2
S1	200	1
S2	200	4
S3	500	1
S4	500	4

Table 2.1: The four scenarios of the simulation study

Before estimating the model, we first compute the underlying expansion into a basis of B-Splines. We obtain the red curves in Figure 2.1 using K = 10 basis functions and equidistant distributed nodes. The functional parameters $\beta(t) = \big(\beta_0(t), \beta_1(t), \ldots, \beta_p(t)\big)$ are chosen as follows: $\beta_0(t_j) = \Big(\log(10 - 2\pi t_j)\Big)^{\rho_0}$ and $\beta_l(t_j) = \rho_1^l \sin\left(\frac{4\pi t_j}{\rho_2^l}\right)$ with $\rho_0,\ \rho_1^l,\ \rho_2^l$ some constants given in Appendix A.1. Figure 2.2 shows the corresponding representations of the functional parameters.



(a) $\beta_0(t)$ **(b)** $\beta_1(t)$ **(c)** $\beta_2(t)$

(d) $\beta_3(t)$ **(e)** $\beta_4(t)$ **(f)** $\beta_5(t)$

Figure 2.2: Functional parameters in the concurrent model.

Given the proposed functional covariates and functional parameters, we can now compute the functional response using the concurrent Model (3.2.1), for two different sampling sizes $n \in \{200; 500\}$. In this experiment, $\varepsilon_i(t)$ is a Gaussian noise with mean 0 and two levels of variance $\sigma^2 \in \{1; 4\}$. Table 2.1 list the 4 simulation scenarios. For each configuration, we run $N = 50$ Monte Carlo simulations.

2.5.2 Assessment criteria

The performance of our estimation procedure is assessed with two criteria: prediction accuracy and estimation error for the model parameters. We extend to the functional framework the well-known Mean Relative Prediction Error (MRPE), which is used to quantify the distance between the actual and the predicted value of the functional response:

$$\text{MRPE} = \frac{1}{m}\sum_{j=1}^{m}\left(\frac{\sum_{i=1}^{n}\left(\mathrm{Y}_i(t_j) - \widehat{\mathrm{Y}}_i(t_j)\right)^2}{\sum_{i=1}^{n}\mathrm{Y}_i(t_j)^2}\right). \tag{2.5.2}$$

We also define one extension of the determination coefficient, which consists of a simple arithmetic average of the classical determination coefficient along the time observation of the functional response. This determination coefficient noted $\widetilde{\mathrm{R}}^2$ is defined as follows:

$$\widetilde{\mathrm{R}}^2 = \frac{1}{m}\sum_{j=1}^{m}\left(1 - \frac{\sum_{i=1}^{n}\left(\mathrm{Y}_i(t_j) - \widehat{\mathrm{Y}}_i(t_j)\right)^2}{\sum_{i=1}^{n}\left(\mathrm{Y}_i(t_j) - \overline{\mathrm{Y}}_i(t_j)\right)^2}\right), \tag{2.5.3}$$

where $\widehat{\mathrm{Y}}_i(t_j)$ is the predicted output of the sample i at time t_j and $\overline{\mathrm{Y}_i}(t_j)$ the mean function of the output sample at time t_j.

To evaluate the performance of the estimation parameters, we compare the actual functional parameters with those provided by our models using the Mean Square Error (MSE) given by:

$$\text{MSE} = \frac{1}{p+1}\sum_{l=0}^{p}\left[\frac{1}{m}\sum_{j=1}^{m}\left(\beta_l(t_j) - \widehat{\beta}_l(t_j)\right)^2\right]^{1/2}. \tag{2.5.4}$$

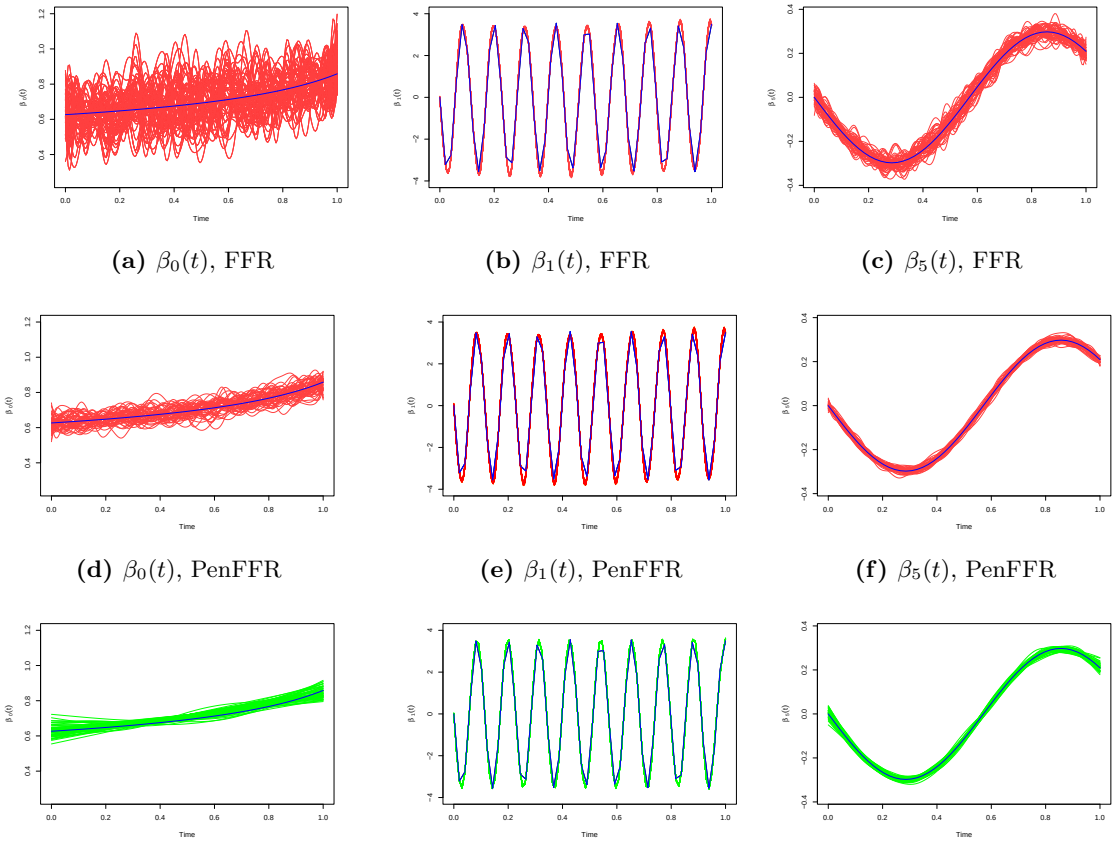
In addition, to assess the quality of produced confidence intervals, we use the coverage proportion. It is the proportion of actual data that fall in the predicted confidence interval. The ideal situation for level τ is $(1-\tau)$. The coverage proportion (denoted by CovP) is given by the formula

$$\text{CovP} \quad = \quad \frac{1}{n\,m}\sum_{i=1}^{n}\sum_{j=1}^{m}\mathbb{I}_{\left\{a_i^L(t_j)\leq \text{Y}_i(t_j)\leq a_i^U(t_j)\right\}}, \tag{2.5.5}$$

where $a_i^L(t_j)$ (resp. $a_i^U(t_j)$) are the lower (resp. upper) bound of the confidence interval for the observation i at time t_j.

2.5.3 Simulation results

Figure 2.3 represents for three of the six, actual parameters and the estimated ones given by FFR, PenFFR and pffr methods for all the Monte Carlo repetitions in the S3 scenario. We noted for any of them (except for the intercept) a good estimation of parameters and we can also notice a substantial reduction of the random fluctuations of penalized parameters compared to the non-penalized one. For a less graphical and more global comparison, Figure 2.4 compares the boxplots of MSE (2.5.4) of the estimated parameters over Monte Carlo repetitions of the 4 scenarios of the FFR method. As expected, for any parameters, the MSE decreases as the number of observations increases (**S1** vs **S3**) and it also decrease when we have smaller variance of the model error (**S2** vs **S4**).



(a) $\beta_0(t)$, FFR **(b)** $\beta_1(t)$, FFR **(c)** $\beta_5(t)$, FFR

(d) $\beta_0(t)$, PenFFR **(e)** $\beta_1(t)$, PenFFR **(f)** $\beta_5(t)$, PenFFR

(g) $\beta_0(t)$, pffr **(h)** $\beta_1(t)$, pffr **(i)** $\beta_5(t)$, pffr

Figure 2.3: Estimated parameters vs actual ones in the simulated scenario 3 for the methods.

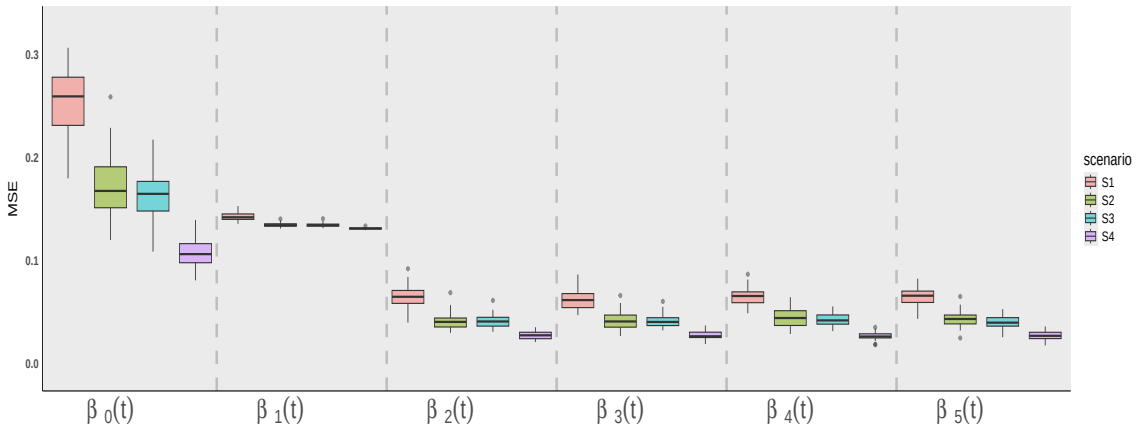


Figure 2.4: Boxplots of MSE of estimated parameters over the $N = 50$ Monte Carlo repetitions for the non penalized method.

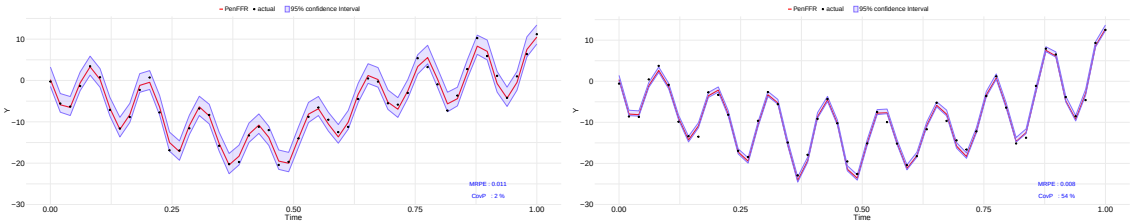
Prediction accuracy : We have generated a test sample with $n_{\text{test}} = 2000$ observations, and we compare the difference between the actual values of the functional response and the prediction given by each method in the 4 scenarios through the MRPE. Table 2.2 shows in its first column the average and standard deviation of MRPE. As in the parameter estimation, the MRPE decreases as the number of observations in the train set increases, but the additive noise has a more greater impact. In any scenario, PenFFR has slightly better performances than FFR which has better performances than pffr.

		Test set MRPE ($\times 10^3$)	$\tilde{R}^2$	CovP	MSE
S1	FFR	55.86 (0.25)	0.922 (0.0016)	11.8% (0.036)	0.109 (0.006)
	PenFFR	55.48 (0.33)	0.906 (0.0007)	10.6% (0.058)	0.088 (0.003)
	pffr	91.70 (0.07)	0.882 (0.0003)	-	0.085 (0.002)
S2	FFR	54.59 (0.20)	0.925 (0.0004)	13.4% (0.058)	0.079 (0.007)
	PenFFR	53.78 (0.17)	0.912 (0.0006)	11.7% (0.101)	0.073 (0.009)
	pffr	91.65 (0.06)	0.883 (0.0001)	-	0.074 (0.001)
S3	FFR	54.52 (0.13)	0.924 (0.0008)	2.6% (0.005)	0.077 (0.004)
	PenFFR	54.45 (0.07)	0.909 (0.0004)	4.9% (0.029)	0.070 (0.006)
	pffr	91.48 (0.05)	0.883 (0.0001)	-	0.074 (0.001)
S4	FFR	54.00 (0.07)	0.926 (0.0003)	9.0% (0.013)	0.058 (0.003)
	PenFFR	53.88 (0.09)	0.921 (0.0003)	8.6% (0.015)	0.054 (0.005)
	pffr	91.35 (0.02)	0.883 (0.0001)	-	0.074 (0.001)

Table 2.2: Average and standard deviation of accuracy criteria for the 4 simulated scenarios.

The functional determination coefficient $\tilde{R}^2$ computed over all the scenarios is also presented in

Table 2.2. From this table, we observe that when the additive noise increases (**S1** vs **S2** and **S3** vs **S4**), the coefficient of determination gets smaller. By increasing the sample size (**S1** vs **S3** and **S2** vs **S4**), we improve the determination coefficient.
The CovP criterion given in Table 2.2 shows a better coverage when the noise level decreases (**S1** vs **S2** and **S3** vs **S4**).
Finally, Figure 2.5 illustrates for two randomly chosen observations, the actual values (black dots), the predictive curves given by PenFFR method (red curve) and the 95% confidence interval of the PenFFR prediction.



(a) prediction 1 **(b)** prediction 2

Figure 2.5: Actual values (black dots) and PenFFR predictive curves (red lines) of the functional response for two randomly chosen individuals.

To sum up, the presented results shows that the PenFFR has the best performance. The FFR method, while effective in certain contexts, often struggles with overfitting due to its lack of regularization, which can lead to less reliable estimates, especially when dealing with complex or noisy data. On the other hand, the pffr method offers more flexibility by incorporating smoothness in both the functional predictor and response, also faster results but it may not fully address the issue of controlling model complexity. As a result, PenFFR consistently provides better predictive accuracy and stability, making it the most robust and reliable choice among the three methods.

2.6 Application to real data

In this section, we apply the proposed methodology for function-on-function regression (FFR and PenFFR) for concurrent and integral models. These models are applied to two well-known data sets in FDA: Canadian Weather (CW) data available in the R package `fda` and Hawai Ocean (HO) data available in the R package `FRegSigComp`. We compare the prediction accuracy obtained using our method with the accuracy obtained with other existing methods: integral and concurrent Penalized Function-on-Function Regression (pffr, Ivanescu et al. (2015)) implemented in the R package `refund`; the signal compression approach (*wSigcomp*) designed by Luo et al. (2016) for the integral model and implemented in the R packages `FRegSigComp`; the Optimal Penalized Function-on-Function Regression (OPFFR) for the integral model (Sun et al., 2018), the Functional Principal Component Analysis (FPCA) and Functional Data Analysis method (FDA) also for the integral model (Ramsay and Silverman, 2005). Due to the unavailability of code for the OPFFR approach, we simply use the published results as presented in their paper (Sun et al., 2018).

Hyper-parameter tuning: For our methods (FFR and PenFFR), we consider cubic B-splines basis functions for both functional predictors and regression coefficients. On the CW data set, we use 100 basis functions to address the functional and complex nature of the predictors and on HO data set, we use 40 basis functions. This choice is motivated by the fact that on the raw data, predictors on CW data set has 365 measurements while predictors in the HO data set have 200 measurements. The number of basis functions of parameters is set to 40 for the concurrent model and set to 10 for the integral model on CW data. For the HO data, knowing that we have 4 functional predictors and by the fact that the number of features of design matrix depends on the squared of the number of basis functions in the integral model, we choose 20 basis functions for the concurrent model and only 6 for the integral model. The penalty parameters λ_l of any predictor is selected using cross-validation on a predefined grid of values (10 equispaced values between 0.1 and 2.0).
For the pffr method we used the default settings prescribed by the software and only set the number of basis functions for both the functional parameters and predictors. To correctly compare to our proposed method, we also used a cubic splines basis for both the functional predictors and parameters for the two (CW and HO) data sets. We also use the same number of basis functions to recover the functional nature of the predictors and on parameters as in our method.
For the *wSigcomp* method designed for the integral model, the default settings of the software are also used. For the HO data set which is tested by authors in their package description, the number of basis functions is set to 40 for the functional parameters and 20 for predictors. For the CW data, we slightly change but in the same proportion these value and set the number of basis functions involved for the functional parameters to 80 and the predictors to 40. We have detailed the choices of the hyperparameters but it should be noted that the performance of all these methods remains slightly sensitive to a reasonable variation of these values.

	Canadian Weather Data			Hawaii ocean data		
Methods	Type of basis	$X_i^\ell(t)$	$\beta_\ell(t)$	Type of basis	$X_i^\ell(t)$	$\beta_\ell(t)$
Integral PenFFR / FFR	cubic B-splines	100	10	cubic B-splines	40	6
Concurrent PenFFR / FFR	cubic B-splines	100	40	cubic B-splines	40	20
Integral pffr	cubic B-splines	100	10	cubic B-splines	40	6
Concurrent pffr	cubic B-splines	100	40	cubic B-splines	40	20
wSigcomp	wavelets + SVD	40	80	wavelets + SVD	20	40
OPFFR	/	/	/	/	/	/
FDA	Cubic B-splines	/	10	/	/	/
FPCA	SVD	/	/	/	/	/

Table 2.3: Number of basis functions for the regression coefficients $\beta_\ell(t)$ and the covariates $X_i^\ell(t)$

2.6.1 Canadian weather data

The data set consists of $m = 365$ daily temperature measurements (average over the years 1961 to 1994) at $n = 35$ weather stations in Canada and their corresponding daily precipitation (in log scale). The weather stations are located in K = 4 climate zones: Atlantic, Pacific, Continental and Arctic. The aim is to use the daily temperature to predict the precipitation at each station.
Figure 2.6 gives the daily average over the years 1961 to 1994 (temperature on the left, precipitation on the right). Note that the stations in the Pacific zone have the highest precipitation values, and

stations from this zone also have the highest temperatures in the winter. The same can be said about the stations in the Arctic zone for low temperatures and precipitation. A positive relationship between temperature and precipitation can therefore be suspected.

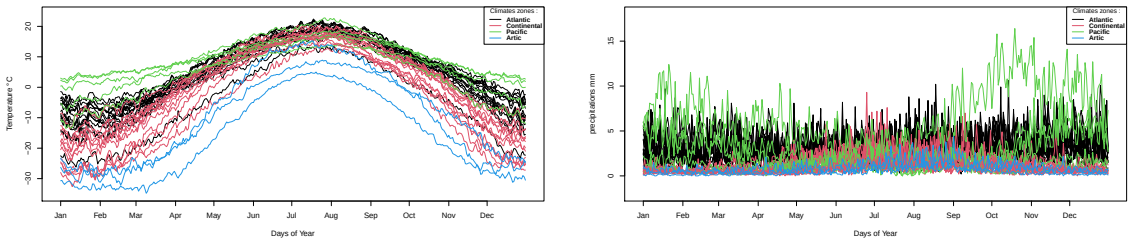


Figure 2.6: 35 daily mean temperature (a) and precipitation (b) measurement curves.

Our methods (FFR and PenFFR) for the concurrent model (2.1.1) and integral model (2.1.2) are compared with the PFFR, OPFFR, FPCA, FDA and *wSigcomp* methods. As previously mentioned, we use for the OPFFR, FDA and FPCA methods, the results presented in Sun et al. (2018) in terms of prediction accuracy over the 365 days of the year through the leave-one-out cross-validation integrated square error (ISE) given by:

$$\mathrm{ISE}_i = \int_0^{365} \left(\mathrm{Y}_i(t) - \widehat{\beta}_{(-i)}\, \mathrm{X}_i(t) \right)^2 dt$$

where the predictor $\mathrm{X}_i(.)$ derives from the noisy daily temperature measurements; the functional response $\mathrm{Y}_i(.)$ is the log daily precipitation and $\widehat{\beta}_{(-i)}$ is the functional parameter estimated in the data set of all the observations except for the i^{th} observation.

For sake of reducing the computational burden, instead of the ISE, the L^2-norm between the actual and prediction values on a grid of values t is used as a surrogate. It is given by:

$$\widehat{\mathrm{ISE}}_i = \sum_{j=1}^{365} \left(\mathrm{Y}_i(j) - \widehat{\beta}_{(-i)}\, \mathrm{X}_i(j) \right)^2. \tag{2.6.1}$$

The average and standard deviation values of $\widehat{\mathrm{ISE}}_i$ over the leave-one-out models for the different models are given in Table 2.4. They show the numerical advantage of our proposed PenFFR method over the other methods. We also note that the variance observed in our predictions remains quite high for the different models. This is due to the quality of the input data. For recall that we are trying to predict precipitation from temperature on a dataset of only 35 very different weather stations with daily measurement for each of them.

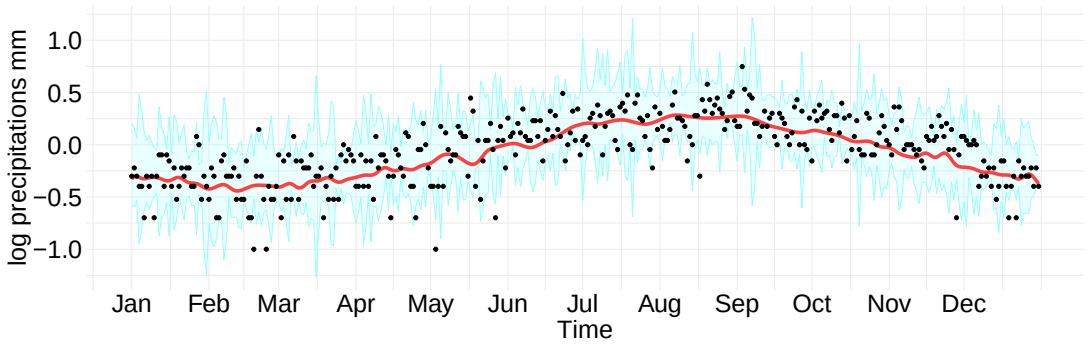
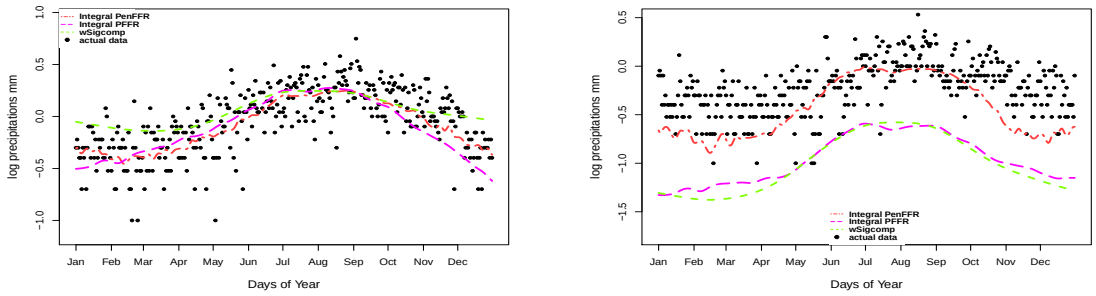


Figure 2.7: Prediction (red curve), actual data (black dots) and confidence interval (cyan region) for the Churchill station given by the PenFFR method.

Methods	$\widehat{\text{ISE}}$
Integral PenFFR	**33.66 (22.99)**
Concurrent PenFFR	36.40 (40.42)
Integral FFR	34.63 (26.03)
Concurrent FFR	36.50 (40.51))
Integral pffr	41.37 (48.91)
Concurrent pffr	89.31 (52.03)
wSigcomp	45.37 (52.45)
OPFFR	40.28 (45.76)
FDA	44.16 (56.95)
FPCA	45.51 (45.78)

Table 2.4: The average (and standard deviation) of $\widehat{\text{ISE}}$ for the Canadian Weather data set. The best result is in boldface.

Figure 2.7 shows for a randomly choosen weather station the predictive curve given by the PenFFR method. We can also observe the good coverage of the 95% confidence interval given by our functional conformal prediction method. Figure 2.8 shows the prediction obtained using the integral model (since it appeared to be the best model for this data set) with FFR, PenFFR, pffr. The prediction is given for two randomly chosen weather stations (Churchill and Inuvik) and are compared with the actual precipitation. Similar results are illustrated by Figure A.3 in the appendix for the concurrent model.



(a) Churchill station **(b)** Inuvik station

Figure 2.8: Prediction on two randomly chosen stations.

2.6.2 Hawaii ocean data

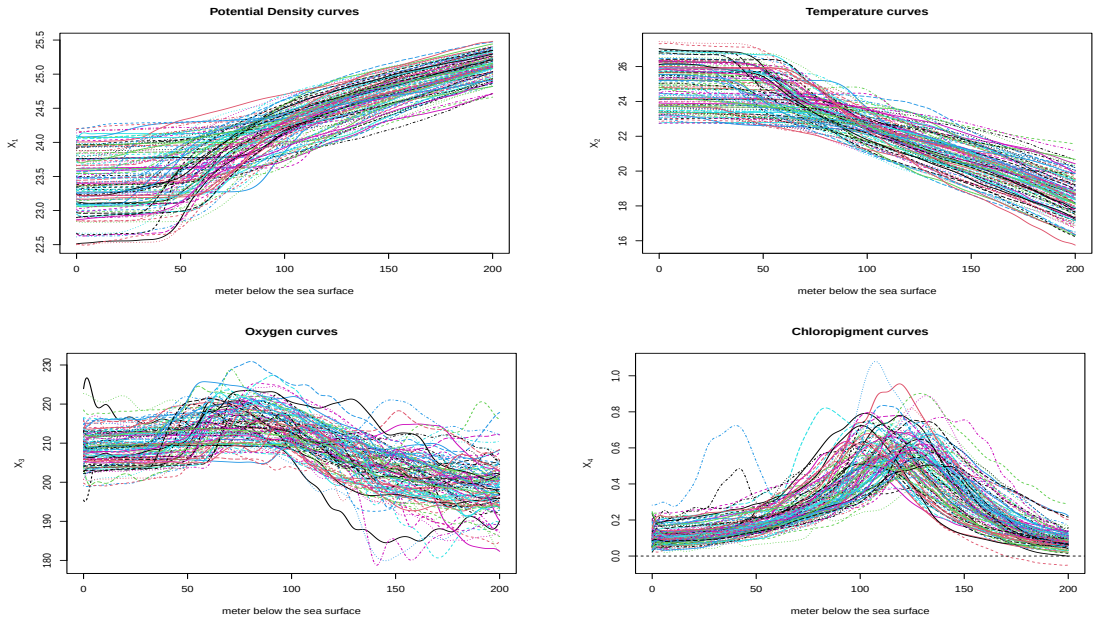


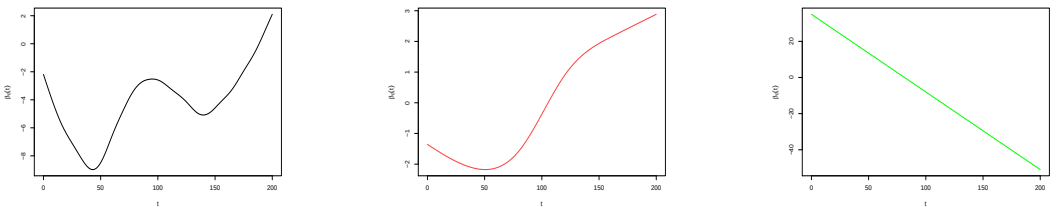
Figure 2.9: Original sample curves of predictors expanded by cubic B-splines basis with 40 basis functions.

This data set is one of those used by Luo et al. (2016) to evaluate their *wSigcomp* approach. The data set includes physical and biochemical oceanographic observational data from the Hawaii Ocean Time-series (HOT) Program, including thermosalinograph, Conductivity, Temperature and Depth (CTD), bottle and biochemical data. The HOT program makes repeated observations of

the physics, biology and chemistry at a site approximately 100 km north of Oahu, Hawaii. In the data set, five variables: Salinity, Potential Density, Temperature, Oxygen and Chloropigment, are observed every two meters between 0 and 200 meters below the sea surface on 116 different days. This data set is available from the R package `"FRegSigComp"`, under the name Ocean data. It consists of 5 functional variables with 116 individuals, each having 101 measurement points. Here, we consider the function-on-function regression model with the salinity curves as the response variable $\mathrm{Y}(t)$ and (Potential Density, Temperature, Oxygen, Chloropigment) curves as functional predictors $\mathrm{X}(t) = \big(\mathrm{X}^1(t), \mathrm{X}^2(t), \mathrm{X}^3(t), \mathrm{X}^4(t)\big)$. We split the full data set into two train/test sub-data sets where the training data consists of the 50 first days (observations) only.

First of all, we expand all the functions considered into a cubic B-spline basis with 40 basis functions. Figure 2.9 displays the sample curves for these variables.

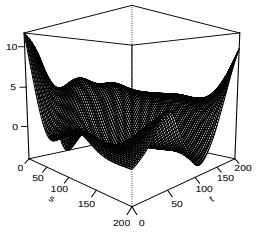
Our FFR and PenFFR methods is compared with pffr and *wSigcomp* in the setting of integral models. We also consider PenFFR and pffr for the concurrent model. The others previous competitors such as OPFFR, FPCA and FDA cannot be use here due to the unavailability of the code. Figure 2.10 and 2.11 show the estimated parameters $\hat{\gamma}_0(t)$ and $\hat{\gamma}_j(t,s)$, $1 \leq j \leq 4$ obtained for the three methods in the case of the integral model. We first notice that the shape of the estimated parameters is smooth for our method (third column). In addition, Figure A.2 in the appendix shows the estimates $\hat{\beta}_j(t)$, $0 \leq j \leq 4$ of the concurrent model with the PenFFR and pffr methods.



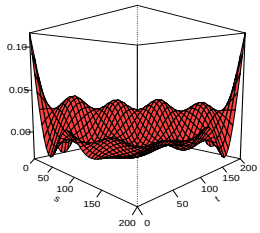
(a) $\hat{\gamma}_0(t)$, *wsigcomp* **(b)** $\hat{\gamma}_0(t)$, pffr **(c)** $\hat{\gamma}_0(t)$, PenFFR

Figure 2.10: Estimates $\hat{\gamma}_0(t)$, $1 \leq j \leq 4$ for the three methods: *wSigcomp* (left column), integral PFFR (middle column) and integral PenFFR (right column).

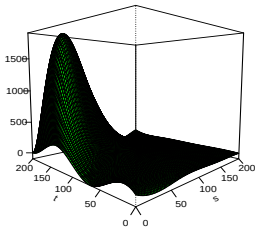
Prediction accuracy using $\widehat{\mathrm{ISE}}$ on a test set of size 66 is shown in Table 2.5. Since the number of individuals for this data (116) is larger than the size of the previous data set, we evaluate the performance on a single test set rather than using cross-validation in order to circumvent the potentially heavy computational burden. The efficiency of our produced confidence intervals is illustrated by Figure 2.12. Also, our method is shown once again to outperform all the other methods as illustrated in Figure 2.13 which shows predictions on two randomly chosen individuals.



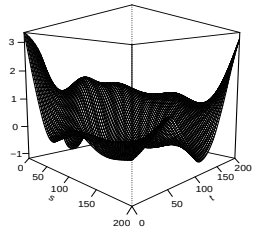
(a) $\hat{\gamma}_1(s,t)$



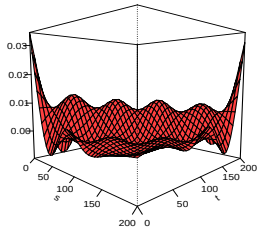
(b) $\hat{\gamma}_1(s,t)$



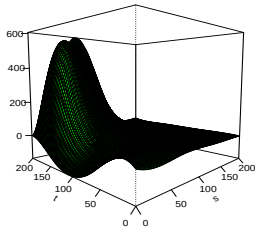
(c) $\hat{\gamma}_1(s,t)$



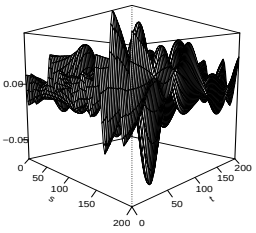
(d) $\hat{\gamma}_2(s,t)$



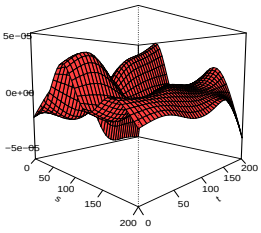
(e) $\hat{\gamma}_2(s,t)$



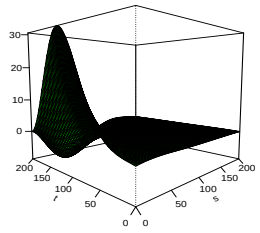
(f) $\hat{\gamma}_2(s,t)$



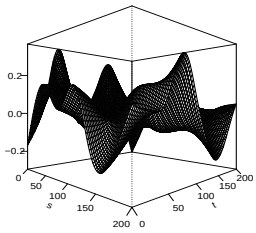
(g) $\hat{\gamma}_3(s,t)$



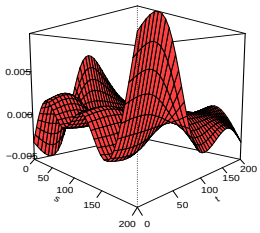
(h) $\hat{\gamma}_3(s,t)$



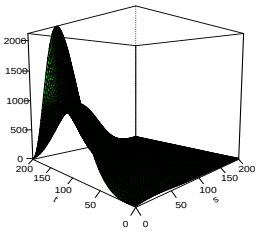
(i) $\hat{\gamma}_3(s,t)$



(j) $\hat{\gamma}_4(s,t)$



(k) $\hat{\gamma}_4(s,t)$



(l) $\hat{\gamma}_4(s,t)$

Figure 2.11: Estimates $\hat{\gamma}_j(s,t)$, $1 \leq j \leq 4$ for the three methods: *wSigcomp* (left column), integral PFFR (middle column) and integral PenFFR (right column).

Methods	$\widehat{\text{ISE}}$ ($\times 10^2$)
Integral PenFFR	0.57 (0.74)
Concurrent PenFFR	1.83 (0.88)
Integral pffr	2.37 (1.55)
Concurrent pffr	**0.52 (0.26)**
wSigcomp	4.79 (4.46)

Table 2.5: The average (and standard deviation) of $\widehat{\text{ISE}}$ for the Hawaii ocean data set. The best result is in boldface.

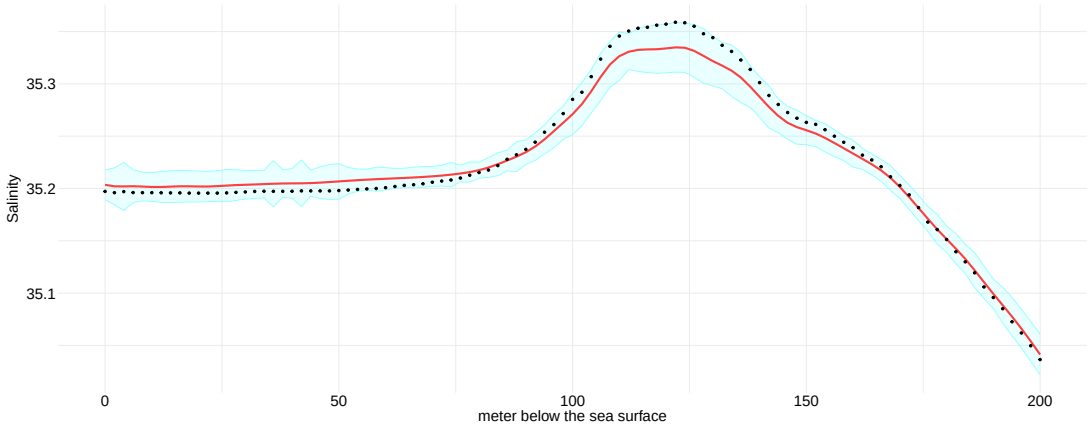


Figure 2.12: Prediction (red curve), actual data (black dots) and confidence interval (cyan region) for a randomly choose observation.

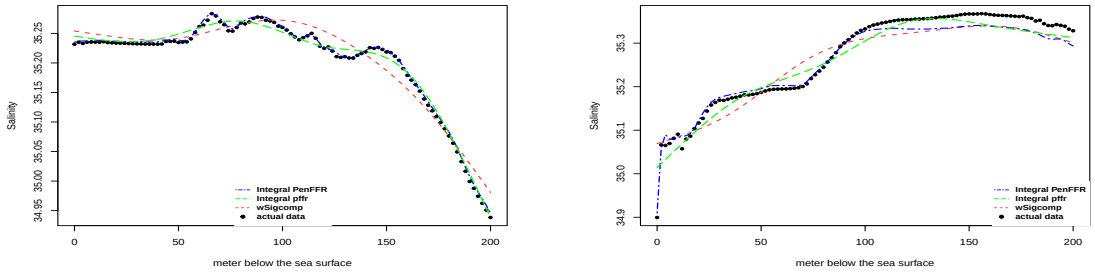


Figure 2.13: Prediction given by the three methods for integral model on two randomly chosen observations in the test sample: PenFFR in blue, PFFR in green and *wSigcomp* in red. Black dots are the true values.

2.7 Conclusion

In this chapter, we have presented a new estimation process for the linear regression model with functional responses and functional covariates. We approach the problem via expanding the functions onto a common B-spline basis, hence allowing the reduction of the functional model to a linear mixed model. Adaptation to unknown smoothness is performed by adding a roughness penalty on second derivatives. Unlike any estimator based on basis functions, our estimates have a smooth shape and sufficient flexibility to capture the encountered variability in various experiments with real-world data sets. We also design a framework for producing confidence interval bands for functional predictions. The method drawn a functional quantile regression by perturbing the standard linear regression and computed functional quantiles by optimal transport. This functional quantile regression is then combined to the conformalized method to build a functional conformalized quantiles regression. We then illustrate the performance of our proposed estimation process in terms of prediction accuracy and parameter interpretability on simulated and real data sets. As perspectives for future work, confidence bounds for predictions can be obtained in a functional mixture-of-Experts regression framework Tamo Tchomgui et al. (2024c). While these models offer flexibility and strong predictive performance, quantifying the uncertainty of their predictions remains a challenging and unexplored area.

Chapter 3

A Mixture of Experts Regression Model for Functional Response with Functional Covariates

This chapter was published in June 2024 in *Statistics and Computing* journal, volume 34 (Tamo Tchomgui et al., 2024c). We have reproduced the entire article as published. For this reason, some concepts may have been repeated, particularly in the introduction about the state of the art. The Section 3.2 is also a brief summary of Chapter 2. The paper addresses the problem of linear function-on-function regression of heterogeneous data in the predictive modelling. We have developed a MoE framework models for data where both response and covariate are functional.

Abstract: Due to the fast growth of data that are measured on a continuous scale, functional data analysis has undergone many developments in recent years. Regression models with a functional response involving functional covariates, also called "function-on-function", are thus becoming very common. Studying this type of model in the presence of heterogeneous data can be particularly useful in various practical situations. We mainly develop in this work a FFMoE regression model. Like most of the inference approach for models on functional data, we use basis expansion (B-splines) both for covariates and parameters. A regularized inference approach is also proposed, it accurately smoothes functional parameters in order to provide interpretable estimators. Numerical studies on simulated data illustrate the good performance of FFMoE as compared with competitors. Usefullness of the proposed model is illustrated on two data sets: the reference Canadian weather data set, in which the precipitations are modeled according to the temperature, and a Cycling data set, in which the developed power is explained by the speed, the cyclist heart rate and the slope of the road.

Keywords: Mixture of Experts (MoE), Functional regression, EM algorithm, Ridge regularized estimation.

3.1 Introduction

During the past few decades, functional data have become a very popular type of measurement in a constantly growing number of industrial, societal and medical applications. A branch of statistics, FDA, was developed as a specific discipline for analysing such data. FDA's flexibility in handling complex, high-dimensional, and structured data makes it applicable to a broad range of scientific and practical problems, providing insights that traditional data analysis methods may not be able to unveil. Broadly speaking, this new paradigm concerns the statistical analysis of data where at least one of the variables of interest is treated as a curve, surface or volume (also called function for simplicity) observed over a domain set. Most notable recent applications encompass, in particular, Healthcare and Medicine (monitoring patient health over time, FMRI data), Environmental Science (temperature or precipitation trends over time), Economics and Finance (evolution of stocks or commodities, modelling consumer behaviour over time), Sports Science (Analyzing athletes' performance data over time or during an event to optimize training and performance), Meteorology (analyzing weather patterns and trends to improve forecasting models), Chemometrics (analyzing spectroscopy data to identify and quantify chemical substances), Genomics and Bioinformatics (analyzing gene expression data over time), Traffic Analysis and Urban Planning.

Our main focus in the present work is the extension of linear regression to the functional data setting, a model which has naturally become a major area of research in the field of FDA. Standard references for FDA are Horváth and Kokoszka (2012); Kokoszka and Reimherr (2017); Ramsay and Silverman (2005); Ramsay et al. (2009). A broad overview of functional linear regression is given in Goldsmith et al. (2011) and Morris (2014). Using the convention that first term denotes the type of the response and second term denotes the type of the covariate three different setups have been analyzed in the literature: Function-on-Scalar, Scalar-on-Function and function-on-function. In the present work, we will focus on the most challenging setup from both statistical and computational perspectives, i.e. function-on-function regression problems. function-on-function regression problems have indeed been much less studied than the two other types of functional regression despite their relevance in many important applications. Recently, Ivanescu et al. (2015) proposed to estimate a function-on-function regression model using a penalized mixed model. A signal compression approach was also recently devised in Luo et al. (2016), based on preprocessing the functional covariates using their wavelet transform and on proposing a method to estimate the functional parameter by characterizing them as solutions to a generalized functional eigenvalue problem. In a vast majority of current works in this area, one of the main issues is how to accurately select the most statistically relevant number of basis functions, and the location of the knots for spline models Li and Ruppert (2008). Another important issue is the interpretability of the obtained estimators James et al. (2009). In Tamo Tchomgui et al. (2023b) proposed a Ridge-type penalization on second derivative of parameters using B-splines expansions for both functional covariates and parameters which is a first attempt at resolving the interpretability and model selection problems using convex sparsity-enforcing penalties.

Often in practice, the available data carry some heterogeneity, and the assumption that a unique relationship between the response variable and covariates holds for the full data set may not be valid. To circumvent this problem, a mixture of regression model can be proposed in DeSarbo and Cron (1988); McLachlan and Peel (2000). As we know (see Titterington et al. (1985) and Lindsay (1995)), mixture models are very powerful at capturing subpopulation behaviour, a crucial capability in most applications. Mixture models have been studied in many different setups and specific algorithms, such as EM-type unpenalized and penalized models have been devised for the

estimation of its parameters Dempster et al. (1977); McLachlan and Krishnan (2007). Accelerated versions using space alternating schemes Celeux et al. (2001) and proximal interpretations Chrétien and Hero (1998); Chrétien and Hero (2000). Sparsity-enforcing penalized versions were studied in Chrétien et al. (2012). Restrictions on relevant variables selected by an ℓ_1-penalized ML estimator is given in Devijver (2015).

Unfortunately, standard mixture models do not permit to parameterize the individual probability of each data to belong to a specific cluster. As this usually hampers the predictive capabilities of mixture models, the framework of MoE models was first suggested in Jacobs et al. (1991) as a powerful supervised learning procedure that can efficiently handle the potential heterogeneity often present in the data. The MoE model is based on a divide-and-conquer principle, which can be simply understood by realizing that each expert can specialize in smaller problems, and their predictive power can be combined together via a gating function in order to solve the full problem. The MoE model can also be viewed as a version of a multilayer supervised network in the sense that it is composed of K separate networks, each of which learning on a subset of the whole data data, as illustrated by Figure 3.1. From a more statistical learning perspective, the MoE model consists in a mixture model where both the mixture weights, a.k.a. Gating Functions, and component densities, a.k.a. Experts, depend on each data's covariate. The mixture model and its extension to MoE model has been investigated in the contexts of regression, clustering and discriminant analysis. A useful overview was proposed in Nguyen and Chamroukhi (2018), in which provides conditions for consistency and asymptotically normal properties are studied. Nevertheless, most MoE models only handle the scalar case. In the functional case, it would be relevant to implement efficient extensions of MoE model as well. This problem has already been tackled in Chamroukhi et al. (2022), but for scalar response. Our contribution is to extend the MoE model to the function-on-function setup and provide an efficient inference algorithm.

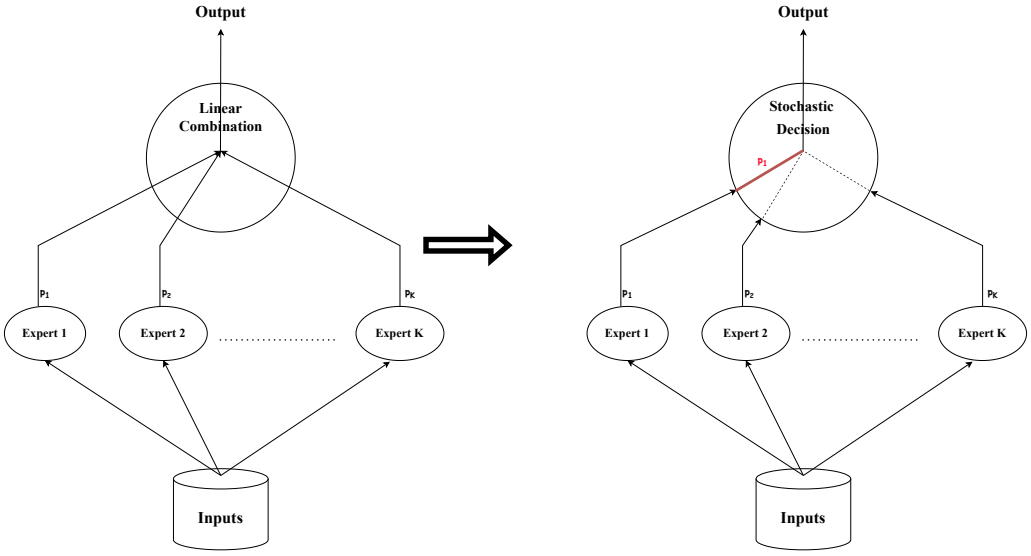
The paper is organised as follows: Section 3.3 presents the function-on-function MoE model we proposed and its inference. Section 3.4 describes how to implement a penalized version of the estimation scheme. Section 3.5 proposes extensive simulation experiments that explore the various aspects of the performance of the method. Section 3.6 finally presents an illustration of the method on two real-world data sets and shows the advantage, in terms of predictive quality, of considering MoE as compared with non-mixture-based approaches.

3.2 The concurrent model

3.2.1 The functional model

The problem under study consists in modelling the relationship between functional covariates $\mathrm{X}^1(t), \ldots, \mathrm{X}^p(t)$ and a functional response $\mathrm{Y}(t)$ based on a n-sample $\{\mathrm{Y}_i(t), \mathrm{X}_i^1(t), \ldots, \mathrm{X}_i^p(t), t \in [0, \mathrm{T}]\}, i = 1, \ldots, n$. The functional response and covariates are assumed to belong to the separable Hilbert space $\mathrm{L}^2([0; \mathrm{T}])$ endowed with the Lebesgue measure. In the present work, we focus on the concurrent model Ramsay and Silverman (2005) which assumes a linear relationship between the response and covariates, where the value of the response at a particular time stamp is modelled as a linear combination of the covariates at that specific time stamp, and the coefficients of the functional covariates are univariate smooth functions of time:

$$\mathrm{Y}_i(t) \quad = \quad \beta_0^\star(t) + \sum_{\ell=1}^{p} \beta_\ell^\star(t)\, \mathrm{X}_i^\ell(t) + \varepsilon_i(t) = \mathrm{X}_i(t)^\top \beta^\star(t) + \varepsilon_i(t), \tag{3.2.1}$$



(a) **(b)**

Figure 3.1: System of Experts and gating networks: The case of weighted linear combination (a) and the case of stochastic decision (b) to produce output.

with $\mathrm{X}_i(t) = (1 \;\; \mathrm{X}_i^1(t) \;\; \cdots \;\; \mathrm{X}_i^p(t))^\top$ and $\beta(t) = (\beta_0^\star(t) \;\; \beta_1^\star(t) \;\; \cdots \;\; \beta_p^\star(t))^\top$.
$\beta_\ell^\star(t)$ are the unknown functional parameters, assumed to be square integrable. The residuals $\varepsilon_i(t)$ are centered random variables with variance σ_i^2, specific to the i^{th} individual (Ramsay and Silverman (2005), Chapter 13).
The noise functions $\varepsilon_i(t)$ can also be rigorously defined using white noise theory as presented in Hida et al. (1993). In the framework of the present project, we will only use the property that when sampled at various times from a finite time set $\mathcal{T}$, the vector $(\varepsilon_i(t))_{t\in\mathcal{T}}$ can be expressed as a sum of a vector with independent and identically distributed (i.i.d.) components and a vector with prescribed covariance matrix, which can be a prescribed to a vector with constant components in the simplest case. Considering the concurrent model is of great interest because, as mentioned in Hastie and Tibshirani (1993), any functional linear model can be reduced to this form.

3.2.2 From functional to multivariate models

The parameters $\beta_\ell^\star(t)$ of Model (3.2.1) can be estimated using the method discussed in Tamo Tchomgui et al. (2023b), where the functional problem is rewritten as a classical multivariate regression problem by expanding the functional covariates and parameters into B-spline series, i.e.:

$$\mathrm{X}_i^\ell(t) = \sum_{j=1}^{q_{\mathrm{x}^\ell}} x_{ij}^\ell \, \mathrm{B}_j^\ell(t) = \mathrm{B}^\ell(t)^\top x_i^\ell \quad \text{and} \quad \beta_\ell^\star(t) = \sum_{j=1}^{q_{\beta^\ell}} b_j^{\star\ell} \, \phi_j^\ell(t) = \phi^\ell(t)^\top b^{\star\ell}, \tag{3.2.2}$$

where $\mathrm{B}^\ell(t) = (\mathrm{B}_j^1(t), \ldots, \mathrm{B}_j^{q_{\mathrm{x}^\ell}}(t))^\top$ is the q_{x^ℓ} -dimensional vector of basis functions for the covariate $\mathrm{X}^\ell(t)$ and $x_i^\ell = (x_{i1}^\ell, \ldots, x_{iq_{\mathrm{x}^\ell}}^\ell)$ the corresponding basis expansion coefficients. Analogously, $\{\phi^\ell(t), \; b^{\star\ell}\}$ are the basis functions and basis coefficients for $\beta_\ell^\star(t)$. Then, using the following notations :

- $\Phi(t) = (\phi^0(t)^\top \;\; \phi^1(t)^\top \;\; \cdots \;\; \phi^p(t)^\top)$, a vector of length $\sum_\ell q_{\beta^\ell}$,
- $b^\star = (b^{\star 0^\top} \;\; b^{\star 1^\top} \;\; \cdots \;\; b^{\star p^\top})^\top$, a vector of length $\sum_\ell q_{\beta^\ell}$,
- $B(t) = (1 \;\; B^1(t)^\top \;\; \cdots \;\; B^p(t)^\top)$, a vector of length $\sum_\ell q_{X^\ell}$,
- $x_i = (x_i^0(t)^\top \;\; x_i^1(t)^\top \;\; \cdots \;\; x_i^p(t)^\top)^\top$, a vector of length $\sum_\ell q_{X^\ell}$,

Model (3.2.1) can be written:

$$\mathrm{Y}_i(t) \;\; = \;\; x_i^\top \mathrm{B}(t)^\top \Phi(t) \; b^\star + \varepsilon_i(t) = \mathrm{R}_i(t)^\top b^\star + \varepsilon_i(t). \tag{3.2.3}$$

From this viewpoint, the concurrent model can be recast as a classical linear regression model with design matrix $\mathrm{R}_i(t) = \Phi(t)^\top \mathrm{B}(t) \, x_i$ and regression parameters b. When restricted to the observation grid consisting of the m successive timestamps $\{t_1, \ldots, t_m\}$, the problem reduces to:

$$\mathrm{Y}_i(t_j) = \mathrm{R}_i(t_j)^\top b^\star + \varepsilon_i(t_j) \quad \text{with } \; 1 \le i \le n \text{ and } 1 \le j \le m. \tag{3.2.4}$$

There is nevertheless one peculiarity with this approach to underline. Indeed, in Model (3.2.4) the random variables $\varepsilon_i(t_1), \ldots, \varepsilon_i(t_m)$ representing the noise can not be assumed independent. In order to circumvent this issue, one possible approach is to use a linear mixed model (LMM) as advocated in Wood (2006). For this purpose, we will assume that the model error can be

decomposed as $\varepsilon_i(t_j) = \mathrm{U}_i + \eta_{ij}$, with η_{ij} a Gaussian white noise and U_i a random variable which takes into account the random effect in each individual $i \in \{1, \ldots, n\}$. In this framework, the estimation procedure proposed in Tamo Tchomgui et al. (2023b) consists in maximizing the ridge-type penalised likelihood, with an ℓ_2-squared penalty on the second derivatives of $\beta_\ell^\star(t)$. Such a penalty is recommended when smooth estimates are sought for and provides sufficient flexibility that can still capture a substantial variety of complex shapes.

3.3 Mixture of experts of linear model for functional response with functional covariates

Mixture Regression (MR) models form a subset of the broad class of statistical models known as finite mixture models McLachlan and Peel (2000), which are designed to account for the statistical heterogeneity in a population through a finite set of empirical latent classes. MR models focus on identifying systematic differences between underlying latent groups in the population by the effect of covariates on the response. These models have to be distinguished from other mixture models that estimate the differences in levels and variance of the response variable between the groups (see DeSarbo and Cron (1988)). MR models assumes that there are $\mathrm{K} \in \mathbb{N}^\star$ mixture components in the population. Component membership is indicated by a latent categorical variable (one-hot encoding as) $\mathrm{Z} = (\mathrm{z}_1, \ldots, \mathrm{z}_\mathrm{K})$ where z_k takes the value 1 if the observation belongs to the component k and 0 otherwise. The MR model can written

$$\mathrm{MR}(\mathrm{Y}|\mathrm{X}) \quad = \quad \sum_{k=1}^{\mathrm{K}} \pi_k \mathbb{E}_k[\mathrm{Y}|\mathrm{X}, \mathrm{z}_k = 1] \tag{3.3.1}$$

where π_k is the mixture proportion of group k associated with the k-th expert $\mathbb{E}_k[\mathrm{Y}|\mathrm{X}]$. In the present functional case, this expert is defined by

$$\mathbb{E}_k[\mathrm{Y}(t)|\mathrm{X}(t), \mathrm{z}_k = 1] \quad = \quad \mathrm{X}(t)^\top \beta_k(t) \tag{3.3.2}$$

where $\beta_k(t) = (\beta_{k,0}(t), \beta_{k,1}(t), \ldots, \beta_{k,p}(t))$the functional parameters of the k^{th} expert.

Within the proposed model, there are two possible options for designing the probabilities π_k, $k = 1, \ldots, K$. The first one assumes that the covariates X are not related to latent classes Z: $\pi_k = \mathbb{P}(\mathrm{z}_k = 1)$. The second, and more general, assumes that Z depends on X: $\pi_k = \pi_k(\mathrm{X}) = \mathbb{P}(\mathrm{z}_k = 1 \mid \mathrm{X})$.

The conditional density of $\mathrm{Y}(t)$ according to the function-on-function Mixture of Expert (FF-MoE) model is

$$f(\mathrm{Y}(t)|\mathrm{X}(t), \Psi(t)) \quad = \quad \sum_{k=1}^{\mathrm{K}} \pi_k(\mathrm{X}(t), \alpha_k(t)) \Phi(\mathrm{Y}(t); \mathrm{X}(t)\beta_k(t), \sigma_k^2), \tag{3.3.3}$$

with

- $\pi_k(\mathrm{X}(t),\ \alpha_k(t))$ the mixture proportion of group k, also called the k^{th} gated network function, depending on the covariate $\mathrm{X}(t)$ through group specific functional parameter $\alpha_k(t)$. More details will be provided in the next section;
- $\Psi_k(t) = (\beta_k(t), \alpha_k(t))$ are the functional parameters;

- $\Phi(\mathrm{Y}(t); \mathrm{X}_i(t)\beta_k(t), \sigma_k^2)$ is the Gaussian density probability function of mean $\mathrm{X}(t)\beta_k(t)$ and variance σ_k^2.

3.3.1 Modelling the gated network function

The MoE model can be seen as a submodel of the Latent class model proposed by Dayton and Macready (1988) named concomitant-variable latent class model. Various models for gated network have been proposed in the past "non-functional" related literature. One instance is the version of Jacobs et al. (1991) where a multinomial logistic model is introduced. Another approach presented in Young and Hunter (2010) considers non parametric models. Turning to the functional setup, various authors have proposed extensions of the logistic regression model of Jacobs et al. (1991). Most of them assume in particular that the functional terms all belong to the space of real square integrable functions $\mathrm{L}^2([0,1])$. See for instance Mousavi and Sørensen (2018) for an overview. In Berrendero et al. (2023), it is shown that the functional nature of covariates raises important technical issues, some of them inherited from the non-functional setup but with higher complexities. Some of the more noticeable issues include the non-existence of maximum likelihood estimators under general conditions, a remedy being working in a tailored Reproducing Kernel Hilbert Space (RKHS).

In the present work, under the realisation $\mathrm{x}_i(t)$ of $\mathrm{X}(t)$, we consider the following gating softmax function:

$$\pi_k(\mathrm{x}_i(t), \alpha_k(t)) = \frac{\exp(h_k(\mathrm{x}_i(t),\ \alpha_k(t)))}{1 + \sum_{k'=1}^{\mathrm{K}-1} \exp(h_{k'}(\mathrm{x}_i(t),\ \alpha_{k'}(t)))}, \tag{3.3.4}$$

where

$$h_k(\mathrm{x}_i(t),\ \alpha_k(t)) \quad = \quad \int_{\mathrm{T}} \alpha_k^\top(s)\mathrm{x}_i(s)ds \tag{3.3.5}$$

with $\alpha_k(t) = (\alpha_{k,0}(t), \alpha_{k,1}(t), \ldots, \alpha_{k,p}(t))^\top$. Notice that, in this model, the mixture proportion is constant over time.

As for the other functional parameters, $\alpha_k(t)$ is assumed to have an expansion into a basis of functions of the form:

$$\alpha_{k,\ell}(t) = \sum_{j=1}^{\mathrm{L}_{\alpha^\ell}} a_{k,j}^\ell \varrho_j^\ell(t) = \varrho^\ell(t)^\top a_k^\ell.$$

Similarly as for $\beta(t)$ in (3.2.2), we can write $\alpha_k(t) = \varrho(t)a_k$ and Equation ((3.3.5)) becomes:

$$h_k(\mathrm{x}_i(t), \alpha_k(t)) = \int_{\mathrm{T}} a_k^\top \varrho(s)^\top \mathrm{B}(s)\ x_i ds = a_k^\top \underbrace{\int_{\mathrm{T}} \varrho(s)^\top \mathrm{B}(s) dt x_i}_{r_i} = a_k^\top r_i,$$

Thus Model (3.3.4) can be written:

$$\pi_k(\mathrm{x}_i(t), \alpha_k(t)) \quad = \quad \frac{\exp(a_k^\top r_i)}{1 + \sum_{k'=1}^{\mathrm{K}-1} \exp(a_{k'}^\top r_i)}. \tag{3.3.6}$$

To guarantee the identifiability of $\alpha_k(t) \in \mathrm{L}^2(\mathbb{R}^{p+1})$, $k = 1, \ldots, \mathrm{K}$, $\alpha_{\mathrm{K}}(t)$ is set to the null function (and hence a_{K} is set to null vector) (Jiang and Tanner, 1999).

3.3.2 Estimation of the functional MoE via the EM algorithm

In practice, as expected, we only have access to a set of (noisy) observations at the timestamps in the set $\{t_1, \ldots, t_m\}$. For an observation i belonging to component k, the k^{th} expert model is given by

$$\mathrm{y}_i(t_j) \quad = \quad \beta_{k,0}(t_j) + \sum_{\ell=1}^{p} \beta_{k,\ell}(t)\mathrm{x}_i^{\ell}(t_j) + \varepsilon_i(t_j) = \beta_k(t_j)^{\top}\mathrm{x}_i(t_j) + \varepsilon_i(t_j), \tag{3.3.7}$$

where $\beta_k(t) = (\beta_{0,k}(t), \beta_{1,k}(t), \ldots, \beta_{p,k}(t))^{\top}$ for $k = 1 \ldots \mathrm{K}$, are the unknown functional experts parameters and are assumed to be square integrable.

As in the simple regression case, the successive observed values of a realisation i can not be assumed statistically independent. The mixed model approach of (Wood, 2006) can again be put to work after decomposing the observation error as $\varepsilon_i(t_j) = \mathrm{U}_i + \eta_{ij}$, with η_{ij} a Gaussian white noise and U_i a random variable which accounts for the random effect in each individual observation $i = 1, \ldots, n$. To sum up, model (3.3.7) consists of a LMM with fixed effects b_k and random effect U_i. In matrix form, this yields:

$$\mathbf{Y} \quad = \mathrm{R}^{\top} b_k + \mathrm{W}\mathrm{U} + \eta, \tag{3.3.8}$$

where $\mathbf{Y} = (\mathrm{y}_1(t_1), \ldots, \mathrm{y}_1(t_m), \mathrm{y}_2(t_1), \ldots, \mathrm{y}_n(t_m))^{\top}$, $\mathrm{R} = (\mathrm{R}_i(t_j))_{i,j}$ the design matrix of dimension $q_\beta \times nm$ with $q_\beta = \sum_\ell q_{\beta^\ell}$, $\mathrm{U} = (\mathrm{U}_1, \mathrm{U}_2, \cdots, \mathrm{U}_n)^{\top} \sim \mathcal{N}(\mathbf{0}, \Gamma)$, $\eta = (\eta_{ij})_{i,j} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbb{I}_{nm})$ and

$$\mathrm{W} = \underbrace{\begin{pmatrix} 1_{m\times 1} & 0_{m\times 1} & \ldots & 0_{m\times 1} \\ 0_{m\times 1} & 1_{m\times 1} & \ldots & 0_{m\times 1} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{m\times 1} & 0_{m\times 1} & \ldots & 1_{m\times 1} \end{pmatrix}}_{(nm\times n)\text{ - matrix}}.$$

We will make use of the notations $0_{k\times l}$ (resp. $1_{k\times l}$) of size $k \times l$ for the matrices of zeros (resp. ones) and the notation $\mathbf{0}$ for the null vector. We will also denote by Γ the unknown covariance matrix of the random effects. $\mathbb{I}_{nm}$ refers to the $nm \times nm$ identity matrix.

The conditional density of $\mathbf{Y}$, given the observations is a mixture of K Gaussian distributions of mean $b_k^{\top}\mathrm{R}$ and variance $\mathrm{V}_k = \mathrm{W}\Gamma\mathrm{W}^{\top} + \sigma_k^2\mathbb{I}_{nm}$. So we have:

$$f(\mathbf{Y}|\mathbf{X}, \Psi) = \sum_{k=1}^{\mathrm{K}} \pi_k(\mathrm{x}_i(t), \alpha_k(t))\Phi_{nm}(\mathbf{Y}; b_k^{\top}\mathrm{R}, \mathrm{V}_k), \tag{3.3.9}$$

where $\mathbf{X}$ is defined in the same way as $\mathbf{Y}$. $\Phi_\ell(x; \mu, \Sigma)$ denotes the probability density function of the L-dimensional Gaussian distribution with mean vector μ and covariance matrix Σ. $\Psi = ((a_1, b_1, \sigma_1^2), \ldots, (a_{\mathrm{K}}, b_{\mathrm{K}}, \sigma_{\mathrm{K}}^2), \mathrm{U}, \Gamma)$ are the vector of parameters of the model to be estimated.

Inference of finite mixture model has been studied by various authors in the literature. We can mention for e.g. Jacobs et al. (1991); Jordan and Jacobs (1994) that compute Maximum Likelihood Estimators (MLE) via EM algorithm; Bayesian approaches have also been proposed as for instance in Peng et al. (1996); Young and Hunter (2010) present a parameter estimation approach in a semiparametric setting.

Now the FFMoE model can be defined using finite representation of functional terms. In this setting, we can easily write the observed data log-likelihood given by:

$$\mathcal{L}(\Psi; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) = \sum_{i=1}^{n} \log(\sum_{k=1}^{\mathrm{K}} \pi_k(\mathrm{x}_i(t), \alpha_k(t)) \Phi_m(\mathbf{y}_i; b_k^\top \mathrm{R}_i, \mathrm{V}_{k,i})) \tag{3.3.10}$$

where $\mathbf{y}_i$ is the vector of size m that contains all the measurements for observation i, R_i and $\mathrm{V}_{k,i}$ are respectively the design matrix and block covariance matrix of V_k associated with i. Then, the log-likelihood of Equation (3.3.10) becomes:

$$\sum_{i=1}^{n} \log(\sum_{k=1}^{\mathrm{K}} \frac{\exp(a_k^\top r_i)}{1+\sum_{u=1}^{\mathrm{K}-1} \exp(a_u^\top r_i)} . \frac{1}{\sqrt{(2\pi)^m |\mathrm{V}_{k,i}|}} \exp(-\frac{1}{2}(\mathbf{y}_i - b_k^\top \mathrm{R}_i)^\top \mathrm{V}_{k,i}^{-1}(\mathbf{y}_i - b_k^\top \mathrm{R}_i))).$$

As is well known in Finite Mixture Models, the log-likelihood maximisation problem is cumbersome to address without introducing clever intermediate steps that form the philosophy of EM-type algorithms, as extensively discussed in the landmark paper Dempster et al. (1977). A basic requirement for the method is to complete the data by imputing latent group membership variables z_i for each observation $i = 1 \dots n$. These latent variables are represented by K binary variables $(z_{i1}, z_{i2}, \dots, z_{i\mathrm{K}})$. This model is called a complete model and leads to the complete data log-likelihood given by:

$$\begin{aligned} \mathcal{L}_c(\Psi; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) = & \sum_{i=1}^{n} \sum_{k=1}^{\mathrm{K}} z_{ik} \log(\frac{\exp(a_k^\top r_i)}{1+\sum_{u=1}^{\mathrm{K}-1} \exp(a_u^\top r_i)} . \frac{1}{\sqrt{(2\pi)^m |\mathrm{V}_{k,i}|}} \\ & \exp(-\frac{1}{2}(\mathbf{y}_i - b_k^\top \mathrm{R}_i)^\top \mathrm{V}_{k,i}^{-1}(\mathbf{y}_i - b_k^\top \mathrm{R}_i))). \end{aligned} \tag{3.3.11}$$

Let $\Psi^{(0)} = ((a_1^{(0)}, b_1^{(0)}, \sigma_1^{2^{(0)}}), \dots, (a_{\mathrm{K}}^{(0)}, b_{\mathrm{K}}^{(0)}, \sigma_{\mathrm{K}}^{2^{(0)}}), \mathrm{U}^{(0)}, \Gamma^{(0)})$ be an initial estimate of Ψ. The EM algorithm is a generic process consisting of repeating two steps to updates parameters such that the log-likelihood value monotonically increases:

- **E-step:** At this step, we compute the conditional expectation of the log-likelihood given the observed data and the current parameter (at iteration l) estimation $\Psi^{(l)}$.
  So we define the Q function for the EM algorithm defined by:

$$\mathrm{Q}(\Psi^{(l+1)}|\Psi^{(l)}) = \mathbb{E}(\mathcal{L}_c(\Psi^{(l+1)}; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) | \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j};\ \Psi^{(l)}). \tag{3.3.12}$$

  This consists of computing the posterior probabilities $p_{ik}^{(l)}$ that the curves i-th sample $(y_i(t), \mathrm{x}_i(t))$ belongs to the k^{th} component of the mixture under the current model:

$$p_{ik}^{(l)} = \mathbb{E}(z_{ik} | \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}; \Psi^{(l)}) = \mathbb{P}(z_{ik} = 1 | \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}; \Psi^{(l)}).$$

  Using Bayes' theorem, this conditional probability $p_{ik}^{(l)}$ can be expressed as:

$$p_{ik}^{(l)} = \frac{\pi_k(\mathrm{x}_i(t), \alpha_k^{(l)}(t)) \Phi_m(\mathbf{y}_i; b_k^{\top(l)} \mathrm{R}_i, \mathrm{V}_{k,i}^{(l)}, t \in \mathrm{T})}{\sum_{u=1}^{\mathrm{K}} \pi_u(\mathrm{x}_i(t),\ \alpha_u^{(l)}(t)) \Phi_m(\mathbf{y}_i; b_u^{\top(l)} \mathrm{R}_i, \mathrm{V}_{u,i}^{(l)})}. \tag{3.3.13}$$

- **M-step:** Given the previous conditional probability and the observed data, this step updates the current parameters $\Psi^{(l)}$ by maximizing the conditional expectation of the complete data log-likelihood, that is $\Psi^{(l+1)}$:

$$\begin{aligned} \mathrm{Q}(\Psi^{(l+1)}|\Psi^{(l)}) &= \mathbb{E}(\mathcal{L}_c(\Psi^{(l+1)}; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j})|\{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j};\ \Psi^{(l)}) \\ &= \mathrm{Q}_1(a_k^{(l+1)}|\Psi^{(l)}) + \mathrm{Q}_2(b_k^{(l+1)},\ \mathrm{V}_k^{(l+1)}|\Psi^{(l)}). \end{aligned} \tag{3.3.14}$$

  The EM algorithm was shown to be a particular case of the celebrated Proximal Point algorithm in Chrétien and Hero (1998); Chrétien and Hero (2000) using a Kullbak-Leibler type divergence for the proximity term. Another interesting interpretation in terms of alternating minimisation in given in Neal and Hinton (1998). Space alternating version of the EM algorithms where proposed in Celeux et al. (2001); Fessler and Hero (1994) and Chrétien et al. (2012) for the nonsmoothly penalised case. In this paper, the maximisation of (Q) will be performed using a modified version of the **R** package (Grün and Leisch, 2008a): in particular, the function `initFlexmix` which allows repeating the EM algorithm with different starting values and choosing the solution with the highest value of the likelihood while allowing concomitant variables, as developed in Grün and Leisch (2008b). The global maximisation problem is split onto two separate maximisation problems (see Appendix B.1 for details):

  - the updating of gated network parameters via the maximisation of the function $\mathrm{Q}_1(a_k^{(l+1)}|\Psi^{(l)})$ and
  - the updating of the expert's parameters via the maximisation of the function $\mathrm{Q}_2(b_k^{(l+1)},\ \mathrm{V}_k^{(l+1)}|\Psi^{(l)})$.

  One will easily recognise in each of these two expressions, the likelihood of the multinomial logistic model $\mathrm{Q}_1(.)$ and of the linear Gaussian model $\mathrm{Q}_2(.)$ for which we know how to compute (at least numerically using e.g. Newton-Raphson iterations) the MLEs.

- The E and M steps are alternated repeatedly until numerical convergence i.e. the difference $\mathcal{L}(\Psi^{(l+1)}; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) - \mathcal{L}(\Psi^{(l)}; \{y_i(t_j); \mathrm{x}_i(t_j)\}_{i,j})$ changes by no more than an arbitrarily small value.

Stability and convergence properties of the method are established in the literature (see McLachlan and Krishnan (2007) for an overview and Chrétien and Hero (2000) for the proximal viewpoint).

With the estimates of gated network and experts parameters obtained, a hard-clustering of the link between $\mathrm{X}(t)$ and $\mathrm{Y}(t)$ is reach using Bayes' rule so that

$$\widehat{z}_{ik} = \begin{cases} 1 & \text{if } k = \mathrm{Arg}\max_{1 \leq k \leq \mathrm{K}} p_{ik}, \\ 0 & \text{otherwise,} \end{cases} \qquad i = 1 \dots n.$$

where p_{ik} is the value of Equation (3.3.13) at convergence.

3.3.3 Model selection

One important challenge in statistical estimation with potentially several possible models depending on hyperparameters is the selection of the most statistically relevant one. In the present model, choosing the correct number of components K is one crucial step of the estimation problem. In the

regression setting, the selection can be done using information criteria such as AIC Akaike (1974) or BIC Schwarz (1978), or using cross validation methods. The latter being time-consuming, we will use information criterion based approaches and more specifically the BIC criterion usually defined using log-likelihood (3.3.10) as:

$$\text{BIC} \quad = \quad \text{-}2\mathcal{L}(\Psi; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) - d\log(n) \tag{3.3.15}$$

where $d = \mathrm{K} \times (1 + \sum_{\ell=0}^{p} \mathrm{L}_{\beta^\ell} + \sum_{\ell=1}^{p} \mathrm{L}_{\alpha^\ell})$ is the number of free parameters of the model and n the number of observations.

3.3.4 Prediction

As we have already mentioned, one of the major limitations of simple mixture models is predictive modelling. Since for a new individual, its prediction will be given by the weighted sum of the predictions of each class. This is so far not ideal as this prediction is entirely driven by the prediction of the most probably class and these class probabilities will not change whatever the characteristics of the new individual. With the MoE model, we have seen that we can make this latent class probability depending on the covariates (concomitant variables). In this case, the prediction is given by expert prediction of the most probable class. To build such predictions we first need the conditional probabilities that any individual i belongs to a component k given by:

$$\pi_k(\mathrm{X}_i(t), \widehat{\alpha}_k) = \frac{\exp(\widehat{a}_k^\top r_i)}{1 + \sum_{v=1}^{\mathrm{K}-1} \exp(\widehat{a}_v^\top r_i)}$$

where $\widehat{a}_k$ for $1 \leq k \leq \mathrm{K} - 1$ are the gated parameters estimators.

We deduce, where component k_m is the most probable class for the i curve, the predictive curve by:

$$\widehat{\mathrm{Y}}_i(t) = b_{k_m}^\top \mathrm{R}_i(t).$$

As a result, estimating the group membership from covariates is essential to predict the response well.

3.4 Regularizing the function-on-function mixture of experts regression

In the FFMoE model (3.3.3) presented in Section 3.3, it is assumed that the functional covariates and parameters can be decomposed into a finite dimensional functional basis. This assumption allows to get the finite representation (3.3.9). The numbers of basis functions of each parameters and covariates should be correctly selected in order to avoid over- or under-fitting. Nevertheless, precise adjustment of these values often induces a high computational effort. In the case of the B-spline basis, even more parameters have to be properly tuned such as the choice of the spline order and the location of the knots. In order to reduce the expected cost of such a computationally demanding procedure, we made the choice of choosing a sufficiently large a priori value for L_β (or L_α) and then apply a penalty. This approach brings the benefit of tuning a single hyperparameter,

which is the number of basis functions and improving the smoothness and then interpretability of the estimated functional coefficients. This last point is very interesting in the case of the linear model because as we already know, the interpretation of the predictors-response relationship becomes more difficult as the shape of the functional parameter $\beta(t)$ (or $\alpha(t)$) does not have any simple structure.

Various approaches to regularize the parameter shape have been proposed in the literature. In our setting of interest, the main goal is to enhance the shape of parameters and then interpretability. Leurgans et al. (1993) are among the first to explore the functional penalization and show that the obtained estimator are less sensitive to the rather subjective choice of the number of basis functions. James et al. (2009) proposed a method called functional linear regression that is interpretable (FLiRTI) which address the issue of choosing relevant penalties. Based on variable selection ideas such as the Lasso penalty, FLiRTI produces accurate, flexible and highly interpretable estimates of the functional parameters. The main idea of FLiRTI method is, instead of enforcing sparsity on the function themselves, to enforce sparsity of the derivatives. Using the notation $\beta^{(l)}(t)$ for the l^{th} derivative of $\beta(t)$, we may deduce that $\beta^{(0)}(t) = 0$ guarantees $\mathrm{X}(t)$ has no effect on $\mathrm{Y}(t)$ at t; $\beta^{(1)}(t) = 0$ implies that $\beta(t)$ is constant at t; $\beta^{(2)}(t) = 0$ means that $\beta(t)$ is linear at t and so on.

3.4.1 Ridge-type penalty on second derivatives

Instead of the Lasso penalty, we proposed to estimated the functional MoE model (3.3.9) by maximizing a Ridge-type penalized log-likelihood. The penalty is based on the second derivative of the functional parameters (both gated and experts). This choice is mainly motivated by the desire to obtain a possibly locally constant relationship if needed. Moreover, the use the ridge penalty is motivated by the lack of exact sparsity observed in real problems and the clear benefits of getting a closed form formula for the estimators.

The corresponding penalized (data) log-likelihood function for the observed data is defined using (3.3.10) by:

$$\mathcal{L}_{pen}(\Psi; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) \quad = \quad \mathcal{L}(\Psi; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) + \mathrm{Pen}(\Psi), \tag{3.4.1}$$

in which the Ridge regularization term is given by

$$\mathrm{Pen}(\Psi) = \sum_{k=1}^{\mathrm{K}} \sum_{\ell=0}^{p} \lambda_{k,\ell} \int \beta_{k,\ell}^{''}(t)^2 dt + \sum_{k=1}^{\mathrm{K}-1} \sum_{\ell=1}^{p} \gamma_{k,\ell} \int \alpha_{k,\ell}^{''}(t)^2 dt$$

where

$$\int \beta_{k,\ell}^{''}(t)^2 dt = \int \left[\sum_{j=1}^{\mathrm{L}_{\beta\ell}} b_{k,j}^{\ell} \phi_j^{\ell''}(t) \right]^2 dt = \sum_{s,u=1}^{\mathrm{L}_{\beta\ell}} b_{k,s}^{\ell} b_{k,u}^{\ell} \Gamma_{su}^{\ell}$$

with $\Gamma_{su}^{\ell} = \int \phi_s^{\ell''}(t) \phi_u^{\ell''}(t) dt$, and

$$\int \alpha_{k,\ell}^{''}(t)^2 dt = \int \left[\sum_{j=1}^{\mathrm{L}_{\alpha\ell}} a_{k,j}^{\ell} \varrho_j^{\ell''}(t) \right]^2 dt = \sum_{s,u=1}^{\mathrm{L}_{\beta\ell}} a_{k,s}^{\ell} a_{k,u}^{\ell} \Upsilon_{su}^{\ell}$$

with $\Upsilon^{\ell}_{su} = \int \varrho^{\ell"}_{s}(t)\varrho^{\ell"}_{u}(t)dt$.
So,

$$\text{Pen}(\Psi) = \sum_{k=1}^{\text{K}}\sum_{\ell=0}^{p}\lambda_{k,\ell}\sum_{s,u=1}^{\text{L}_{\beta\ell}} b^{\ell}_{k,s}b^{\ell}_{k,u}\Gamma^{\ell}_{su} + \sum_{k=1}^{\text{K}-1}\sum_{\ell=1}^{p}\gamma_{k,\ell}\sum_{s,u=1}^{\text{L}_{\beta\ell}} a^{\ell}_{k,s}a^{\ell}_{k,u}\Upsilon^{\ell}_{su} \tag{3.4.2}$$

where $\lambda_{k,\ell}$ and $\gamma_{k,\ell}$ are the usual tuning regularization parameters which control the importance we want to place on the smoothness of estimators. As we know, selecting a good value of $\lambda_k = (\lambda_{k,\ell})_\ell$ (resp. $\gamma_k = (\gamma_{k,\ell})_\ell$) is very important to reduce the noise that less influential covariates create.

By using matrix terms, we get:

$$\mathcal{L}_{pen}(\Psi; \{y_i(t_j), \text{x}_i(t_j)\}_{i,j}) = \mathcal{L}(\Psi; \{y_i(t_j), \text{x}_i(t_j)\}_{i,j}) - \sum_{k=1}^{\text{K}} b_k^{\top}(\lambda_k \text{P})\, b_k - \sum_{k=1}^{\text{K}-1} a_k^{\top}(\gamma_k \text{Q})\, a_k$$

where $(\lambda_k \text{P}) \in \mathbb{R}^{\text{L}_\beta \times \text{L}_\beta}$ is given by:

$$(\lambda_k \text{P}) = \begin{pmatrix} \lambda_{k,0}\Gamma^0 & 0_{\text{L}_{\beta^0}\times\text{L}_{\beta^1}} & \dots & 0_{\text{L}_{\beta^0}\times\text{L}_{\beta^p}} \\ 0_{\text{L}_{\beta^1}\times\text{L}_{\beta^0}} & \lambda_{k,1}\Gamma^1 & \dots & 0_{\text{L}_{\beta^1}\times\text{L}_{\beta^p}} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{\text{L}_{\beta^p}\times\text{L}_{\beta^0}} & 0_{\text{L}_{\beta^p}\times\text{L}_{\beta^1}} & \dots & \lambda_{k,p}\Gamma^p \end{pmatrix} \quad \text{with } \Gamma^{\ell} = \begin{pmatrix} \Gamma^{\ell}_{11} & \Gamma^{\ell}_{12} & \dots & \Gamma^{\ell}_{1\text{L}_{\beta\ell}} \\ \Gamma^{\ell}_{21} & \Gamma^{\ell}_{22} & \dots & \Gamma^{\ell}_{2\text{L}_{\beta\ell}} \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma^{\ell}_{\text{L}_{\beta\ell}1} & \Gamma^{\ell}_{\text{L}_{\beta\ell}2} & \dots & \Gamma^{\ell}_{\text{L}_{\beta\ell}\text{L}_{\beta\ell}} \end{pmatrix};$$

and $(\gamma_k \text{Q}) \in \mathbb{R}^{\text{L}_\alpha \times \text{L}_\alpha}$ by:

$$(\gamma_k \text{Q}) = \begin{pmatrix} \gamma_{k,0}\Upsilon^0 & 0_{\text{L}_{\alpha^0}\times\text{L}_{\alpha^1}} & \dots & 0_{\text{L}_{\alpha^0}\times\text{L}_{\alpha^p}} \\ 0_{\text{L}_{\alpha^1}\times\text{L}_{\alpha^0}} & \gamma_{k,1}\Upsilon^1 & \dots & 0_{\text{L}_{\alpha^1}\times\text{L}_{\alpha^p}} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{\text{L}_{\alpha^p}\times\text{L}_{\alpha^0}} & 0_{\text{L}_{\alpha^p}\times\text{L}_{\alpha^1}} & \dots & \gamma_{k,p}\Upsilon^p \end{pmatrix} \quad \text{with } \Upsilon^{\ell} = \begin{pmatrix} \Upsilon^{\ell}_{11} & \Upsilon^{\ell}_{12} & \dots & \Upsilon^{\ell}_{1q_{\alpha\ell}} \\ \Upsilon^{\ell}_{21} & \Upsilon^{\ell}_{22} & \dots & \Upsilon^{\ell}_{2\text{L}_{\alpha\ell}} \\ \vdots & \vdots & \ddots & \vdots \\ \Upsilon^{\ell}_{\text{L}_{\alpha\ell}1} & \Upsilon^{\ell}_{\text{L}_{\alpha\ell}2} & \dots & \Upsilon^{\ell}_{\text{L}_{\alpha\ell}\text{L}_{\alpha\ell}} \end{pmatrix}.$$

Here, $0_{\text{L}_1\times\text{L}_2}$ is the standard notation for the null matrix of size $\text{L}_1 \times \text{L}_2$. As Γ^{ℓ} (resp. Υ^{ℓ}) is a symmetric positive-definite matrix for any $0 \leq \ell \leq p$, we can easily find its Cholesky decomposition, which can be efficiently leveraged in the implementation.

And for the penalized complete (data) log-likelihood, we made the same process and by using (3.3.11) we get:

$$\mathcal{L}^{c}_{pen}(\Psi; \{y_i(t_j), \text{x}_i(t_j)\}_{i,j}) = \mathcal{L}_c(\Psi; \{y_i(t_j), \text{x}_i(t_j)\}_{i,j}) - \sum_{k=1}^{\text{K}} b_k^{\top}(\lambda_k \text{P})\, b_k - \sum_{k=1}^{\text{K}-1} a_k^{\top}(\gamma_k \text{Q})\, a_k. \tag{3.4.3}$$

3.4.2 Penalized Maximum Likelihood Estimation via the EM algorithm

The EM algorithm for the regularized FFMoE is developed for maximizing the penalized (data) log-likelihood (3.4.3). The algorithm is simply the same as in non penalized version with small

changes. The E-step is exactly the same and the M-step is done by splitting the problem into two maximization problems as (see Appendix B.2 for details):

$$\begin{aligned} \mathrm{Q}_{pen}(\Psi^{(l+1)}|\Psi^{(l)}) &= \mathbb{E}(\mathcal{L}^{c}_{pen}(\Psi^{(l+1)})|y(t),\ \mathrm{x}(t);\ \Psi^{(l)}) \\ &= \mathrm{Q}_{1,pen}(a_k^{(l+1)}|\Psi^{(l)}) + \mathrm{Q}_{2,pen}(b_k^{(l+1)},\ \sigma_k^{2\ (l+1)}|\Psi^{(l)}). \end{aligned} \tag{3.4.4}$$

3.5 Simulation study of function-on-function mixture of experts models

The goal of this section if to evaluate, on the basis of simulated data, the proposed model in the case of function-on-function regression model. The data simulation process if inspired of Chamroukhi et al. (2022).

3.5.1 Data simulation process

100 data sets are simulated according to the FFMoE model with $K = 3$ components and $p = 1$ covariate, on a time domain $[0, 1]$.
The covariate is simulated with $\mathrm{X}_i(t) = x_i^\top \mathrm{B}(t)$, where $x_i = W.v_i$ with W a $10 \times 10-$matrix of $\mathcal{U}(0, 1)$, v_i a $10-$vector of $\mathcal{N}(0, 10)$ and $\mathrm{B}(t)$ is a 10-dimensional B-splines basis.
The functional parameters are $\beta_{1,0}(t) = -5t$, $\beta_{2,0}(t) = 0$ and $\beta_{3,0}(t) = 5t$, $\beta_{2,1}(t) = -\beta_{1,1}(t)$, $\beta_{3,1}(t) = 100(t - 0.5)^2 - 10$ and

$$\beta_{1,1}(t) = \begin{cases} -50(t-0.5)^2 + 2 & \text{if } 0 \leq t < 0.3 \\ 0 & \text{if } 0.3 \leq t < 0.7 \\ 50(t-0.5)^2 - 2 & \text{if } 0.7 \leq t < 1 \end{cases}$$

The functional parameter of the gated network are $\alpha_{1,0} = \alpha_{2,0} = -10$, $\alpha_{3,0}) = 0$, $\alpha_{1,1}(t) = 80(t - 0.5)^2 - 8$, $\alpha_{2,1}(t) = -\alpha_{1,1}(t)$ and $\alpha_{3,1}(t) = 0$.
Finally, the residuals are simulated with $\varepsilon_i(t) \sim \mathcal{N}(0, 4)$.

The number n of observations and the number m of sampling points are given in Table 3.1, defining thus four scenarios.

Scenarios	sampling level: m	number of observations: n
S1	20	300
S2	20	800
S3	100	300
S4	100	800

Table 3.1: The four scenarios of the simulation study

Figure 3.2 plots the discrete covariate observations (left panel) and their corresponding B-splines smoothing (right panel) for Scenario 3.

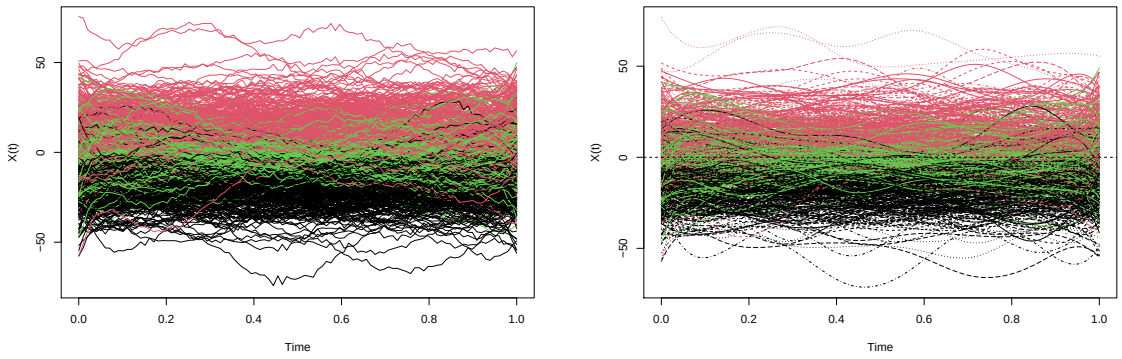
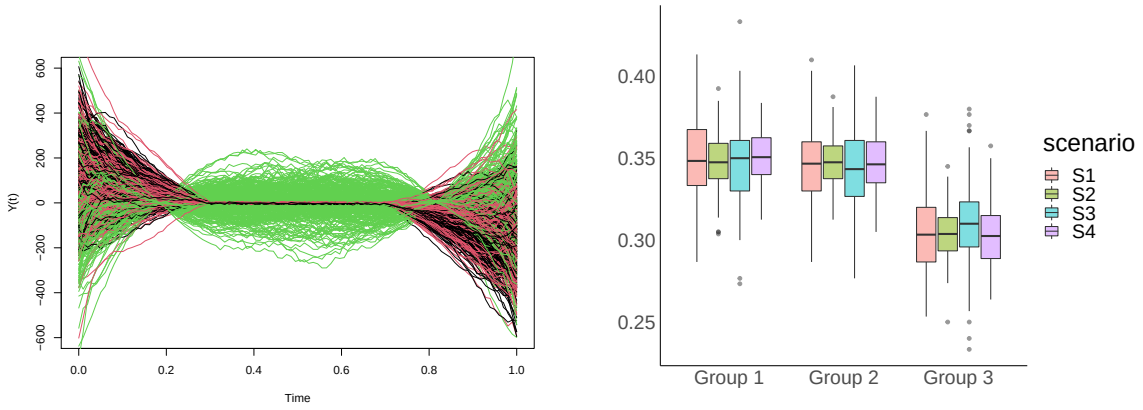


Figure 3.2: Discrete observations (left) and cubic B-splines smoothing (right) of the functional covariate. Color depends on the component membership.

Figure 3.3 plot the discrete observations of the output $Y(t)$ (left) for Scenario 3, and the proportions of observations of each component on the mixture (right) for the four Scenario.



(a) **(b)**

Figure 3.3: Discrete observations of the functional output (left) and proportions of observations of each component on the mixture (right).

3.5.2 Assessment criteria of goodness of fit

The assessment of the proposed FFMoE model is evaluated through two indicators: the quality of the parameter estimation, the quality of the prediction. In addition, the efficiency of BIC for selecting the number of components is also investigated.

The quality of parameter estimation is evaluated with

$$\text{MSE}\Big(\beta_l(.)\Big) = \left[\frac{1}{m}\sum_{j=1}^{m}\Big(\beta_l(t_j) - \widehat{\beta}_l(t_j)\Big)^2\right]^{1/2}. \tag{3.5.1}$$

Knowing that the label switching problem sometimes occurs, we will take care to re-label the clusters on the basis of the confusion matrix. Let notice that this criteria can be computed only when the true number of mixture component has been selected.
The quality of prediction is evaluated through the Mean Relative Prediction Error (MRPE) on a generate test sample of length $n_{test} = 2000$ for each scenario:

$$\text{MRPE} \quad = \quad \frac{1}{m}\sum_{j=1}^{m}\left(\frac{\sum_{i=1}^{n_{test}}\Big(\text{Y}_i(t_j) - \widehat{\text{Y}}_i(t_j)\Big)^2}{\sum_{i=1}^{n_{test}}\text{Y}_i(t_j)^2}\right). \tag{3.5.2}$$

Let notice that this criterion can be highly perturbed if the observation is associated to the wrong expert. Consequently, two additional criteria will be defined: MRPE.good, computed only of the observations associated to the correct expert, and MRPE.bad for thoses associated to a wrong expert.

3.5.3 Competitors

The competitors are the non-mixture penalized function-on-function regression models PenFFR (and its non penalized version, FFR) (Tamo Tchomgui et al., 2023b) and pffr (Ivanescu et al., 2015).

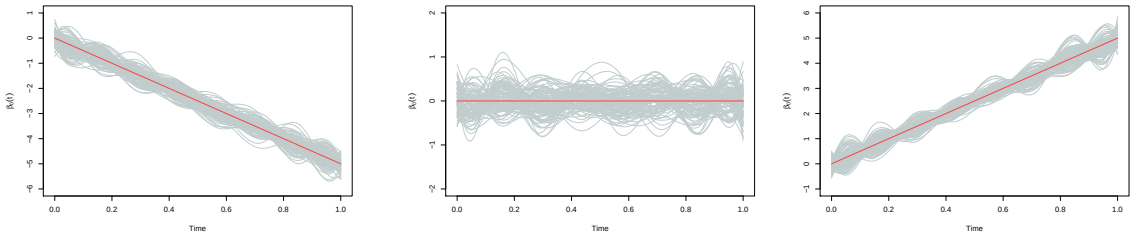
The FFR (resp. PenFFR) estimation process uses basis expansion of functional covariates and parameters to transform a functional model to multivariate. Estimation scheme is achieved by maximising the (resp. penalised) log-likelihood using a ridge-type penalty on the second derivatives. Cubic B-splines basis functions were employed for both for functional covariates and the functional parameters. The number of basis functions was set to 10 for both the functional parameters and covariates. This number was selected for leading to perfect reconstruction of the functional covariates.

The pffr estimation process uses observed values of functional covariates. An approach that matches with densely or sparsely sampled functions. The functional parameters is estimated using restricted maximum likelyhood (REML) in an associated mixed model. For the implementations of the method, we used default settings of the `pffr` function available in the `R` package `refund`. We only set the number of basis functions to 10 both for functional covariates and parameters.

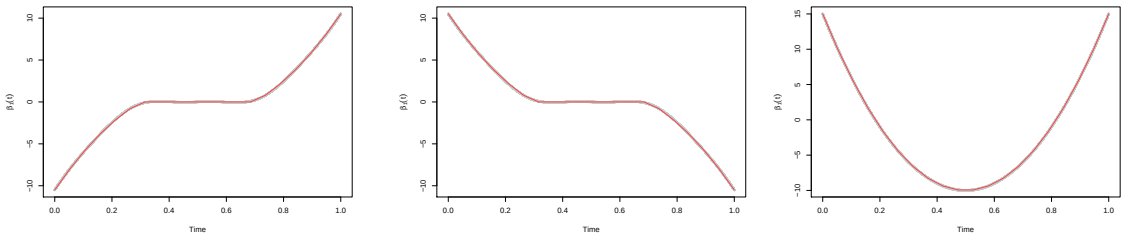
Finally, for FFMoE and PenFFMoE, we also set the number of basis functions to 10 both for both for functional covariates and parameters.

3.5.4 Simulation results

Concerning the ability of BIC to select the right number of components, BIC selects the correct number $K = 3$ in 100% of the case for the four scenarios, and thus both for FFMoE and PenFFMoE. We now present our empirical study of the computational complexity, estimation accuracy and prediction accuracy.



(a) $\beta_{1,0}(t)$ **(b)** $\beta_{2,0}(t)$ **(c)** $\beta_{3,0}(t)$



(d) $\beta_{1,1}(t)$ **(e)** $\beta_{2,1}(t)$ **(f)** $\beta_{3,1}(t)$

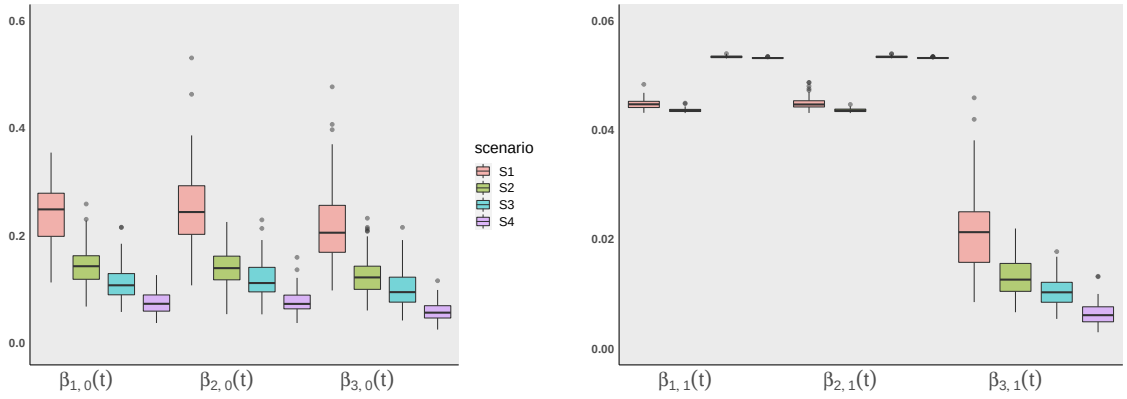
Figure 3.4: Estimation of the regression coefficients for Scenario S3 with FFMoE. The red curves are the actual parameters, the gray curves are the estimation.

Computation time

In simulation scenario S1, the average execution time for the various models is measured on a Dell computer with 250GB of RAM memory, a processor Intel Xeon w7-2495X processor (48 Thread CPU with 45MB Cache operating up to 4.8GHz), a Windows 11 Pro operating system and a R version 4.3.2 with necessary packages. The different results are as follows: PenFFMoE took 249.74 ± 65.16 seconds, FFMoE took 41.64 ± 18.78 seconds, FFR took 1.77 ± 0.08, PenFFR took 16.92 ± 0.24 seconds, and pffr took 1.46 ± 0.23 seconds. The computation time for the mixture models (FFMoE and PenFFMoE) is larger due to the additional required task of selecting the number K of clusters, $K = 1, \ldots, 5$. Additionally, PenFFMoE requires cross-validation to select the optimal penalty parameter λ_ℓ.

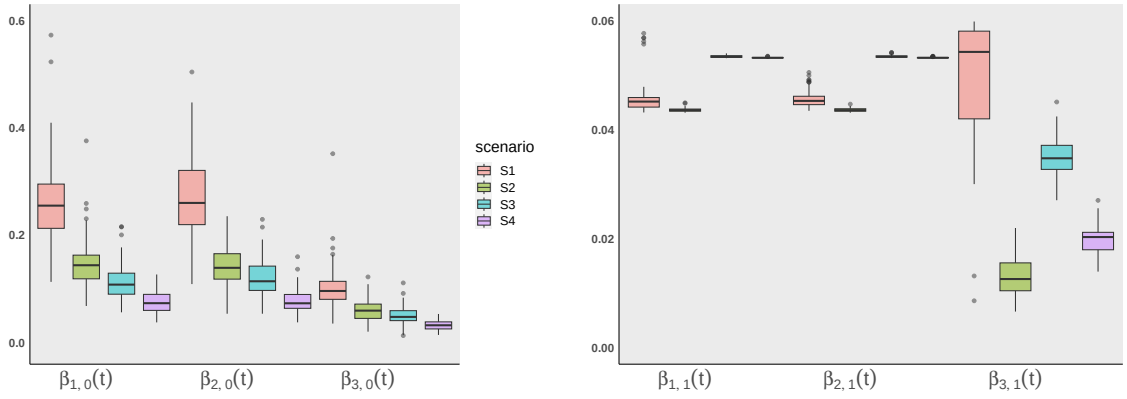
Parameter estimation

The relevance of our model is reflected by the parameter estimation. Figure 3.4 for FFMoE and Figure B.1 for PenFFMoE in appendix show the estimated versus actual parameters from Scenario S3. The estimation of the covariate effect is remarkably accurate. This observation is also supported by the MSE values reported in boxplot given in Figure 3.6 for PenFFMoE and Figure 3.5 for FFMoE in all scenarios.



(a) Intercept **(b)** Effect of X

Figure 3.5: Boxplot of MSE between actual and estimated parameters for FFMoE. Functional intercept $\beta_0(t)$ (left) and functional effect $\beta_1(t)$ of $\mathrm{X}(t)$ (right) in each of the 3 components mixture for our 4 simulated scenarios.



(a) Intercept **(b)** Effect of X

Figure 3.6: Boxplot of MSE between actual and estimated parameters for PenFFMoE. Functional intercept $\beta_0(t)$ (left) and functional effect $\beta_1(t)$ of $\mathrm{X}(t)$ (right) in each of the 3 components mixture for our 4 simulated scenarios.

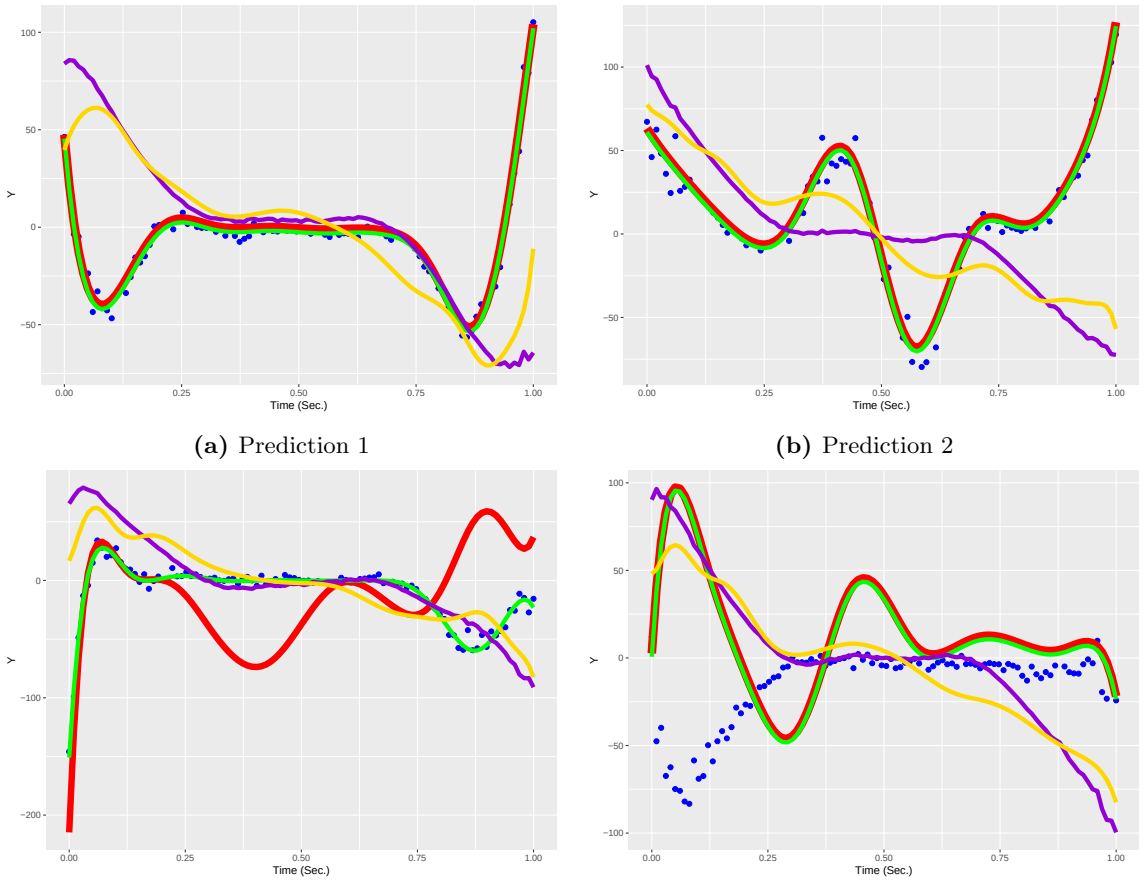
		Expert affectation accuracy	MRPE.good	MRPE.bad	MRPE
S1	FFMoE	91.3% (0.014)	0.006 ($<10^{-4}$)	2.560 (0.30)	0.230 (0.07)
	PenFFMoE	92.3% (0.018)	0.006 ($<10^{-4}$)	2.568 (0.29)	0.205 (0.07)
	FFR	-	-	-	1.584 (0.11)
	PenFFR	-	-	-	1.213 (0.07)
	pffr	-	-	-	1.266 (0.08)
S2	FFMoE	93.0% (0.005)	0.006 ($<10^{-4}$)	2.500 (0.18)	0.180 (0.02)
	PenFFMoE	93.3% (0.003)	0.006 ($<10^{-4}$)	2.501 (0.18)	0.174 (0.01)
	FFR	-	-	-	1.561 (0.07)
	PenFFR	-	-	-	1.192 (0.04)
	pffr	-	-	-	1.252 (0.05)
S3	FFMoE	92.0% (0.026)	0.016 ($<10^{-3}$)	2.715 (0.49)	0.280 (0.38)
	PenFFMoE	92.8% (0.040)	0.016 ($<10^{-3}$)	2.730 (0.45)	0.219 (0.16)
	FFR	-	-	-	1.612 (0.12)
	PenFFR	-	-	-	1.227 (0.07)
	pffr	-	-	-	1.290 (0.09)
S4	FFMoE	93.9% (0.004)	0.015 ($<10^{-4}$)	2.628 (0.21)	0.174 (0.02)
	PenFFMoE	94.3% (0.003)	0.016 ($<10^{-4}$)	2.614 (0.20)	0.165 (0.01)
	FFR	-	-	-	1.646 (0.08)
	PenFFR	-	-	-	1.237 (0.05)
	pffr	-	-	-	1.314 (0.06)

Table 3.2: Expert affectation accuracy and average (standard deviation) of MRPE on a test sample.

Prediction accuracy

In Table 3.2, we present the predictive accuracy through MRPE of the proposed method (FFMoE and PenFFMoE) and of its competitors (FFR, PenFFR and pffr) on the test sample. The results are clearly better for FFMoE and PenFFMoE. Let's remark that the difference between MRPE.good and MRPE.bad show that it is important to correctly affect the observations to the correct expert.

Finally, Figure 3.7 gives the prediction for four randomly chosen observations compared to actual values. Figure (3.7a) and Figure (3.7b) correspond to situations where the observations are assigned to the correct clusters; Figure (3.7c) corresponds to a case where the data is assigned to the correct clusters by the penalized method but not for the non penalized method and Figure (3.7d) corresponds to cases where the data is assigned to a wrong cluster, for both methods.



(a) Prediction 1 **(b)** Prediction 2

(c) Prediction 3 **(d)** Prediction 4

Figure 3.7: observed data vs Fitted response functions for four chosen individuals on the test sample. Red and green lines match to FFMoE and PenFFMoE resp.; gold and violet lines are the prediction pffr and PenFFR resp.; the actual data isthe blue dots.

3.6 Application to real data

In this section, we perform our proposed methodology FFMoE and PenFFMoE on two real-world data sets: Canadian Weather (CW, available in the R package `fda`) and Cycling (available in the R package `FREG`). In each of these data sets, the prediction accuracy of FFMoE and PenFFMoE is compared with the competitors PenFFR (Tamo Tchomgui et al., 2023b) and pffr (Ivanescu et al., 2015). Let us remark that PenFFR and pffr consider a single model and not a mixture as compared with FFMoE and PenFFMoE. Comparison is done by the leave-one-out cross-validation integrated square error (ISE):

$$\mathrm{ISE}_i = \int_0^T \left(\mathrm{Y}_i(t) - \widehat{\mathrm{Y}}_i^{(-i)}(t)\right)^2 dt\,,$$

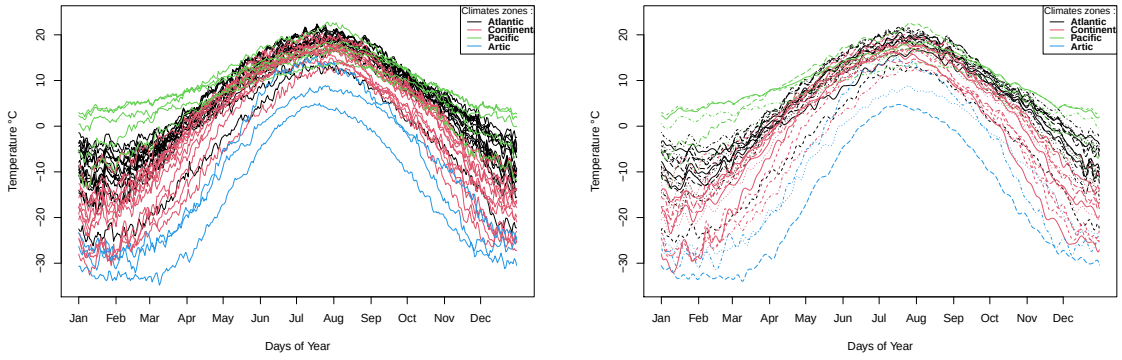
where $\widehat{\mathrm{Y}}_i^{(-i)}(t)$ is the prediction of the i^{th} observation given by the model trained on a dataset of all the observations without the i^{th} one. Computationally, this criterion is approximated by the L^2-norm between the actual and prediction values on a grid of values t is used as a surrogate. It is given by:

$$\widehat{\mathrm{ISE}}_i = \sum_{j=1}^{m} \left(\mathrm{Y}_i(t_j) - \widehat{\mathrm{Y}}_i^{(-i)}(t_j)\right)^2. \tag{3.6.1}$$

3.6.1 Canadian Weather data

The data set consists of $m = 365$ daily temperature measurements (over the year 1961 to 1994) at $n = 35$ weather stations in Canada, and their corresponding daily precipitation (in log scale). Our goal is to predict the (log) daily precipitations functions $\mathrm{Y}_i(t)$ using its corresponding temperatures $\mathrm{X}_i(t)$, for $t \in [0, 365]$.

Figure 3.8 displays the raw temperatures and their cubic B-splines smoothing with $\mathrm{L_X} = 100$ basis functions and equispaced knots. Figure 3.9 shows the raw log precipitations profiles to predict.



(a) **(b)**

Figure 3.8: 35 daily mean raw (a) and processed (b) temperature measurement curves.

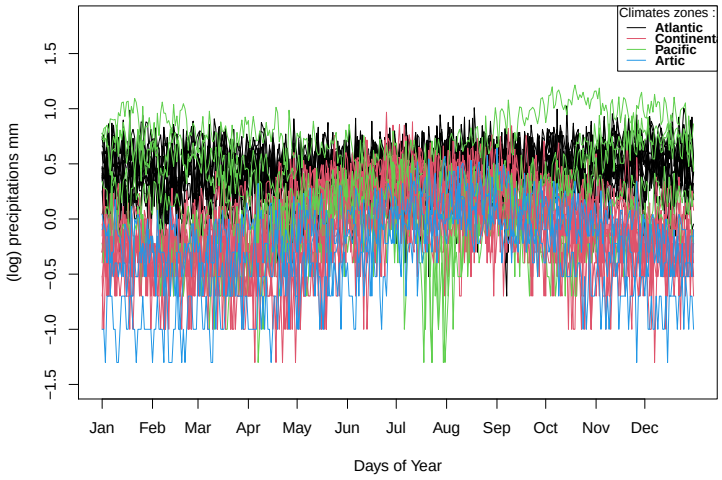


Figure 3.9: raw log precipitations profiles of the 35 Canadian weather stations.

Following the target to obtain smooth estimates of parameter curves (or surfaces) and accurate predictions, we must correctly choose the number of basis functions of functional parameters without forgetting that the number of parameters of the simple model is nearly multiplied by the number of components to get the number of parameters of MoE models. So we set L_β the number of basis functions to 8 both for FFMoE and PenFFMoE. And for the non-mixture models (PenFFR and pffr), we set L_β to 40. The penalty parameters λ_0 and λ_1 for the intercept and temperature effect are selected using cross-validation on a predefined grid of values (3 equispaced values between 0 and 0.5). Model selection is made using the BIC criterion for each LOO model with the number of expert components K in the set $\{1, 2, 3, 4, 5\}$. We observed in Table 3.3 that both for FFMoE and the PenFFMoE, the number of experts component is mostly selected to K = 4. The same situation is observed between the two methods due to the fact that the cross-validation procedure leads to selecting mostly a null value of λ.

Number of components	1	2	3	4	5
FFMoE	0%	0%	20.00%	51.43%	28.51%
PenFFMoE	0%	0%	11.43%	65.71%	22.86%

Table 3.3: Proportion of number of experts per model obtained by BIC selection.

Table 3.4 shows the average value of $\widehat{\text{ISE}}_i$, the standard deviation and the median over the $n = 35$ weather stations. It is important to recall that the statistics are computed over LOO cross-validation, so on different model estimations (including the choice of K). We note a little enhancement in the predictive quality of the mixture models (PenFFMoE, FFMoE) compared to the models without mixture (PenFFR, pffr), and also a smaller inter-individual variance.

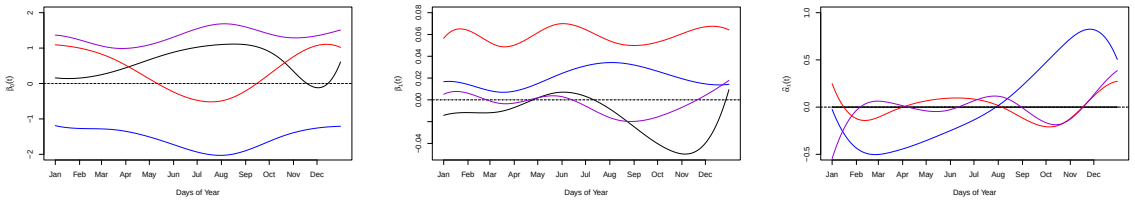
Methods	average $\widehat{\text{ISE}}$	sd $\widehat{\text{ISE}}$	median $\widehat{\text{ISE}}$
PenFFMoE	**29.91**	**21.07**	22.83
FFMoE	30.00	30.37	21.17
PenFFR	36.40	40.42	**21.04**
pffr	89.51	52.06	71.22

Table 3.4: Average, standard deviation and median of $\widehat{\text{ISE}}_i$ for the Canadian Weather data set. The best result is in boldface.

Another advantage of mixture models is the interpretation of the mixture component belonging. For this, new estimations of PenFFMoE and FFMoE are performed on the whole data set. The BIC criterion selects K = 3 components for FFMoE and K = 4 for PenFFMoE.

Figure 3.10 shows the regression coefficient $\hat{\beta}_k(t)$ and gated network parameters $\hat{\alpha}_k(t)$ for the PenFFMoE version. Note that for the gated network parameters, we only have $K-1$ curves due to the identifiability condition, which imposes that $\alpha_1(t) = 0$. We also observed that PenFFMoE parameters are slightly smoother than for FFMoE parameters (see Appendix B.4. This led to a better highlighting of all components of the impact of temperatures on precipitations at different times of the year.

Figure 3.11 plots the temperature curve (covariate) according to the group membership. Similarly, Figure 3.12a plots the geographical positions of the stations. We note a high correlation with the four climate zones of Canada, which is confirmed by the confusion matrices given in Table (3.12b). Finally, Figure 3.13 gives predictions for two randomly chosen weather stations (Churchill and Edmonton) and are compared with the actual precipitation.



(a) Intercepts **(b)** Expert parameters **(c)** Gated parameters

Figure 3.10: Functional coefficients and gated network parameters obtained by PenFFMoE on Canadian Weather data. Color depends on group membership.

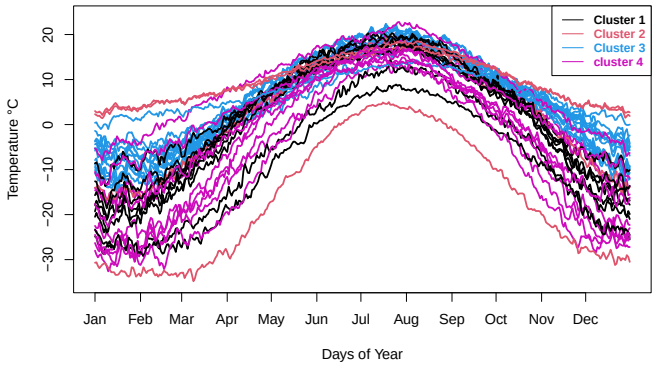
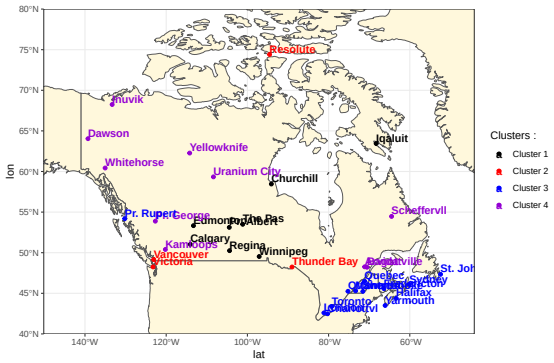


Figure 3.11: Temperature curves with color indicating group membership

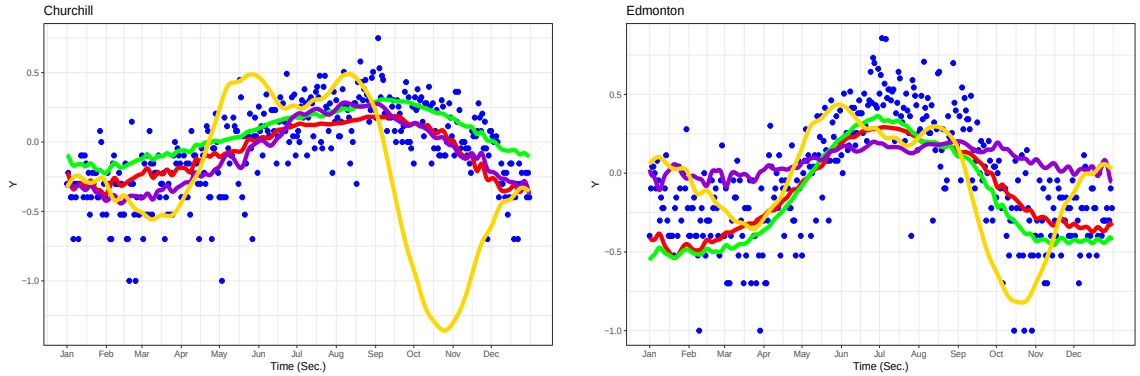


(a)

PenFFMoE	**Clusters**			
	1	2	3	4
Atlantic	0	0	12	3
Continental	7	1	0	4
Pacific	0	2	1	2
Artic	1	1	0	1

(b)

Figure 3.12: Geographic visualization of the 35 weather stations clustering by PenFFMoE and confusion matrix between clusters and climates zones.

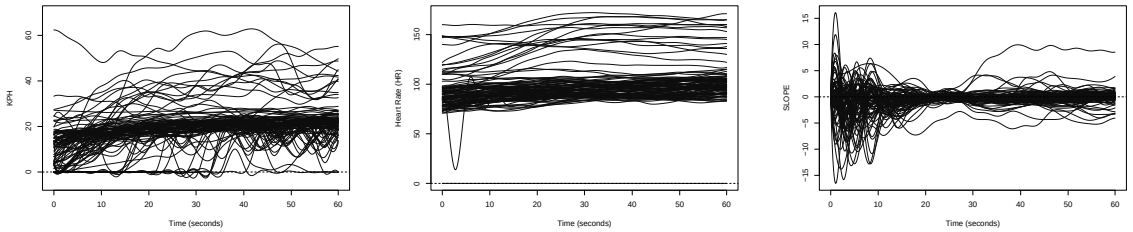


(a) Churchill station **(b)** Edmonton station

Figure 3.13: Prediction for two randomly chosen stations. Blue points are the actual data, red and green lines are the predictions for FFMoE and PenFFMoE resp. The violet and gold lines are the predictions given by PenFFR and pffr resp.

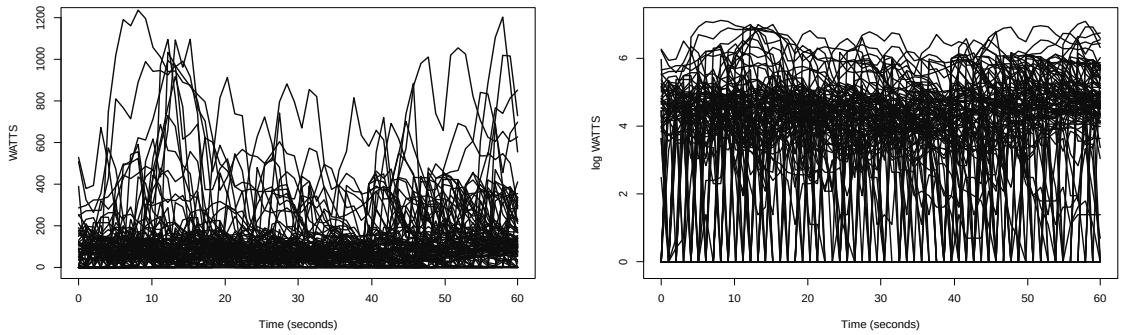
3.6.2 Cycling Data

This data set, initially studied in Jacques and Samardzic (2022), contains the measurements of several parameters during 216 cycling sessions of 30 minutes. The parameters are the power developed by the cyclist (in watts), its heart rate (in beats per minute), the pedalling cadence (in rotation per minute), the speed (in km/h), the slope (in percentage), the outdoor temperature (in Celsius degree) the altitude (in meters). The sampling rate is one measure per second. Our goal in this study is to predict the developed power according to the three parameters known to have an impact (Jacques and Samardzic, 2022): speed (KPH), heart rate (HR) and slope (SLOPE). Due to the high variability of these parameters during a period of 30 minutes, we restrict our analysis to a small portion of the curve, corresponding to the 20^{th}-minute (chosen arbitrarily).



(a) FD KPH **(b)** FD HR **(c)** FD SLOPE

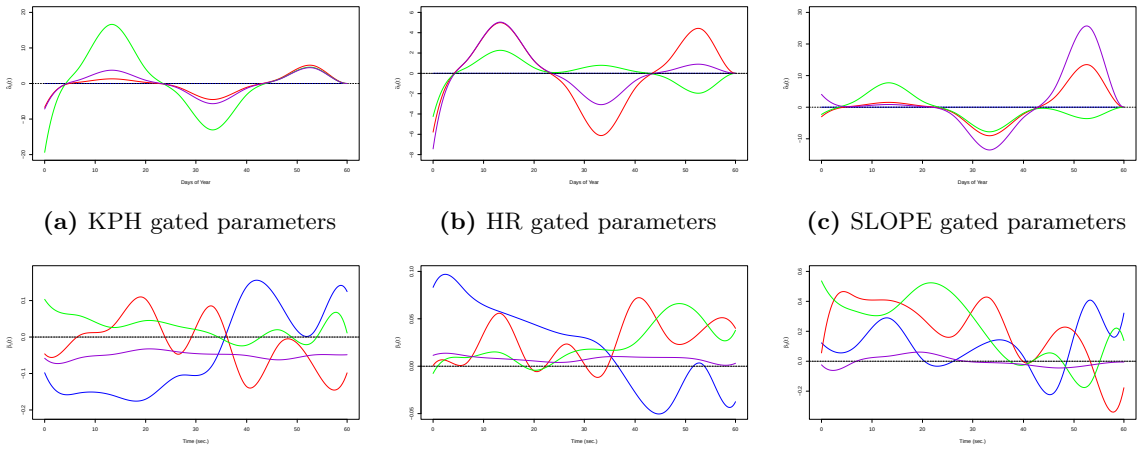
Figure 3.14: Raw and functional expansion curves of speed (a,d), heart rate (b,e) and slope (c,f) for 100 cyclists.



(a) raw WATTS **(b)** logarithmic transformation

Figure 3.15: Power developed by 100 cyclists and the corresponding logarithmic transformation.

Figure 3.14 shows the functional expansion in cubic B-splines as a function of the three covariates. Figure 3.15a plots the developed power. Due to its dispersion, a logarithmic transformation is applied (3.15b).



(a) KPH gated parameters **(b)** HR gated parameters **(c)** SLOPE gated parameters

(d) KPH's expert parameters **(e)** HR's expert parameters **(f)** SLOPE's expert parameters

Figure 3.16: Functional gated (first row) and expert (second row) parameters obtained by FFMoE on Cycling data. corresponding colors matched for the same cluster.

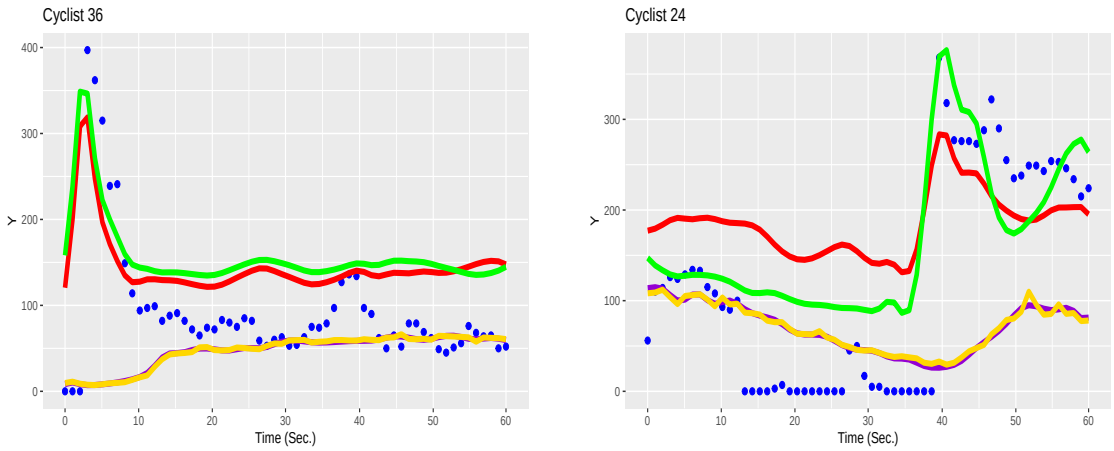
We evaluate on this data set FFMoE, PenFFMoE, PenFFR and pffr. Predictive performances are evaluated through the ISE. The data set is split into train and test subsets with proportions 80% and 20%. The number of components for FFMoE, PenFFMoE is made using BIC with nK in the set $\{1, 2, \ldots, 15\}$. We obtained K = 4 for FFMoE and K = 3 for PenFFMoE, with a better BIC for FFMoE. Figure 3.16 shows the gated and expert parameters for FFMoE, which allow interesting interpretation. For instance, for the fourth cluster (violet), the effect of the three features is almost

constant, which means that the cyclist has a regular effort, with regular speed, heart rate and slope. On the contrary, for the second cluster (in blue), the effect of KPH goes from negative to positive, whereas the effect of HR does the contrary: probably that this session corresponds to an end of a climb: during the climb, the cyclist goes slowly whereas developing a high power and high HR, and then, after the summit of the climb, keeping a high power allows him to go fastly with a decreasing HR. Figure B.3 in Appendix B.5 shows the results for PenFFMoE.

Table 3.5 presents the average and standard deviation of ISE (over the test set) for the different models. If we consider the ISE averaged over the individuals of the test set, the best results are obtained with pffr. But looking at the median ISE, we conclude that most individuals are better predicted with the PenFFMoE method. This is in particular confirmed by Figure 3.17, which plots the predictions on two randomly chosen cycling sessions, on which we can see that the prediction with FFMoE and PenFFMoE better follow the general shape of the curves.

Methods	BIC	Nb clusters	Average $\widehat{\text{ISE}}$	sd $\widehat{\text{ISE}}$	median $\widehat{\text{ISE}}$
PenFFMoE	26729.9	3	160.72	225.64	**34.63**
FFMoE	26275.2	4	155.20	202.94	47.66
PenFFR		/	155.07	181.85	47.31
pffr		/	**154.78**	**181.61**	46.82

Table 3.5: The average and standard deviation of $\widehat{\text{ISE}}$ for Cycling data set. The best result is in boldface.



(a) **(b)**

Figure 3.17: Prediction on two randomly chosen cycling sessions. Blue points are the actual data, and red and green lines are the predictions for FFMoE and PenFFMoE resp. The violet and gold lines are the predictions given by PenFFR and pffr resp.

3.7 Conclusion

Functional data analysis has now reached a high level of maturity and its manifold applications span a wide range of scientific fields. In the present paper we developed a novel estimation scheme for MoE in the framework where both covariates and response are of functional type using the concurrent linear model with Gaussian error. Preliminary investigations based on plain maximum likelihood estimation, and using functional expansions in standard bases, lead us to the observation of a lack of smoothness of the estimators in various experiments with real-world datasets. In order to circumvent these issues, we introduced a ridge-type penalisation on the second derivatives and obtained a more stable estimator, still capable of handling substantial variability of first-order behaviours. Numerical experiments showed that the FFMoE (also PenFFMoE) has satisfactory behaviour in terms of parameter estimation (interpretability) and predictive accuracy on simulated datasets.

We then illustrate this performance on two real-world datasets. On Canadian weather dataset, PenFFMoE and FFMoE cluster the weather stations in $K = 4$ clusters that match the various climate zones. The predictive accuracy shows a definite advantage of mixture of experts over non-mixture based models. On Cycling data, the predictive quality is certainly not as good as non-mixture models, but it gives predictions that detect regime changes more easily.

Extensions of this work are potentially manifold. One possible avenue is to explore the more general exponential family on the functional response side. A second possible direction would be to investigate possible solutions for producing relevant prediction bounds, using for instance conformal prediction Angelopoulos and Bates (2023) which has attracted great interest lately in the machine learning community.

Chapter 4

Functional Linear Model for Predicting the Streaming Video QoE

The work presented in this chapter focuses on the application of the PenFFR approach, detailed in Chapter 2, for predicting the QoE of streaming video services. This work has been published as a mini conference paper (limited to 7 pages) in the proceedings of the 20th International Conference on Network and Service Management (CNSM), Tamo Tchomgui et al. (2024a). The content has been adapted and expanded for inclusion in this PhD thesis. While the fundamental methodologies and results remain consistent with the published paper, additional insights, analysis, and context have been provided to further elaborate on the contribution to the broader research objectives of this dissertation.

Abstract: Offering the best-in-class QoE for the services they deliver is a challenge to MNOs, what requires them to identify the obstacles to overcome. Lord Kelvin is quoted as saying *"If you can not measure it, you can not improve it"*. To properly achieve this, MNOs require efficient device and network performance monitoring able to maintain this QoE at optimum levels. ML techniques can represent a significant breakthrough compared to classic supervision and measurement techniques. Among other telecommunications services, streaming video is an ubiquitous service used by billions of people every day. This chapter use the FDA-based approach PenFFR presented in Chapter 2 to identify the underlying functions between the QoE and the features that characterize the streaming videos thus enabling to predict future values and envisage corrective actions. The experimental results show that the performance of the prediction is on par for one metric with Deep Learning-based methods and even better for another metric. To facilitate the understanding of our method and enable hands-on application, the code is made available on Github Tamo Tchomgui (2023).

Keywords: Functional data analysis, function-on-function regression, Quality of Experience, QoE prediction, Video streaming.

4.1 Introduction

With the rise in popularity and usage of video streaming services and platforms triggered by larger available mobile network bandwidths and new devices with high resolution screens, we have been witnessing a huge increase in data traffic over mobile networks Cisco (2019) and forecasts confirm this trend for the coming years Ericsson (2024). Surveys show that these services now represent a significant share of the time spent online with more than 33% of people above 18 spending at least an hour a day in average on an online video service [1]. In a context of competitive markets, consumers are naturally expecting a high-quality streaming experience. As a consequence, MNOs face the challenge of ensuring the best QoE possible for their users, with a specific focus on video streaming services, to keep their promise on service experience and avoid churn. QoE is a measure of overall customer satisfaction with factors that encompass the whole service. In the case of streaming video, such factors include video playback quality (both objective and subjective), buffering times, sound smoothness of the playback ... Delivering a good QoE requires the possibility to know and master those factors. However MNOs generally do not have access to the factors that directly influence QoE, unless they are also application providers. These are located in service platforms or on users' devices, but MNOs only have access to QoS or network performance metrics or parameters. QoS relies on network parameters such as bandwidth, one-way or round-trip time (also known as latency), jitter, throughput, packet loss rate, ... or application performance metrics. It becomes wishable for MNOs to infer the quality level of end-user perception that they cannot measure directly and the problem could be cataloged to find a relationship between QoE and QoS. MNOs are increasingly turning to ML techniques to model QoE from network data, thanks to the availability of massive amounts of data, as well as the increased maturity of ML tools and models and associated computing facilities. In the specific case of video streaming services,

[1] https://www.statista.com/statistics/611750/millennial-time-spent-with-online-video/

these models can predict QoE by analyzing vast amounts of data generated from network usage patterns, video streaming metrics from the devices and user feedback. The predictive capabilities of ML theoretically enable operators to proactively manage network resources, optimize streaming quality, and preemptively address potential issues before they affect the user experience. But there are challenges to overcome, related either to the high variability over time while streaming of factors such as network throughput, video bit rate and display size, or to different optimization strategies of streaming services (maximizing video quality versus minimizing rebuffering events). In this chapter we apply the PenFFR method presented in Chapter 2. It is a statistical method based on the FDA paradigm that enables to identify the underlying function in the data and allows, here, to compute the QoE from the input data. Section 4.2 provides a state of the art on QoE and methods to compute or predict it as well as on FDA. Then, Section 4.3 describes the dataset used for the evaluation of the method and the transformations performed on the data. Results of these experiments are then detailed, compared to several baselines and analyzed in Section 4.4.

4.2 Background

The integration of ML models for predicting QoE is now a relatively mature field of study (cf. Mittag (2022)). Modelling of QoE was initially based on psycho-physic, by a replication of human perception (vision, hearing) and opinion building. Such models were very explanatory, with building blocks corresponding to well known physical or mental processes. With ML, the models become independent from psycho-physics and can rely therefore on more sources of information. They also benefit of the flexibility, modularity and scalability of ML techniques.

4.2.1 Video quality assessment

Video Quality Assessment (VQA) refers to the process of evaluating the visual quality of video streams as perceived by human viewers. This assessment is crucial in video transmission, streaming, and broadcasting over networks because maintaining acceptable video quality is essential for user satisfaction (QoE). To assess the QoE of real-time video streaming, VQA methods are categorized into three main types: subjective, objective, and hybrid approaches.

Subjective methods are conducted to obtain information on the quality of a service (telephony, multimedia streaming, etc.), as perceived by end-users, using opinion scores. They gather human observers in a laboratory to watch or listen to specific stimuli and evaluate their respective quality according to their point of view and their perception. All individual quality scores produces by testers are averaged in so-called Mean Opinion Scores (MOS). We commonly have three subjective methods: the first one is the Absolute Category Rating (ACR) where the viewers rate the perceived quality on a scale typically on a five point discrete scale (1 for bad to 5 for excellent, see exact labels in Table 4.1). The second one is the Double Stimulus Continuous Quality Scale (DSCQS) where viewers are shown two stimuli: one reference (original) and one distorted. They rate both and the difference between their ratings indicates quality degradation. The third one is the Single Stimulus Continuous Quality Evaluation (SSCQE) where testers continuously assess quality while watching or listening to the content under test in real-time, using a slider to indicate the perceived quality over time.

MOS	Quality	Perception
5	Perfect	Imperceptible
4	Good	Perceptible
3	Fair	Slightly annoying
2	Poor	Annoying
1	Bad	Very annoying

Table 4.1: Corresponding satisfaction level according to MOS values

Subjective Assessments are very expansive in term of human resources, cost and time consumption. They are also not scalable in real time applications. However, such a technique cannot be used as an automatic measurement, and that is why objective methods are needed. They use the network performances in algorithms or mathematical models to approximate the results of subjective quality evaluation, generally on the same MOS scale from 1 to 5. Some of the most known objective models are classified into three categories. In the list below we focus on these models for video and multimedia quality:

- **Full-reference** (FR) methods compare the distorted video with the original, reference video to measure quality loss. The key metrics of the method are Peak Signal to Noise Ratio (PSNR) that measures the ratio between the maximum possible power of a signal and the power of corrupting noise. The Structural Similarity Metric (SSIM) Wang et al. (2004) evaluates image quality by comparing luminance, contrast, and structure between reference and distorted videos. Video Multimethod Assessment Fusion (VMAF) Aaron et al. (2015) developed by Netflix uses ML models to predict subjective video quality based on a reference and distorted video sequence. FR methods provide high accuracy due to the direct comparison with an available reference video but the important property limiting their application in network environment is the fact that they require the entire reference signal at the evaluation point.

- **No-reference** (NR) methods do not require any reference video for quality evaluation. They rely on the analysis of the distorted video to predict the quality. The key metrics of such methods are video blockiness (block artifacts commonly caused by compression) or the amount of blurring in the video, which can occur during transmission or compression. Blind/No-Reference Image Spatial Quality Evaluator (BRISQUE) Mittal et al. (2012) predicts image and video quality by analyzing statistical deviations from natural scene characteristics. NR methods are applicable to user-uploaded video-centric services such as YouTube and Facebook, where the pristine references are unavailable. But they are generally less accurate. Novel approaches are developed to handle this problem. We can mention Ghadiyaram et al. (2017) that solely relies on the 'quality-aware' natural statistical models in the space-time domain.

- **Reduced-reference** (RR) methods are a compromise between FR and NR metrics. They require partial information from the reference video (such as key features or statistics) to assess the quality of the distorted video. Examples of such methods are the RR-SSIM (or the Multi-Scale Structural SiMilarity Wang et al. (2003)), similar to SSIM but requiring only certain statistical features from the reference video, and the Video Quality Model (VQM) Pinson and Wolf (2004) that uses reduced information to predict quality by examining temporal and spatial variations in video content. RR algorithms can be used in network environment as they require less data than FR but still requires some information from the reference video.

By combining subjective and objective approaches, often by enhancing objective metrics with machine learning techniques trained on subjective datasets, we get hybrid methods. These offer better correlation with human perception compared to traditional objective methods. An example is the VMAF metric. Although it is primarily an objective metric, it was trained on large subjective datasets, making it a hybrid approach that bridges the gap between subjective and objective assessments.

Apart from the metrics quoted above, we also found in the literature various methods and models for assessing subjective quality of video streaming. Among them, the ITU-T has developed a series of standards (called "recommendations") providing standardized methodologies to assess how users perceived the quality of video streaming services. Here are some of the key relevant recommendations:

- **ITU-T P.1203 (ITU, 2017):** It is a hybrid model combining perceptual analysis with bitstream analysis. It has three components : **P.1203.1** for video quality model, **P.1203.2** for audio quality model and **P.1203.3** the module for combining the two models.

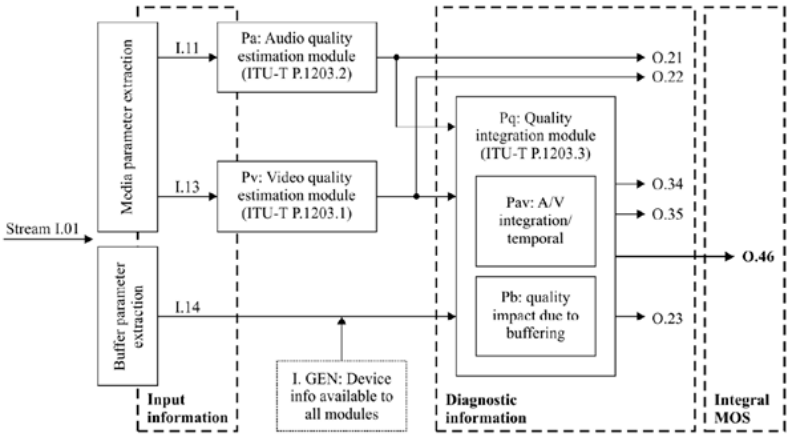


Figure 4.1: ITU-T P.1203 model architecture

- **ITU-T P.910 (ITU, 2008b):** It is a methods for conducting subjective quality assessments of video content, where human viewers directly rate the quality of the video, either using the Absolute Category Rating (ACR) method where the viewers rate the quality of video clips on a scale, the Degradation Category Rating (DCR) method where the viewers rate the quality by comparing a degraded video to a reference video, or the pair comparison method where viewers compare two video clips and choose the one with better quality.

- **ITU-T P.1202 (ITU, 2012):** It is a bitstream based model that doesn't need the original uncompressed video. It has two components : **P.1202.1** for video quality using bitstream model, **P.1202.2** for reduced reference models that use both bitstream and some metadata from the original video.

At the end, VQA methods are employed to provide a measure of the perceived video quality. Depending on the use case, either subjective, objective, or hybrid approaches may be employed, with each having its trade-offs in terms of accuracy, scalability, and practicality.

4.2.2 State of the Art

We can mention the work of Tasaka and Watanabe (2007) which proposes a method for estimating QoE for audio-video transmission over IP in a psychological scale. Most QoE models require access to the audio or video signals, which is most of the time very challenging, and cannot be used for network-centric operations. This is why a few recent studies addressed alternative approaches, where information from the end-users' devices or the network could be envisaged as input for QoE prediction. In the field of voice quality, a good example of such models is the ITU-T Recommendation P.565.1 (ITU, 2008a) "Machine learning model for the assessment of transmission network impact on speech quality for mobile packet-switched voice services" also know as sQLEAR. It is using a combination of regression algorithms, namely a Random Forest (RF) regressor and support vector (SV) regressor. As far as we are aware, there exists no similar model for video quality of streaming services. The development of such models relies on training with a large amount of data representative of network equipment behaviour and performance. The collection of such data is a huge task. Some solutions are proposed based on network simulators (cf. Schwarzmann et al. (2022)) or automated monitoring solutions (cf. Dobreff et al. (2023)). We can also mention the work of Zhang et al. (2020) which developed the `DeepQoE` method, a novel framework using Deep Learning (DL) to predict video QoE. The end-to-end framework first uses a combination of DL techniques such as word embeddings to extract generalized features. Next, these features are combined and fed into a neural network for representation learning. The work of Robitza et al. (2018) carried a study for the standardized ITU-T Rec. P1203 procedure for audiovisual HTTP Adaptive Streaming (HAS). Recall that the ITU-T Rec. P1203 is a standard that provides a model for estimating QoE for video streaming services. The model predicts the user's QoE for sequences of up to five minutes length. Our dataset comprises four subjective databases from the P.1203 competition, including MOS and individual ratings, as well as a thorough evaluation of the model performance in its different application Modes, and a Python implementation of the standard that is free to use for research purposes. Another work from Abar et al. (2017) presents a method for predicting QoE in Software Defined Networks (SDN) using full reference parametric metrics (SSIM, VQM) addressing the limitations of subjective methods like Degradation Category Rating (DCR) methods. They use four algorithms such as Decision Tree (DT), .Neural Network, K-Nearest Neighbors (KNN), and Random Forest (RF) due to their proven efficiency. There are several challenges faced by designers of such network-based QoE prediction models. The biggest one lies in the heterogeneity of data source in terms of data nature (radio, IP, etc.), source location (access, core, device) and time series structure (real time, average overt time, etc.).

With the emergence of new generations of networks, we can collect information at very high frequencies in various places. It has become necessary to develop new tools for exploiting and analyzing this ever increasing volume of data. This is one of reason why FDA has become very popular and useful in a constantly growing number of applications : medical Ullah and Finch (2013), economics Das et al. (2019) and commerce Jank and Shmueli (2006), ... Unlike traditional data analysis, which focuses on discrete observations, FDA involves analyzing data that is inherently continuous, such as curves, surfaces, and shapes. Extension of linear regression

to the functional setting has therefore naturally become a major area of research in FDA. While the literature is too vast to cover here, the recommended references for this field are Horváth and Kokoszka (2012); Kokoszka and Reimherr (2017); Ramsay and Silverman (2005); Ramsay et al. (2009), which provide excellent introduction to FDA. A broad overview of Functional Linear Regression (FLR) methods is provide in Goldsmith et al. (2011) and Morris (2014). In mobile network context, Ben Slimen et al. (2017) use FDA to detect future malfunctions, capacity degradation, accessibility and call drops anomalies for Long-Term Evolution (LTE) networks. As it shown in Muelas et al. (2015), FDA can be use not only to handle the diversity both in terms of the situations that must be faced and the data used to reach conclusions, but also optimized and improve the scalability of solutions in network management task (*e.g.* anomaly detection).

In contrast to simpler methods that reduce the observations to scalar summary values, FDA retains all important information by directly using the functional observations in the analysis. As we previously mentioned, data sets collected over time occur in different fields and the way we analyse it differ from the paradigm we use. Thus we can have FDA through functional regression models and time-series through AutoRegressive (AR) models for time collected data. Compared to the AR time series approach, FDA offers several advantages Gertheiss et al. (2023). While it is able to handle complex and high dimensional data, FDA also preserves the temporal correlation and inherent dynamics in the data while AR can only model dependence on past values that may limit their ability to anticipate rapid changes or complex future patterns. FDA also doesn't need regular spaced data and is able to handle missing data by modelling data as smooth functions. There is also an overlap between the two areas, ideas and methods. FDA have more recently been used to make AR models less parametric and more flexible.

4.3 Dataset and Data analysis

This section first introduces the dataset used for our experiments, then provides some insights on the data it contains. This dataset is then used in the experiments described in Section 4.4. But before that, it's important to say a few words about the Youtube database we've been working on.

Youtube dataset :
During our study, we have focused on the "YouTube goes 5G" dataset (Ul Mustafa et al., 2022) collected on commercial 4G and 5G networks. They authors recorded Channel Level Metrics (CLM) such as: Channel Quality Indicator (CQI), Received power in the whole band - 4G only (LTERSSI), current downlink and uplink, Received Signal Received Quality (RSRQ), etc. And YouTube QoE logs such as events, Current quality, Video Bytes Downloaded, etc. with 1-second granularity.
We initially considered rebuffering events and display quality changes. We implemented an adaptation of ITU-T P.1203 model (cf. Figure 4.1) partly based on the estimation of the impact of these events to build the target QoE scores for our model. But we failed at training and finding a performing model, mostly because the dataset was too small and didn't have enough variation in quality conditions. We could not find, apart from the following NetFlix dataset, another dataset of streaming videos with associated timestamped information on session events. This is why we headed

unfortunately for a much less ambitious target (but still an interesting application): modeling QoE global scores from QoE KPIs.

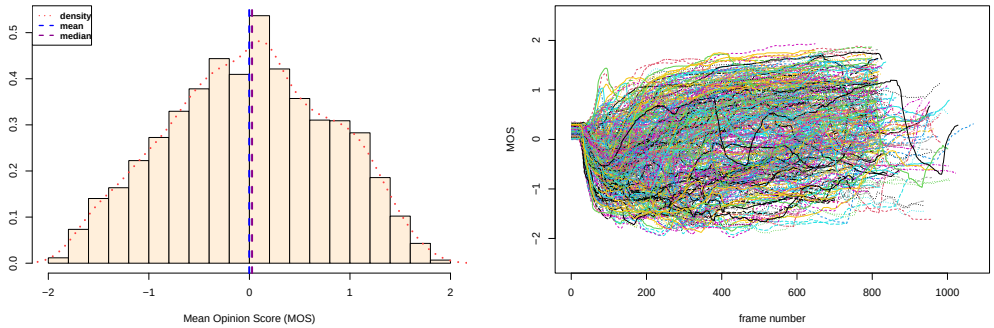
4.3.1 The LIVE-NFLX-II dataset

One challenge when using novel methods is to find appropriate data. There are a few datasets for QoE prediction Li (2020); Mittag et al. (2023) but almost all of them are built to predict a single scalar value, the MOS score of the whole video. For example a dataset for the QoE prediction of video streaming will contains only a single scalar value for the QoE of each video, or each portion of video. To the best of our knowledge the only dataset that contains values for QoE at every timestep or frame of a video is the LIVE-NFLX-II Bampis et al. (2018b). LIVE-NFLX-II is a QoE dataset built by The University of Texas at Austin's LIVE subjective testing lab using realistic adaptive streaming pipeline model that contains four main modules: an encoding module, a video quality module, a network transmission module and a client-based video playout module. This streaming pipeline is designed to recreate a complete end-user QoE through 3 streaming dimensions: encoding, network throughput and the selected Adaptive Bitrate Streaming (ABR) algorithm. To have a look at every of those dimensions, LIVE-NFLX-II dataset is based on 15 video contents, 7 actual network traces and 4 adaptive algorithms, yielding 420 video streams in total. Each of the 15 videos belongs to a content genre: action, documentary, sports, animation and video games. Each of these genres has different dynamics of the content of the video as well as different types of content of each frame. Concerning the network throughput, the 7 network traces were manually selected from the HSDPA dataset Riiser et al. (2013) which is widely used to compare adaptation algorithms, each of these traces allowing to simulate specific network conditions. For the ABR algorithm, 4 representative adaptive algorithms are selected to cover the large design space of adaptation algorithms. We have the buffer-based (BB) that prioritize buffer occupancy to decrease rebuffering events Huang et al. (2014), rated-based (RB) that measure the download speed of previous video chunks to determine which bitrate to choose next Lin et al. (2021), Quality-based (QB) that selects video rates for future video segments based on segment duration and the client's buffer size Rahman et al. (2021) and a version of QB which uses the actual network traces, instead of throughput estimates, the Oracle QB (OQB) approachs. Each sample in the dataset is a stream of one of the 15 videos, encoded by one of the 4 adaptive algorithms and streamed over one of the 7 network conditions. The dataset also provides a subjective continuous-time evaluation of each frame of each of these samples by a subset of at least 22 human subjects among a group of 65. In total there are 9750 QoE evaluation available in the dataset. We refer readers to Bampis et al. (2018b) for more detailed description of the dataset and the protocol followed for the evaluation by human subjects.

4.3.2 Data analysis and data engineering

The collected data of LIVE-NFLX-II consists of 420 distorted videos, each of them viewed by at least 22 subjects. Each of the 65 subjects made 150 evaluation. Overall we have thus $65 \times 150 = 9750$ continuous z-normalized scores to study the subjective QoE. Figure 4.2a shows the distribution of the MOS scores of the overall 420 distorted videos which has a non surprisingly normal form (after a z-normalized transformation). Figure 4.2b shows for each frame (along the x-axis, from the first frame of the video to the last) the perceived MOS on the y-axis and there is one line for each of

the 420 samples, the figure does not show a specific trend.



(a) Mean Opinion Score (MOS) distribution **(b)** Mean Opinion Score (MOS) curves

Figure 4.2: Graphical distribution of the MOS metric

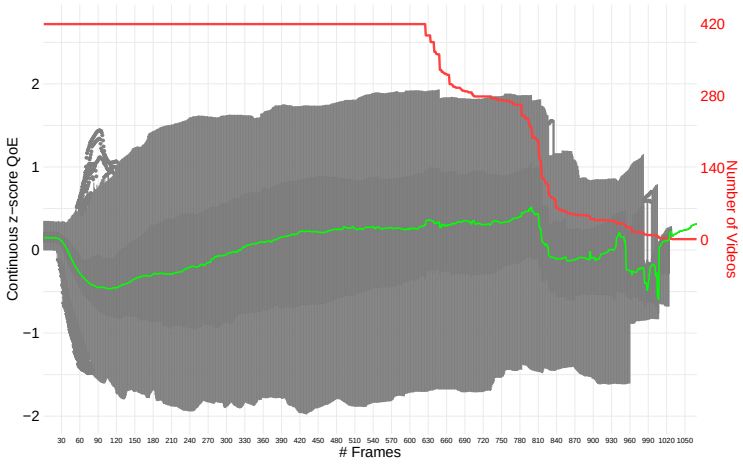
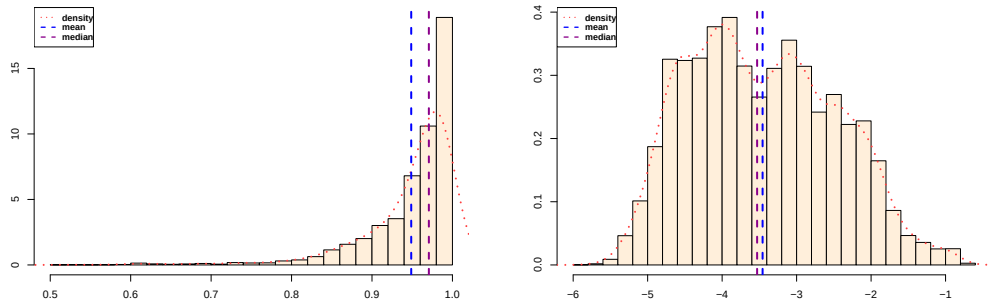


Figure 4.3: Boxplots of MOS for each frame, red line is the total number of videos

Figure 4.3 shows for each frame (along the x-axis) the distribution of the value of the QoE in a boxplot. The darker section in the middle represent the 50% around the median value. The green line show the median values of QoE for each frame (the y-axis on the left side of the figure). The red line shows the total number of videos that have that many frames (the y-axis on the right side of the figure). Indeed, all videos are composed of at least 625 frames, but not all of them have the same number of frames. This figure, like Figure 4.2b, does not show a general trend except for the very first frames where the perceived quality for all videos is within a small interval. For further frames the distribution of the perceived QoE is similar.

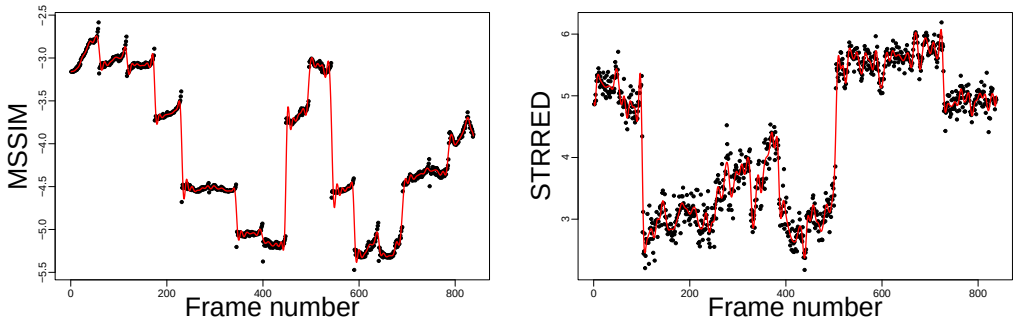
The rest of this section details the features used and transformations applied to them. To

understand the behavior of end-user quality, key metrics are collected on top of playout bitrate, number of rebuffersn and rebuffering time across adaptive algorithms. Others several well-known QoS metrics are used for our model for QoE prediction. The main metrics in the dataset are the Peak Signal to Noise Ratio (PSNR), the Spatio Temporal Reduced Reference Entropic Differencing scores (ST-RRED), the Structural SIMilarity index measure score (SSIM), the MultiScale SSIM (MS-SSIM), the Video Multi-Method Assessment Fusion (VMAF) and the throughput traces. Since we are trying to fit a linear Gaussian model whose variables are assumed to follow this distribution we applied $x \mapsto \log(x)$ and $x \mapsto \log(1-x)$ transformations respectively for metrics whose distribution was either skewed to the left or to the right, to obtain a more or less normal distribution. Fig. 4.4 shows an example, for the MS-SSIM metric, of the histograms before and after this transformation.



(a) Original data **(b)** log Transformation

Figure 4.4: MS-SSIM scores and its logarithm transformation



(a) MS-SSIM **(b)** STRRED

Figure 4.5: Raw data (black dots) and Functional expansion (red curve) of a randomly choose video for MS-SSIM and STRRED covariates

Functional basis expansion of the covariates using $L_\beta = 150$ B-splines basis functions give

satisfactory results. An illustration is given in Figure 4.5: Figure 4.5a for MS-SSIM and Figure 4.5b for ST-RRED scores. We note that unless we have more than thousand raw observations, B-splines basis expansion is able to capture the trend of the dynamic evolution.

4.4 Experiments and results

This section first details the experimental protocol followed to apply the model on the dataset described in Section 4.3. It then gives the results and compare them to two types of baseline, one from the original work that introduced the dataset and a more basic baseline based on a RF model.

4.4.1 Baseline

We compare the results of our PenFFR model, described in Chapter 2, to the two prediction algorithms presented in Bampis et al. (2018a). The first of these models is based on AR Neural Networks (NNs) (G-NARX) and the other on recurrent NNs (G-RNN). These 2 models were trained using VMAF measurements per frame as the continuous-time Video Quality Assessment (VQA) feature. They also included two other continuous-time features: a per-frame boolean variable indicating the presence of rebuffering and another indicating the time elapsed since the last rebuffering event. G-NARX used 8 input delays and 8 feedback delays. G-RNN used 5 layer delays. Both approaches used 8 hidden nodes and the training process was repeated three times, resulting in a set of three test predictions per distorted video, which were averaged to obtain more reliable time series predictions. Those results from Bampis et al. (2018b) are used here as a baseline to compare with our methods. The experience was not reproduced, we compare our results to the result reported in this article.

We also compare our FDA-based PenFFR method and these two algorithms to a RF model. To train the RF model, the per-frame metrics described in Section 4.3.2 are used. For each of them, statistical features are computed : *maximum, minimum, total, quartiles, standard deviation, mean, skewness* and *kurtosis*. The output of the RF model is the mean value of continuous QoE z-scores of distorted videos. We also train another RF model with output variable as standard deviation of continuous z-scores in order to build the confident interval of these predictions. When all the features are computed, a grid search is performed for the hyper-parameters of the RF model: the maximum depth of trees, and the number of trees. Once the best values for the hyper-parameters are found, an initial RF model is trained using all the features. The features of the trained RF model are ranked based on their importance expressed by their Mean Decrease in Impurity (MDI) Louppe et al. (2013). Then an additional training step is performed, with the features added one by one in order of importance until a high accuracy beyond a pre-defined threshold is reached. This enables the selection of the smallest number of features that is required to achieve high accuracy. Once the final set of features and hyper-parameters are chosen, the final model is trained.

Figure 4.6 shows the pipeline of the 4 different models: PenFFR, G-NARX and G-RNN and RF. All models rely of course on the same input data. Additional statitical feature are added in the case of the RF model as detailed above. It is important to understand that each of these methods has a different format of output. The RF model in the baseline outputs a single scalar value for each of the n videos (the mean MOS). Both G-NARX and G-RNN output n vectors (one for each video) of values, i.e. the MOS for each frame. Finally the proposed PenFFR method computes a function for each of the videos that allow to compute the MOS for any frame f.

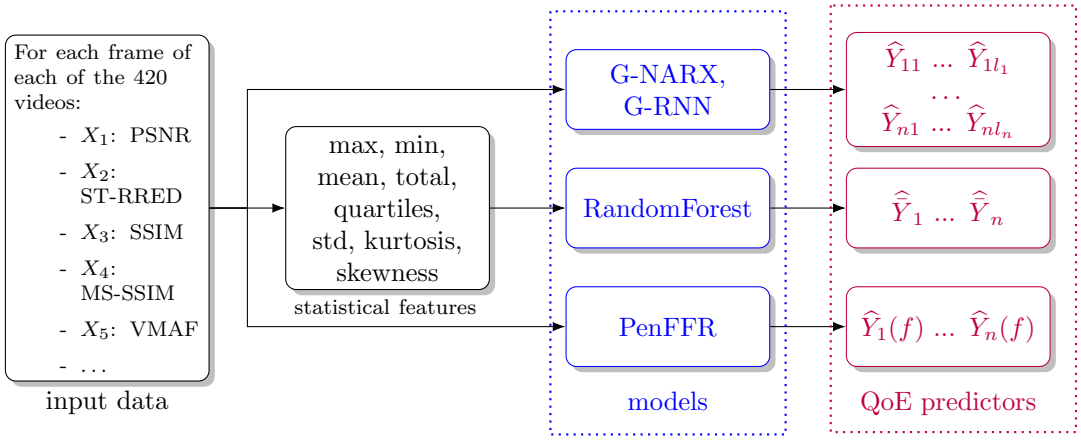


Figure 4.6: Pipelines of the models

For all the 4 algorithms (PenFFR, G-NARX, G-RNN and RF), the train/test split process is performed based on which of the 7 network conditions was used : choosing 5 types of network conditions for training and 2 for testing each time, which yields 300 videos (15 contents, 4 adaptors and 5 traces) for training and 120 videos (15 contents, 4 adaptors and 2 traces) for testing. After describing implementations details of the methods, selecting appropriate evaluation metrics predictive outputs to ground truth is crucial. In traditional VQA Bampis and Bovik (2017b) and models of retrospective QoE Duanmu et al. (2016), Seshadrinathan and Bovik (2010), Spearman Rank Order Correlation Coefficient (SROCC), measuring monotonicity, and Pearson's Linear Correlation Coefficient (PLCC), measuring linear accuracy, are commonly used. These metrics, also used in continuous-time QoE prediction Ghadiyaram et al. (2015), Chen et al. (2014), prompt the question of a singular evaluation metric for continuous-time QoE scores. Considering each metrics merits collectively, we discuss their advantages and limitations for comparing continuous ground truth and predicted QoE. However, continuous-time subjective QoE's dynamic nature requires different evaluation metrics, as SROCC and PLCC assume independence of measurements, which doesn't hold true for subjective QoE with his time dependencies and non-stationarities. Other metrics are suitable for our purpose Bampis and Bovik (2017a). We decide to use the following ones:

1. The **Root Mean Square Error** (RMSE) is a metric commonly used for the evaluation of the accuracy of predictions. It provide a quantitative measure of how well predictions p_i align with ground truth values g_i. His formula is:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(p_i - g_i)^2}$$

   where N is the number of frames. RMSE is intuitive and easy to interpret but sensitive to outliers.

2. The **Outage Rate** (OR) assess the quality of prediction by quantifying the proportion of instances (time) where the predicted value falls outside a predefined tolerance range or thresh-

old. Here we consider the 95% confidence interval of the ground truth g_i at frame i across all videos. OR is calculated as :

$$\text{OR} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}_{\{p_i - \frac{\varepsilon}{2} \leq g_i \leq p_i + \frac{\varepsilon}{2}\}}$$

where ε is the length of the confident interval.
OR should be interpreted in conjunction with other performance metrics to provide a comprehensive assessment of prediction quality. The choice of tolerance threshold should be carefully selected based on domain-specific requirements and the desired level of prediction accuracy.

Using these 2 metrics allows us to compare the performance of our model and of the RF baseline to the G-NARX and G-RNN models since these metrics were used in Bampis et al. (2018b). The results are given and discussed below.

4.4.2 Results and discussion

This section introduces the results of the 4 different algorithms described above. For G-NARX and G-RNN the results are extracted from Bampis et al. (2018b). For RF and PenFFR the results come from the experiments described in Section 4.3. The results are summed up in Table 4.2.

The RF baseline has the worst RMSE value, 150 % of the G-NARX algorithm, which gives the best RMSE. The functional PenFFR method has a RMSE of 0.289, 108 % of G-NARX.

For the OR metric, PenFFR offers the best result with 4.72 % of predicted values of all frames out of the 95 % confidence interval, when G-NARX reaches 7.14 % and G-RNN 5.96 %. RF reaches 5.6 %.

The proposed PenFFR method allows for at least a 25 % improvement of the OR metric over the G-NARX and G-RNN methods while getting a 8 % worse RMSE value. This method does not only predict the values of the QoE metric but also outputs a function that describes how the QoE metric can be computed from the input features.

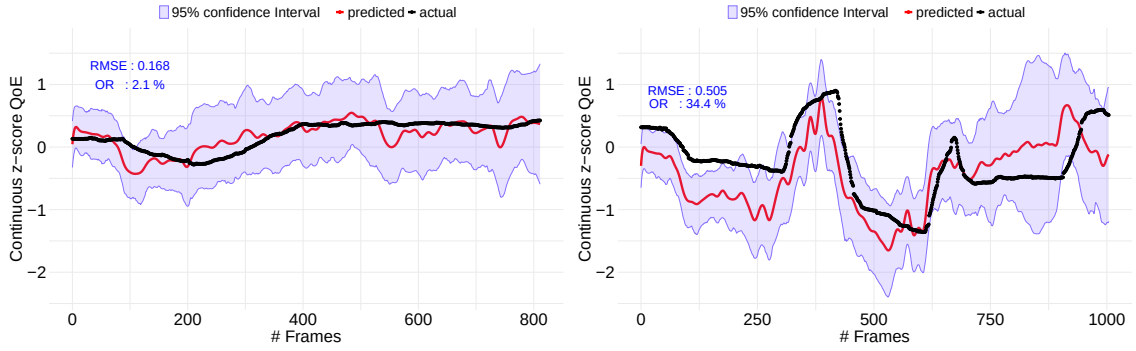
Methods	RMSE	OR
G-NARX	**0.267**	7.14%
G-RNN	0.276	5.96%
RF	0.402	5.6%
PenFFR	0.289	**4.72%**

Table 4.2: Results of RMSE and OR metrics for the 4 methods

Figure 4.7 shows the prediction (red line) compared to the actual values (black dots) for two examples taken from the test set. The example in Figure 4.7a reaches better values for both the RMSE and OR metrics, while the second example in Figure 4.7b reaches worst values. The purple area shows the 95% confidence interval.

4.5 Conclusion and future works

This chapter details PenFFR, a new model based on the FDA framework, and how to apply it for the prediction of the QoE of a mobile telecommuniaction service, in this case video streaming. This



(a) **(b)**

Figure 4.7: Examples of prediction vs ground truth where metrics are better (4.7a) or worse (4.7b) than average

model allows to identify the underlying function between an output variable (here the QoE) and the input variable (here the characteristics of the video) when both input and output variables are functions. In the dataset used for the experiment, both the input and output variables are functions of the frame within a video stream. The experimental results show that this model can outperform current state of the art Deep Learning models for the Outage Rate metric with a slight degradation of the RMSE.

This initial application of FDA in a Network Management use case could be further pursued on other datasets with comprehensive evaluations and on the development of advanced methodologies that leverage the inherent strengths of this expanding framework. Possible other FDA-based models include another of our works Tamo Tchomgui et al. (2024c) that presents, in a theoretical aspect, the functional MoE dedicated to handle heterogeneous data. We are confident that by applying this MoE model in the task of QoE prediction for streaming video, we will be able, in a single model, to model data coming from multiple sources of streaming applications. Ultimately, this work could also help steering the standard on QoE evaluation towards new continuous metrics.

Chapter 5

Function-on-Function Mixture-of-Experts Model for Predicting the Voice over IP QoE

In this chapter, we focus on predicting the QoE of VoIP service using our FFMoE approach described in Chapter 3. This research includes significant contributions from the work done by Roberto Petoh Tsene during his MSc : "DataScale : Gestion des données et extraction de connaissances à large échelle" internship from March to September 2024, which I supervised. Its substantial contributions on the data preparation and on the application of the PenFFR approach are used as baseline to compare the performances of FFMoE. The methodologies and results discussed herein build upon the collaborative efforts developed during this internship.
At the time of writing, this work has not yet been published but is part of ongoing research that is expected to lead to future publications. While the content has been adapted and expanded for inclusion in this PhD thesis, it closely aligns with the collaborative research conducted during the internship.

Abstract: With the advent of the fifth generation of mobile networks (5G), we are witnessing a transformation in the way networks are designed and behave, with the aim of improving the end-user experience. A necessary condition to achieve this goal is the shift from the QoS to the QoE paradigm. Factors with an influence on QoE for telephony applications include audio quality, connection stability and delay. For various services and particularly VoIP, the impact of network performance on user-centric QoE translated into a MOS score is not well understood while a large literature reports about studies on this topic and this work proposes a Machine Learning (ML) approach. Traditional models often fail to detect the complex and time-varying relationships between network performance metrics and user experience, particularly under diverse and dynamic network conditions. That's why we apply the novel FFMoE linear regression model tailored to predict the MOS score. The proposed FFMoE model addresses these limitations by leveraging the principles of Functional Data Analysis (FDA) and Mixture-of-Experts (MoE). Functional data analysis allows us to treat both the response variable (QoE) and the covariates (network performance metrics) as continuous functions over time, capturing their inherent temporal dependencies. The mixture of experts framework enhances this by dividing the input space into regions, each governed by an expert model specialized in handling specific network conditions. The obtained results demonstrate significant improvements in explainability and prediction accuracy compared to non mixture regression models. Key performance metrics such as Mean Squared Error (MSE), Mean Relative Prediction Error (MRPE), and Coverage Proportion (CovP) indicate the model's superior ability to handle the complex and variable nature of VoIP service quality.

Keywords: Voice over IP (VoIP), Mean Opinion Score (MOS), Quality of Experience (QoE), Functional data, Functional regression models.

5.1 Introduction

The constant evolution of telecommunications systems has led to a transformation of the way we communicate, access information, and interact with the world around us. Mobile networks are used simultaneously by a large number of users, who may or may not be moving. At the same time, IP packets data transfer has become the norm, enabling better allocation of communication resources. As well as the voice transmission, these technologies make it possible to increase data rates and reduce download times for all kinds of services, or to view videos instantly. The necessity for MNO to assess the voice quality of these new telecommunications systems arises from the fact that, unlike fixed-line telephony where degradations are well known and relatively well controlled, and so quality of service (QoS) is virtually guaranteed, in new technologies like mobile networks and IP transfer, degradation have not only been extended, but new ones have also been added, such as packet loss and distortions due to reduced voice rate coding, which no longer guarantee this quality and therefore user satisfaction. That's why we need a measure that reflects user perception, i.e. belonging to QoE. This concept refers to all the characteristics that satisfy, retain or give confidence to a user throughout the lifecycle of a service. It differs from QoS in that the latter is based on objective characteristics of the components that make up the service, and does not always allow QoE to be deduced. Measuring the QoE of many services requires information at the application level (audio band such as NB, WB, FB or the presence of denoising devices to correct packet loss,

broadcast quality, interruption time in the case of video streaming, for example). This information is generally not available to MNOs, unless they are also content providers or have agreements with these providers (WhatsApp, Viber, Signal, Telegram ...) to access this information. In the absence of such agreements, QoE measurement becomes difficult. However, the advent of 5G (through one of the components of his architecture, Network Data Analytics Function (NWDAF)), we can gather data from different network entities, process it using advanced analytics techniques, such as ML and statistical analysis and generate reports and notifications for MNOs, enabling them to make informed decisions. Thus, an MNO can apply ML techniques to train and deploy models to estimate QoE based solely on the network statistics to which it has access.

QoE estimation by ML is an issue that has been studied extensively and presents a variety of challenges, both in its design and implementation. While from a theoretical point of view, it can be summarized as finding the most accurate ML approach, it has other requirements on the MNO side, notably in terms of execution speed and resource consumption, which will have an impact on the triggering of corrective actions on the network in real time, for example. In the specific case of VoIP that interests us in this chapter, we need to perform QoE's estimation based on data from phone conversations of a certain duration (in our study, we considered as long as 50 seconds), throughout which we have collected various network parameters and calculated a score giving information on the level of QoE. The methodological difficulty lies in the fact that these data are collected at various times and periodicity and can be specific to each conversation. The best framework analysis that takes into account not only this type of data but also the temporal dimension is the FDA framework. To address the variability in network conditions and user experiences, the MoE approach is integrated into the functional regression framework. The FFMoE model divides the input space into regions using a gating network function, which assigns weights to different expert models. Each expert is a function-on-function linear regression model tailored to specific conditions of the network metrics where different expert models are trained to predict QoE. This allows the model to handle heterogeneous data more effectively by tailoring predictions to specific conditions.

In the flow of this work, Section 5.2 is devoted to the voice quality assessment and related work in QoE prediction for VoIP. Section 5.3 describes the data at our disposal and the various treatments we have carried out on these data. Section 5.4 gives a brief recall of the functional MoE model that we use. Section 5.5 is devoted to experimental results, demonstrating the improvement of the MoE models compare to the non mixture one.

5.2 Background and Motivation

VoIP and more generally the lates generations of mobile telecommunications have revolutionized the way we communicate, offering a versatile and cost-effective alternative to traditional, circuit-switched fixed-line telephony. However, maintaining a high QoE for VoIP users is a persistent challenge due to the complex and dynamic nature of internet-based communication. Indeed, IP packets are sent in the network independently of each other, which doesn't guarantee that they arrive at the receiver in the right pattern. Routers ensure that each IP packet is routed through the network using the shortest route or the most secure one. However, packets can sometimes be delayed or lost. This is reflected at the receiver side by a variable delay depending on the delay of each packet, called jitter, or by lost packets, resulting in cuts in the received speech signal. Jitter, packet loss and delay are therefore the three major degradations induced by VoIP which forces MNOs to reliably assess the voice quality of their telecommunications systems before being able to repair them and provide a high quality grade service to their customers.

5.2.1 Voice quality assessment

The quality of a voice message in general is a phenomenon that depends subjectively on the evaluating person, but also on a multitude of factors. we can mention Guéguin (2006):

- **The independent factors** such as past experience, expectations, mood, ... of the person receiving the message;
- **The context:** listening only, speaking only or conversation
- **The voice message content:** information, emotions, ...;
- **The environment:** location, surrounding noice, ...;
- **The key quality criteria:** message comprehension, voice recognition, listening efforts;

This quality can also be affected by factors linked to communication techniques and their deterioration. We can also mention the voice processing terminals and equipment at both sending and receiving sides (coding, echo cancellation, noise suppression), the network transmission (core, access) and the audio signal processing bandwidth (Narrow Band, Wide Band, Full Band). All these criteria have more or less significant influence on vocal quality. So it's a complex and multidimensional phenomenon that requires a special process to be measured.

As mentioned above, the assessment of voice quality is above all subjective. So the best method to evaluate it is to rely on users and interview them, in the form of surveys or laboratory tests. These methods are called subjective quality assessment methods. To guarantee reliability and reproductibility, subjective evaluation methods are based on procedures described in standards framed by the ITU-T. as far as voice quality is concerned, the corresponding standrad is the Recommendation P.800 standard (ITU, 1996). Compliance with these standards concerns the sound reproduction equipment, the listening environment, the balance between the different quality levels evaluated, or their order of passage. In this context, different speech samples, typically 8 seconds long (test) and representing the different conditions to be tested (generally 50 different conditions can be evaluated in a single test), are presented in random order to several participants who rate their quality on the MOS scale, ranging from 1 to 5 (cf. Table 4.1). The scores obtained per test condition are then averaged to obtain the final score. However, this is a restrictive and expensive protocol. That's why we need objective methods as an alternative.
Objective methods aim to produce MOS scores that are as faithful as possible to the results of a subjective test, using physical measurements of the transmission system or analysis of received signals. Voice quality is estimated using a "model" which combines various parameters (delay, echo, packet loss rate, etc.):

- **Signal-based models:**
  Since the judgment of the quality of a speech (or music) signal is made after listening to it, it seems natural to attempt to reproduce the processes of hearing and cognition through so-called psychoacoustic models. Several such models have been developed over the past twenty years. The best-known, PESQ (withdrawn ITU-T standard P.862 in ITU (2001)) and POLQA (ITU-T standard P.863 in ITU (2011)), work by comparing the signal to be analyzed with

the associated reference signal, which requires this reference to be transmitted at the end of the chain (this is known as an intrusive method).

- **Parameter-based models:**
  Signal-based models have one major drawback: they require a great deal of dedicated computing power. In many cases, such as telecom network supervision, this limits the number of measurement points and therefore the granularity of their representativeness. To compensate for this shortcoming, we use parametric methods, which correlate less well with human judgment, but have a much lighter footprint. Based on the measurement of technical factors in networks and terminals, these models are able to determine the combination of their impact on voice quality. In telephony, the best-known of these models is called the E-model. Initially developed by the ETSI (European Telecommunications Standard Institute), it was later standardized and perfected by the ITU through the G.107 standard family (ITU, 2015).

For his ability to provide objective, comprehensive, predictive and cost-effectiveness assessments of voice quality, the E-model in its most up-to-date version (standard G.107.2, applying to all audio bandwidths up to full band -20 to 20 000 Hz- telephony), has been used to get the target variable data of this study. The result of the model calculation is a R-Factor that ranges from 0 to 148 where the higher values the better quality and given by the formula:

$$\mathrm{R} = \mathrm{R}_{0,fb} - \mathrm{I}_{s,fb} - \mathrm{I}_{d,fb} - \mathrm{I}_{e,eff,fb} + A \tag{5.2.1}$$

where R_0, fb represents the signal-to-noise ratio including noise sources such as circuit noise and room noise; The I_s, fb factor is a combination of all the degradations that occur more or less simultaneously with the voice signal; The I_d, fb factor represents the degradation caused by delay, echo and the degradation due to the actual equipment; $\mathrm{I}_{e,eff}, fb$ degradation caused by low bit-rate codecs and also includes degradation due to random and burst packet losses; The advantage factor A can be used for compensation when there are other advantages of access to the user (for example, mobility or connections in hard-to-reach areas). After this calculation, the R-Factor can be mapped to MOS scores (on a full band scale), which is a common measure of voice quality based on user perception, according to:

$$\mathrm{R}x = \mathrm{R}/1.48 \tag{5.2.2}$$

$$\mathrm{MOS} = \begin{cases} 1 & \text{if } \mathrm{R}x < 0 \\ 4.5 & \text{if } \mathrm{R}x > 100 \\ 1 + 0.035\mathrm{R}x + \mathrm{R}x(\mathrm{R}x - 60)(100 - \mathrm{R}x) \times 7 \times 10^{-6} & \text{otherwise.} \end{cases} \tag{5.2.3}$$

This mapping helps translate the R-Factor into a more intuitive 1-5 scale. The E-model, with its comprehensive consideration of various impairments, provides a robust framework for predicting voice quality in telecommunications networks.

5.2.2 Related work

The growing demand for multimedia applications and services requires different models for managing client satisfaction and services qualities which reveal the need of a global centralized view of the network and management dynamic resource. In this context, Software Defined Network (SDN)

has emerged as one of the most popular networks where we can widely find the use of enormous amount of multimedia applications. This may include multimedia services like VoIP, telemedicine and gaming. The purpose of MNO being to improve user satisfaction, it was necessary to improve the QoS and application parameters which affects the perception of the users by reducing response time, decreasing packet loss, etc. As we already mentionned, this process is not cheap and not feasible in real time. To solve this issue we can instead use the ML models to predict the QoE from many parameters such as network parameters (packet loss, delay, etc.) and application parameters (bit rate, resolution, etc.). Currently there are several surveys that were carried out around QoE and ML models. We reported about some of them in Section 4.2.2. Another work from Abar et al. (2017) presents a method for predicting QoE in Software Defined Networks (SDN) using full reference parametric metrics (SSIM, VQM) addressing the limitations of subjective methods like Degradation Category Rating (DCR) methods. They use four algorithms such as Decision Tree (DT), Neural Network, K-Nearest Neighbors (KNN), and Random Forest (RF) due to their proven efficiency. In Rodriguez et al. (2013), authors are interested by VoIP and they decided to predict the MOS (calculated subjective method PESQ) using Decision tree, Neural network, Bayesian, they concluded also that Decision Tree is the best predictor. Finally, the work of Machado et al. (2011) had as objective the estimation of QoE metrics based on QoS metrics (throughput, packet loss, jitter, delay) in WiMAX networks using Artificial NN (ANN), they showed that ANN had a very good prediction with satisfactory errors on testing and validation steps.

5.3 Dataset and Data engineering

This section introduces the data and presents the transformations made on it to performs the different models.

5.3.1 Dataset presentation

We have at our disposal the data used in the German research project **BigQoE** (Schwarzmann et al., 2019) which worked on the same problem for two services (VoIP, VoD) in the context of fourth-generation (4G) mobile networks. These data were generated and collected on simulated communications using the OMNeT++ library (Varga, 2001). The simulations consist of a single access node which serves a varying number of active UEs (mobile terminal), which differ with respect to their mobility characteristics. For out target VoIP service, they consider calls lasting around 50 seconds for 400 mobile users within a 4G network cell were simulated, under 69 different network conditions. These 69 conditions are divided into :

- 4 sets of initial positions (labelled **ds0** to **ds3**) of the users in the cell ;
- 4 sets of target points to be reached (labelled **p1** to **p4**) where
  - **p1** : All users converge on the bottom left edge of the cell ;
  - **p2** : All users converge towards the center of the cell;
  - **p3** : users move to 10 predetermined points in the cell. The point to which a user will move depends not only on its proximity, but also on the number of points moving towards it;

  - **p4** : Same as **p3** but with 100 predetermined points.

- 3 distributions of user movement speed (labelled $\mathbf{s0-100}$, $\mathbf{s100-0}$ and $\mathbf{s50-50}$) where
  - $\mathbf{s0-100}$: All users move at the speed of a car (50 kmph);
  - $\mathbf{s100-0}$: All users move at pedestrian speed (3 kmph);
  - $\mathbf{s50-50}$: 50% for each of the two previous speeds.
- We thus have $4 \times 4 \times 3 = 48$ conditions for mobile users. In the 21 remaining conditions, labelled **ds0_nm** to **ds20_nm**, the 400 users are static and placed randomly in the cell.

In this dataset, we therefore have a total of around 400×69 i.e 27,000 clients, with multiple positions and movements (speed and direction), for which various network parameters have been collected at the UE, cell base station and network equipment level (server, gateway, router). In addition, since communication lasts around 50 seconds, each variable is collected as a time series, with a recording frequency specific to certain metrics. The difficulty here lies in the fact that, from one client to another, the collection times for a given metric are specific to that client. There are three different collection frequencies for our different metrics:

- A **"MOS frequency"** series where a recording is observed at the end of each spurt (the speech/silence distribution is made according to a probability law defined in the OMNeT++ simulation parameters). It concerns our target variable MOS, of course, but also other terminal metrics;
- A **"20 ms frequency"** series for the base station variables. And so, over a 50 s conversation, each client's series for one of these metrics has thousands of records;
- A **frequency less than 1 ms** which concerns network equipment metrics. However, for this last category, data is collected not for each user, but by simulation condition. In other words, all users in the same condition have the same values for these variables.

This data generated by the BigQoE project assumes that clients are using the G.726 coder standardized in ITU (1990), also known as Adaptive Differential Pulse Code Modulation or ADPCM, at a data rate of 32 kbit/s, which is mostly dedicated to fix-line networks and therefore does not reflect the configuration of current networks. As voice processing, and in particular voice coding, is a major factor influencing the perceived quality of telephony services. We therefore felt it necessary to include this element as an additional variable in our study. In addition to this G.726 codec, we agreed with the BigQoE project to extend the initial database with new conditions, using other coding technologies more tailored to mobile networks and of better quality, namely : AMR-WB at 12.65 kbit/s (used in 3G networks); AMR-WB at 23.85 kbit/s (used in 4G and 5G networks) both standardized in ITU (2003) ; and EVS at 24.4 kbit/s (used in 4G and 5G networks) in 3GPP (2020) standard.
For each of these codecs, we repeated the 69 initial conditions, this time applied to 800 users in the cell, i.e. a total of 250,000 clients. In order to obtain even more degraded transmission conditions, and thus a more balanced occupation of the MOS quality range between 1 and 5.

The network parameters collected and their temporal granularity are shown in the table below:

Terminal (UE)

Labels	Definition	Temporal granularity
`Xlocation`	Client position in the cell, x-axis	seconds
`Ylocation`	Client position in the cell, y-axis	seconds
`averageCQI`	CQI = Channel Quality Indicator, measures the quality of the radio link, downlink	20ms
`sentPacketToUpperLayer`	number of IP packets sent by the user to the network (total over unknown duration)	20ms
`rcvPk`	number of IP packets received by the user from the network (total over unknown duration)	20ms
`passedUpPk`	number of IP packets received by the client since the start of the session	20ms
`endToEndDelay`	total transmission time (network + terminal)	20ms
`MOS`	Mean Opinion Score of the voice quality (calculated using the E model of ITU-T G.107)	After each spurt
`packetLossRate`	Rate of IP packets lost on the section between base station and terminal	By each MOS score
`playoutDelay`	Terminal processing time	By each MOS score
`playoutLossRate`	Rate of IP packets lost during terminal processing	By each MOS score
`taildropLossRate`	Rate of IP packets dropped by receiving terminal (saturated receiving queue)	By each MOS score

Table 5.1: Collected metrics on UE

Base Station (LTE)

Labels	Definition	Temporal granularity
`rlcPacketLossDl`	Rate of IP packets lost on the section between the core network and the base station (Downlink)	20ms
`rlcPacketLossTotal`	Rate of IP packets lost on the section between the core network and the base station (Uplink & Downlink)	20ms
`rlcDelay`	Total transmission time from core network to base station	20ms
`harqErrorRateDl`	HARQ = Hybrid Automatic Repeat reQuest : error rate on data before radio transmission	20ms
`harqErrorRate_1st_Dl`	Error rate on data before radio transmission	20ms

Table 5.2: Collected metrics on base station (LTE)

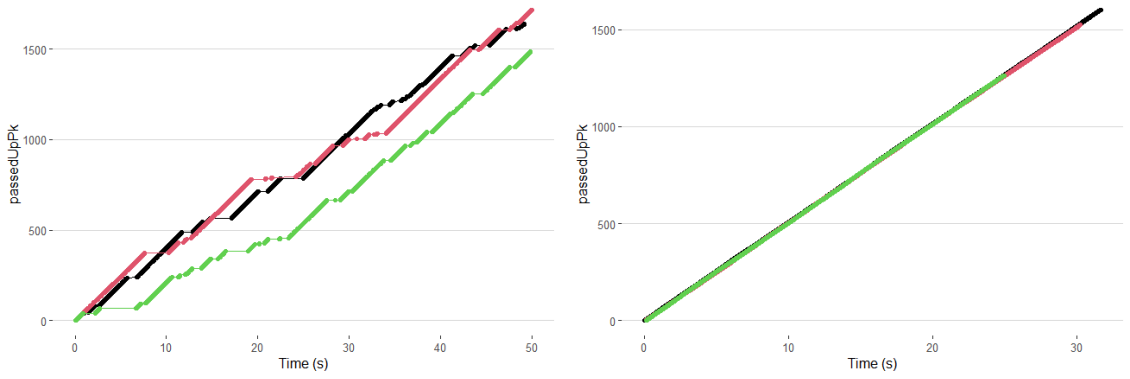
Network equipments (pgw, router, server)

Labels	Definition	Temporal granularity
`queueLength`	Length of queue where IP packets to be sent are stored	<1ms
`rcvdPk`	number of IP packets received by the device from the network (total over an unknown period)	<1ms
`queueingTime`	time an IP packet spends in the queue before being sent	<1ms
`txPk`	number of IP packets received by the device and sent to the network (total over unknown duration)	<1ms

Table 5.3: Collected metrics on network equipments

5.3.2 Data engineering

An initial observation of the different series previously showed us that they do not have the same level of granularity. Closer observation, particularly of the 20ms frequency's series, revealed that the latter have missing observations. Being in a simulation environment, we understand that this occurs when no voice transmission is taking place. More clearly, when none of the interlocutors is speaking, the simulation environment does not collect data. These time intervals have repercussions on all the metrics and add up to the real dynamics of the quantities we wish to have. It therefore seems appropriate, and even essential, to correct the various series for periods of inactivity, since we intend to take the time dimension very much into account. We can see from the series of 3 clients in the example Figure 5.1a below, for a variable (PassedUpPk, collected at the level of the device) that normally has a record every 20 ms (number of packets sent), that there are periods that seem to stop the time and modify its real dynamics. As can be seen in Figure 5.1, to be able to apply prediction models, we will first of all polish the time scale so that it becomes equivalent for all conversations (Figure 5.1b).



(a) Before **(b)** After

Figure 5.1: Series of number of packets passed for 3 users, before and after the correction of inactivity periods.

The choice of the metric `passedUpPk` is based on the fact that, it is only a counter for IP packets sent to the client, and should therefore have the same evolution: simple time series of steps of 20ms, with a step equal to the size of an IP packet. To remove these periods of inactivity from the network require a transformation of the time scale. An additional issue arising from the form of the data, and which in fact derives from the previous treatment, is the temporal alignment of our observations. The different observations we have on clients are assumed to be realizations of a random process, and series functions based on different time interval would violate this fundamental assumption. We are therefore faced with the need to correct this aspect of time scales. To this end, we have explored two options and noted their advantages and limitations.

- The first, which correspond to the more realistic situation and that we're not going to retain consisted in: considering that the set of series is defined on the single interval $\mathcal{T} = [0\,;\, t_{max}]$ with t_{max} the final time of the longest conversation. Thus, in this approach, we consider that we have time series actually defined over an interval, but for which we only have observations over part of this interval. The reasons for not adopting this approach is due to the fact that by representing the functional basis expansion, we get at the beginning and at the end of curves a high variability.

- The second explored method consists in stretching the time scale : Define the final time t^i_{max} for all the metrics of a given client i as the time of the last MOS note collected, i.e. for each metric of this client i delete all data collected after t^i_{max} (in reality, this doesn't represent much data); then find the maximum final instant $t_{max} = \max_i t^i_{max}$ and finally "calibrate" all the series on this maximum interval by transforming $\dfrac{t_{max}}{t^i_{max}}$ from the client i's time grid.

In this study, for each metric in the database, we have observations (users) which not only have a temporal dimension, but more importantly are not collected at the same time and in the same numbers from one user to another. As a result, we can't use traditional analysis without simplifying the data. This is exactly what the BigQoE project, which provided us with the data, did. They modeled the average of each user's target metric series by the central trend and dispersion characteristics of the explanatory metric series. The advantage of this method is that it is simple to implement and gives good results on the mean of our target metric (MOS). Data are therefore collected in the form of :

$$\begin{bmatrix} \mathrm{Y_1} \\ \mathrm{Y_2} \\ \mathrm{Y_3} \\ \vdots \\ \mathrm{Y_n} \end{bmatrix} = \begin{bmatrix} \mathrm{y_{11}} & \mathrm{y_{12}} & \dots & \dots & \dots & \mathrm{y_{1m_1}} & \\ \mathrm{y_{21}} & \mathrm{y_{22}} & \dots & \mathrm{y_{2m_2}} & & & \\ \mathrm{y_{31}} & \mathrm{y_{32}} & \dots & \dots & \dots & \dots & \mathrm{y_{3m_3}} \\ \vdots & \vdots & & & & & \\ \mathrm{y_{n1}} & \mathrm{y_{n2}} & \dots & \dots & \mathrm{y_{nm_n}} & & \end{bmatrix} ; \begin{bmatrix} \mathrm{X_1^k} \\ \mathrm{X_2^k} \\ \mathrm{X_3^k} \\ \vdots \\ \mathrm{X_n^k} \end{bmatrix} = \begin{bmatrix} \mathrm{x_{11}^k} & \mathrm{x_{12}^k} & \dots & \dots & \dots & \mathrm{x_{1m_1}^k} & \\ \mathrm{x_{21}^k} & \mathrm{x_{22}^k} & \dots & \mathrm{x_{2m_2}^k} & & & \\ \mathrm{x_{31}^k} & \mathrm{x_{32}^k} & \dots & \dots & \dots & \dots & \mathrm{x_{3m_3}^k} \\ \vdots & \vdots & & & & & \\ \mathrm{x_{n1}^k} & \mathrm{x_{n2}^k} & \dots & \dots & \mathrm{x_{nm_n}^k} & & \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \overline{Y}_1 \\ \overline{Y}_2 \\ \overline{Y}_3 \\ \vdots \\ \overline{Y}_n \end{bmatrix} \sim \begin{bmatrix} \overline{X_1^k} & \mathrm{Var}(X_1^k) & q_\tau(X_1^k) & \mathrm{Sk}(X_1^k) & \dots \\ \overline{X_2^k} & \mathrm{Var}(X_2^k) & q_\tau(X_2^k) & \mathrm{Sk}(X_2^k) & \dots \\ \overline{X_3^k} & \mathrm{Var}(X_3^k) & q_\tau(X_3^k) & \mathrm{Sk}(X_3^k) & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \\ \overline{X_n^k} & \mathrm{Var}(X_n^k) & q_\tau(X_n^k) & \mathrm{Sk}(X_n^k) & \dots \end{bmatrix}$$

In the FDA framework, we instead consider each observation of a metric as a realization of random function. For each conversation, we have several MOS scores calculated at various times throughout its duration. The important point to note is that these time points and their number were completely independent from one conversation to another. Thus, the response variable is presented in the following way:

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} y_{11} & y_{12} & \dots & \dots & \dots & y_{1m_1} & \\ y_{21} & y_{22} & \dots & y_{2m_2} & & & \\ y_{31} & y_{32} & \dots & \dots & \dots & \dots & y_{3m_3} \\ \vdots & \vdots & & & & & \\ y_{n1} & y_{n2} & \dots & \dots & y_{nm_n} & & \end{bmatrix} \quad \text{Where } y_{ij} = Y_i(t_{ij})$$

For the independents covariates, they are listed in Table 5.1, Table 5.2 and Table 5.3 including various voice coding and transmission parameters collected from the UE, the core network and base station. To these we have also added the configuration parameters of the simulation scenarios carried out (codec type, number of clients, speed of movement in the cell) throughout a conversation,

- The network parameters X^k are either collected at the same time as the voice quality score is calculated, or every 20ms

$$X^k = \begin{bmatrix} X_1^k \\ X_2^k \\ X_3^k \\ \vdots \\ X_n^k \end{bmatrix} = \begin{bmatrix} x_{11}^k & x_{12}^k & \dots & \dots & \dots & x_{1m_1}^k & \\ x_{21}^k & x_{22}^k & \dots & x_{2m_2}^k & & & \\ x_{31}^k & x_{32}^k & \dots & \dots & \dots & \dots & x_{3m_3}^k \\ \vdots & \vdots & & & & & \\ x_{n1}^k & x_{n2}^k & \dots & \dots & x_{nm_n}^k & & \end{bmatrix} \quad \text{with } x_{ij}^k = X_i^k(s_{ij}^k)$$

$$\text{where} \quad X^k = \begin{bmatrix} X_1^k \\ X_2^k \\ X_3^k \\ \vdots \\ X_n^k \end{bmatrix} = \begin{bmatrix} x_{11}^k & x_{12}^k & \dots & x_{1m}^k \\ x_{21}^k & x_{22}^k & \dots & x_{2m}^k \\ x_{31}^k & x_{32}^k & \dots & x_{3m}^k \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1}^k & x_{n2}^k & \dots & x_{nm}^k \end{bmatrix}$$

- We also have parameters collected at network equipment level (Router, Server, PGW). These are collected at a frequency of 1 ms (or less) and, like the simulation parameters, are identical for all clients in a simulation condition. We'll refer to them as scalar variables $Z = (Z_1, \dots, Z_q)$. Other scalars covariates used are essentially the OMNeT++ simulation parameters. The following Table 5.4 lists them :

Labels	Definition
`pos.init`	Initial configuration of cell clients (21 modalities)
`pos.final`	Configuration of targets to be reached when clients move in the cell (4 types)
`config.vit`	Client travel speed configuration (3 types)
`codec`	Codec type used (4 types)
`rcvPk_pgw`, `rcvPk_server`, `rcvPk_router`	Number of IP packets received by the device from the network at base station, server and router level respectively.
`queueingtime_pgw`, `queueingtime_server`, `queueingtime_router`	Time an IP packet spends in the queue before being forwarded to the base station, server and router respectively.
`txPk_pgw`, `txPk_server`, `txPk_router`	Number of IP packets received by the equipment to the network at base station, server and router level respectively.

Table 5.4: Collected and computed scalar covariates

5.4 Prediction model for the VoIP QoE

The final goal of this work is to be able to detect, at any moment of a conversation, the network parameters that contribute to the degradation of voice quality. Given the nature and type of data available to achieve this, it was appropriate to delve into the framework of FDA in order to make relevant analysis. Given the variability of network simulation conditions, a single regression structure for all data may seem simplistic. For this reason, we delved into the framework of expert mixture models to handle heterogeneous data more effectively by tailoring predictions to specific conditions.

The FFMoE model was covered in Chapter 3 where the estimation scheme for concurrent regression model are presented. Recall that for a K components mixture model with $\mathrm{Z} = (z_1, \ldots, z_\mathrm{K})$ as class membership variable, the MoE is defined by

$$\mathrm{MoE}\Big(\mathrm{Y}(t)\,|\,\mathrm{X}(t)\Big) \quad = \quad \sum_{k=1}^{\mathrm{K}} \pi_k\Big(\mathrm{X}(t), \alpha_k(t)\Big)\,\mathbb{E}\Big[\mathrm{Y}(t)\,|\,\mathrm{X}(t), z_k = 1\Big] \tag{5.4.1}$$

The corresponding conditional density of $\mathrm{Y}(t)$ given covariates $\mathrm{X}(t)$ according to the FFMoE model is given by

$$f(\mathrm{Y}(t)|\mathrm{X}(t), \Psi(t)) \quad = \quad \sum_{k=1}^{\mathrm{K}} \pi_k(\mathrm{X}(t), \alpha_k(t))\Phi(\mathrm{Y}(t); \mathrm{X}(t)\beta_k(t), \sigma_k^2), \tag{5.4.2}$$

where

- $\pi_k(\mathrm{X}(t),\ \alpha_k(t))$ is the gated network function;
- $\Psi_k(t) = (\beta_k(t), \alpha_k(t))$ are the functional parameters;
- $\Phi(\mathrm{Y}(t); \mathrm{X}_i(t)\beta_k(t), \sigma_k^2)$ is the Gaussian density probability function of mean $\mathrm{X}(t)\beta_k(t)$ and variance σ_k^2.

After functional basis expansion where the functional design matrix $\mathrm{X}(t)$ is transformed a multivariate one $\mathrm{R}(t_j)$, the parameters are estimated using EM algorithm. The predictions can be build by firstly compute the conditional probabilities that any clients i belongs to a component k given by:

$$\hat{\pi}_k(\mathrm{X}_i(t),\, \widehat{\alpha}_k) = \frac{\exp(\widehat{a}_k^\top r_i)}{1 + \sum_{v=1}^{\mathrm{K}-1} \exp(\widehat{a}_v^\top r_i)}$$

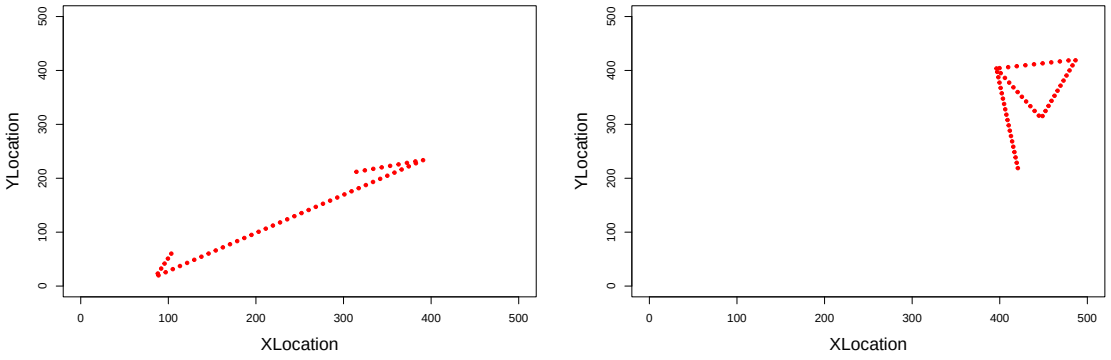
where $\widehat{a}_k$ for $1 \leq k \leq \mathrm{K}-1$ are the gated parameters estimators. Then deduce, where component k_m is the most probable class for the i curve, the predictive curve by:

$$\widehat{\mathrm{Y}}_i(t) = \widehat{b}_{k_m}^\top \mathrm{R}_i(t).$$

Applying the FFMoE model to predict the QoE of VoIP service addresses the complexity and variability of network conditions. It also offers several advantages but the main one is the interpretability. Indeed, by using functional linear regression within each expert model, we ensure a comprehensible relationship with the linear one that link network metrics and QoE.

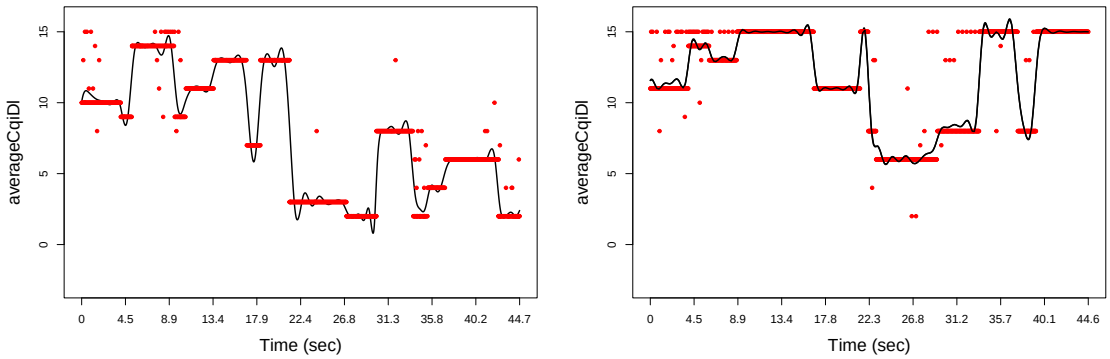
5.5 Experiments and results

We evaluated on the presented dataset not only the proposed MoE method FFMoE and PenFFMoE but also the non mixture methods PenFFR and pffr. The goal is to assess that the mixture model's performance in the usecase of VoIP service can be effectively improved compared to non mixture models. We used for evaluation classical predictive accuracy criteria such as MAE, RMSE dedicated for capturing the overall accuracy of the model. We can also use, only for mixture models, the explainability given by the formed clusters. We thus compare for example the formed clusters and the type of codec using confusion matrix.



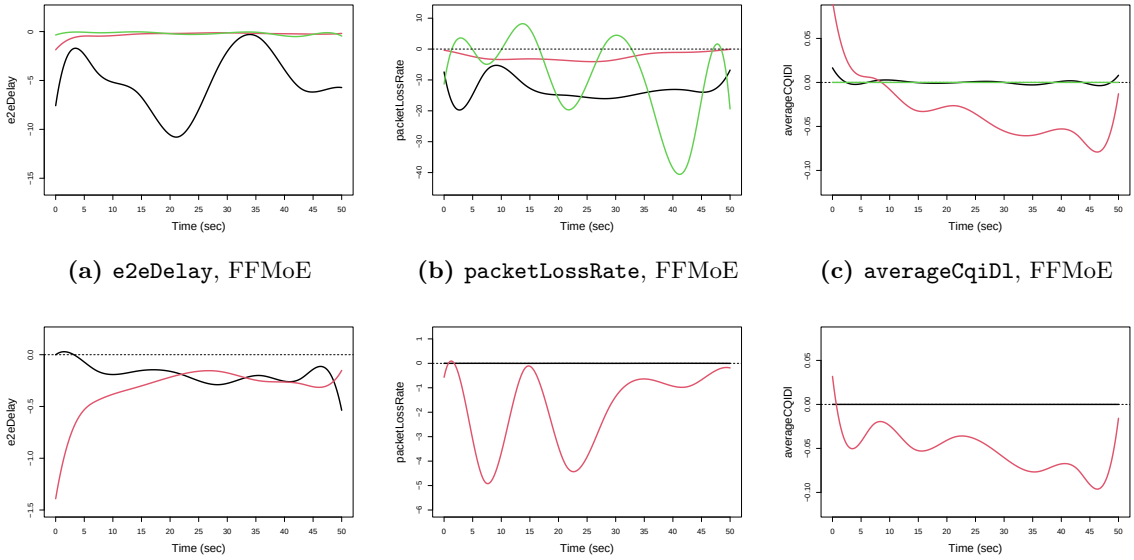
(a) Conversation 1 **(b)** Conversation 2

Figure 5.2: Position (`Xloc, Yloc`) of the user in the 4G cell for two randomly chosen conversations



(a) Conversation 1 **(b)** Conversation 2

Figure 5.3: Raw data (red dots) and the corresponding functional basis expansion curve (black line) for the `averageCqiDl` metric of two randomly chosen conversations



(a) `e2eDelay`, FFMoE **(b)** `packetLossRate`, FFMoE **(c)** `averageCqiDl`, FFMoE

(d) `e2eDelay`, PenFFMoE **(e)** `packetLossRate`, PenFFMoE **(f)** `averageCqiDl`, PenFFMoE

Figure 5.4: Estimated functional experts for FFMoE (first row) and PenFFMoE (second row) obtained on BigQoE data. Corresponding colors matched for the same cluster.

The number of basis functions is set to $L_\beta = 10$ both for FFMoE and PenFFMoE in order to obtain smooth estimates of parameter curves. This number is chosen according to the average number of raw observation we have for each curve. For the non mixture model PenFFR and pffr, the

number of basis function is set to $L_\beta = 25$ due to the fact that the number of parameters of the non mixture model is nearly multiplied by the number of components to get the number of parameters of the mixture model. So we set more basis functions for the non mixture model to get the ability to handle more complex shapes. The penalty parameter λ is selected using cross-validation on predefined grid of values. Model selection is made using BIC criterion with the number of expert components K in the set $\{1, 2, 3, 4, 5\}$. We obtained K = 3 for FFMoE and K = 2 for PenFFMoE with a lower BIC for the PenFFMoE method.

		Clusters					
		FFMoE				**PenFFMoE**	
		1	2	3		1	2
Codec type	G.726	22.7	59.3	18.0		82.5	17.5
	AMR-WB 12.65kbit/s	9.1	12.3	78.6		23.4	76.6
	AMR-WB 23.85kbit/s	15.2	9.3	75.5		23.2	76.8
	EVS	15.5	7.3	77.2		21.8	78.2

Table 5.5: Confusion matrix between produced clusters and codec type for FFMoE and PenFFMoE methods.

		Clusters					
		FFMoE				**PenFFMoE**	
		1	2	3		1	2
User speed	**static**	13.3	29.5	57.2		43.5	56.5
	s0-100	22.1	37.4	40.5		60.4	39.6
	s100-0	19.1	25.3	55.6		44.3	55.7
	s50-50	22.4	26.7	50.9		49.8	50.2

Table 5.6: Confusion matrix between produced clusters and the type of movement for FFMoE and PenFFMoE methods.

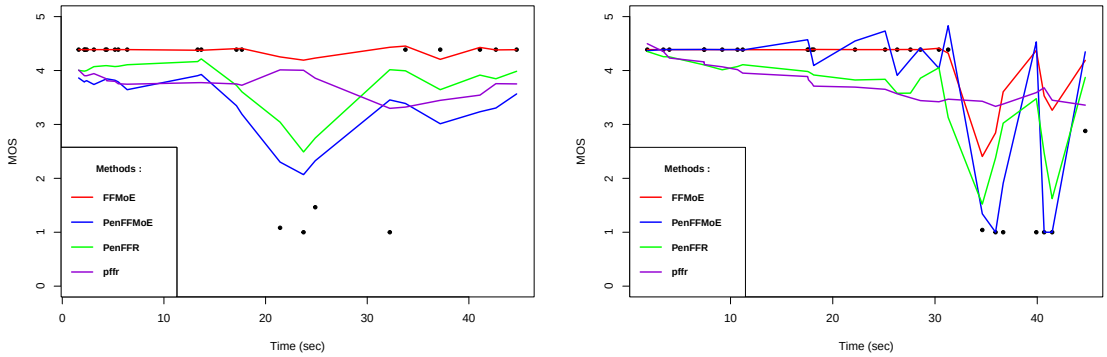
To explain the forming components given by the two mixture methods, we will compare the component membership of each train conversation to simulation conditions such as users mobility, the using codec type and the speed of the users. According to the results, it seems that in the FFMoE method, clusters **1** and **2** are combined to form cluster **1** of PenFFMoE. Table 5.5 shows, in proportion, the confusion matrix given by the produced components and the codec type. We note that for any codec type, we have a clear predominant cluster for PenFFMoE (**1** for G.726, **2** for the others). In Table 5.6, the difference between cluster is less pronounced. This means in a 4G cell, the user speed doesn't not affect the QoE of the voice call which may be different when it moves from one cell to another.

Table 5.7 compares on the test set the performances of the four presented methods through MAE and RMSE. By looking on the MAE metric, PenFFMoE outperform all the other methods that indicating that it provides the most accurate predictions in terms of absolute error. Additionally,

it reduces the RMSE significantly compared to its non penalized version FFMoE, showing that it handles larger errors better. By looking on the RMSE, PenFFR achieves the lowest value among all methods, which indicates that it handles large errors more effectively than any other method in the table. It is also competitive in terms of MAE, showing that it is a robust method for this prediction task. Overall, it appears that penalization improves performance across both mixture and non mixture models, as evidenced by the reduced MAE and RMSE which highlighting its importance both for the predictive accuracy and for the interpretability of parameters. This suggests that regularization helps in controlling overfitting and improving generalization.

Methods	MAE	RMSE
FFMoE	0.704	1.045
PenFFMoE	**0.670**	0.922
pffr	0.826	1.077
PenFFR	0.678	**0.892**

Table 5.7: Results of RMSE and OR metrics for the 4 methods



(a) Conversation 1 **(b)** Conversation 2

Figure 5.5: Prediction on two randomly chosen conversations. Blacks dots are the actual data, and red and blue lines are the predictions for FFMoE and PenFFMoE resp. The green and violet lines are the predictions given by PenFFR and pffr resp.

5.6 Conclusion and future works

In this chapter, we have explored various methods for predicting the QoE of VoIP service, focusing on their ability to capture the underlying linear relationships between network performance parameters and user-perceived quality. We compares several models in the framework of MoE and linear regression designed for fully functional data. The results reveal that penalized methods, such as PenFFMoE and PenFFR, offer robust performance, outperforming their unpenalized counterparts

by effectively controlling overfitting and capturing complex, linear patterns in the data. These methods demonstrate superior predictive capabilities, with PenFFMoE excelling in minimizing average error (MAE) and PenFFR showing remarkable ability in reducing larger errors (RMSE). This indicates that a careful balance between model complexity and regularization is key to achieving accurate QoE predictions for VoIP services.

Overall, this chapter highlights the importance of selecting appropriate modeling techniques when addressing QoE prediction challenges in VoIP networks. By focusing on both the accuracy and robustness of the models, we contribute to the ongoing efforts to enhance the end-user experience in modern telecommunications. Future work may involve extending these methods to accommodate even more complex network environments and exploring the integration of additional covariates, such as user behavior or contextual factors, to further refine QoE predictions.

Chapter 6

Conclusion

This thesis presented here has advanced our understanding of QoE in the context of internet-based communication services, addressing the critical challenges associated with its measurement and prediction. Traditional QoE prediction models have often struggled to capture the complex temporal dynamic of factors influencing user experience, particularly in scenarios where the response variable i.e. QoE is also functional, observed over continuous domains like time. We made significant contributions by integrating the FDA and the MoE frameworks, creating a novel and robust methodology for QoE prediction. By leveraging the power of FDA, which treats data as entire functions rather than discrete points, and the MoE approach, which partitions data into more homogeneous subspaces, the research has developed a flexible model capable of handling the inherent heterogeneity in QoE data. More precisely,

In Chapter 2, the two proposed function-on-function regression models we developed known as concurrent and integral models have demonstrated the ability to account for the dynamic nature of QoE, providing more accurate and reliable predictions than traditional methods. Then, a regularized model designed by imposing a Lasso-type penalty on second derivatives of parameters provides interpretable parameters. Finally, we use conformal prediction for producing confidence bands for the functional response. Indeed, we draw a functional quantile regression by perturbing the standard functional linear regression and compute functional quantiles by optimal transport. This functional quantile regression is then combined to the conformalized method to build robust confident intervals of the predicted functional QoE. This work not only enhances the theoretical understanding of QoE but also offers practical tools for network operators and content providers to better manage and optimize the quality of their services. Experiments on simulated data and real data showed the effectiveness of our approach. Application of this method to the streaming video use case is also provide in Chapter 4.

In Chapter 3, beyond the FDA methodology we addressed the limitations of the previous model for handling heterogeneous data by the MoE approach. In the presence of heterogeneity, assumption that a unique relationship (model) between response and covariates holds for the full dataset may not be valid. Mixture model is a powerful framework for capturing sub-population behavior through a finite set of empirical latent class. In a predictive modelling, the MoE setup is more relevant as the mixing proportions are covariate-dependent. We thus developed an estimation method for MoE regression for functional response and functional covariate named FFMoE. Like most of the inference approach for models on functional data, we use basis expansion (B-splines) both for

covariates and parameters. A regularized approach is also proposed based on Lasso-type penalty on second derivatives both for gated and parameters. It accurately smoothes functional parameters in order to provide interpretable estimators. The usefulness of the proposed model is illustrated on simulated and real data sets. Application for VoIP use case is provide in Chapter 5 which show compared to the non mixture models the relevance of the method.

The potential improvements to be made to this work relate firstly on the using models. The first above all for us is the development and implementation of the (historical) integral model in the MoE framework which are not made in the current work. Mostly due to multicolinearity and some convergence issues occurring in the EM algorithm for the maximization of the likelihood. In addition, in some practical applications the relationship between network performance metrics and QoE might be non-linear. So the simple linear function-on-function regression may not capture this complexity. The second improvement relate on the quality of the data used in the models. Improving the QoE prediction necessitates to capture all relevant metrics affecting QoE, including network-level metrics (e.g., bandwidth, latency, jitter), application-level metrics (e.g., video resolution, frame rate), and user-level metrics (e.g., interaction times, geographic location). It is not a mystery that incomplete and inaccurate data can severely impact the performance of the model. The data may also be imbalanced, where negative QoE experiences (e.g., service disruptions) are rare compared to positive experiences in the collected data. This lead to bias models toward overpredicting good QoE.

To sum up, this dissertation has addressed the limitations of existing QoE prediction models by introducing innovative statistical methodologies within the FDA and the MoE frameworks. These advancements pave the way for future research and applications in QoE assessment, ensuring that the rapidly growing demand for high-quality internet-based communication services can be met with greater precision and effectiveness.

List of Publications and Communications

Journal papers

- Tamo Tchomgui, J.S. and Jacques, J. and Fraysse, G. and Barriac, V. and Chretien, S. A mixture of experts regression model for functional response with functional covariates. Stat Comput 34, 154 (2024). https://doi.org/10.1007/s11222-024-10455-z
- Tamo Tchomgui, J.S. and Jacques, J. and Barriac, V. and Fraysse, G. and Chretien, S. (2024). A Penalized Spline Estimator for Functional Linear Regression with Functional Response. *Working paper (submitted).* hal-04120709.

Conference papers and presentations

- Tamo Tchomgui, J.S. and Barriac, V. and Fraysse, G. and Jacques, J. and Chretien, S. Functional Linear Regression for the prediction of streaming video QoE. *20th International Conference on Network and Service Management, October 2024, Prague, Czech Republic.*
- Tamo Tchomgui, J.S. and Jacques, J. and Fraysse, G. and Barriac, V. and Chretien, S. Penalized Spline Regression for Gaussian Function-on-Function Mixture-of-Experts. *ENBIS-24 Conference, European Network for Business and Industrial Statistics (ENBIS), September 2024, Leuven, Belgium.*
- Jacques, J. and Tamo Tchomgui, J.S. Mixture of function-on-funtion regression models. *Royal Statistical Society 2024 International Conference, September 2024, Brighton, United Kingdom.*
- Tamo Tchomgui, J.S. and Jacques, J. and Chretien, S. and Fraysse, G. and Barriac, V. Function-on-Function Mixture-of-Experts Regression. *fda-lille : Functional Data Analysis Workshop, March 2024, Lille, France.*
- Tamo Tchomgui, J.S. and Jacques, J. and Chretien, S. and Fraysse, G. and Barriac, V. Modèle de mélanges d'experts pour données fonctionnelles. *54es Journées de la Statistique de la SFdS, Société Française de Statistique (SFdS), July 2023, Bruxelles, Belgique.*
- Tamo Tchomgui, J.S. and Jacques, J. and Chretien, S. and Fraysse, G. and Barriac, V. Prédiction de la Qualité d'Expérience dans les Réseaux Mobiles : Cas de la VoIP. *53es journées de la Statistique de la Société Française de Statistique (SFdS), June 2022, Lyon, France.*

- Tamo Tchomgui, J.S. and Jacques, J. and Chretien, S. and Fraysse, G. and Barriac, V. Function-on-Function Mixture of Experts Regression Models. *15th International Conference of the ERCIM WG on Computational and Methodological Statistics. 16th International Conference on Computational and Financial Econometrics, December 2022, London, United Kingdom.*

Bibliography

3GPP (2020). 3GPP TS 26.445: Codec for Enhanced Voice Services (EVS); Detailed Algorithmic Description. Technical Report TS 26.445, 3GPP, Sophia Antipolis, France.

Aaron, A., Li, Z., Manohara, M., Lin, J. Y., Wu, E. C.-H., and Kuo, C.-C. J. (2015). Challenges in cloud based ingest and encoding for high quality streaming media. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 1732–1736.

Abar, T., Ben Letaifa, A., and El Asmi, S. (2017). Machine learning based qoe prediction in sdn networks. In *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 1395–1400.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.

Angelopoulos, A. N. and Bates, S. (2023). Conformal prediction: A gentle introduction.

Antoch, J., Prchal, L., Rosaria De Rosa, M., and Sarda, P. (2010). Electricity consumption prediction with functional linear regression using spline estimators. *Journal of Applied Statistics*, 37(12):2027–2041.

Bampis, C. G. and Bovik, A. C. (2017a). An augmented autoregressive approach to HTTP video stream quality prediction. *CoRR*, abs/1707.02709.

Bampis, C. G. and Bovik, A. C. (2017b). Learning to predict streaming video qoe: Distortions, rebuffering and memory. *CoRR*, abs/1703.00633.

Bampis, C. G., Li, Z., Katsavounidis, I., and Bovik, A. C. (2018a). Recurrent and dynamic models for predicting streaming video quality of experience. *IEEE Transactions on Image Processing*, 27(7):3316–3331.

Bampis, C. G., Li, Z., Katsavounidis, I., Huang, T.-Y., Ekanadham, C., and Bovik, A. C. (2018b). Towards perceptually optimized end-to-end adaptive video streaming. *arXiv preprint arXiv:1808.03898*.

Ben Slimen, Y., Allio, S., and Jacques, J. (2017). Anomaly prevision in radio access networks using functional data analysis. In *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, pages 1–6.

Berrendero, J. R., Bueno-Larraz, B., and Cuevas, A. (2023). On functional logistic regression: some conceptual issues. *TEST: An Official Journal of the Spanish Society of Statistics and Operations Research*, 32(1):321–349.

Besse, P. C. and Cardot, H. (1996). Approximation spline de la prevision d'un processus fonctionnel autorégressif d'ordre 1. *Canadian Journal of Statistics*, 24(4):467–487.

Besse, P. C., Cardot, H., and Ferraty, F. (1997). Simultaneous non-parametric regressions of unbalanced longitudinal data. *Computational Statistics & Data Analysis*, 24(3):255–270.

Beyaztas, U., Shang, H., and Saricam, S. (2024). Penalized function-on-function linear quantile regression. *Computational Statistics*.

Cai, X., Xue, L., Cao, J., and for the Alzheimer's Disease Neuroimaging Initiative (2022). Robust estimation and variable selection for function-on-scalar regression. *Canadian Journal of Statistics*, 50(1):162–179.

Cardot, H., Crambes, C., and Sarda, P. (2005). Quantile regression when the covariates are functions. *Nonparametric Statistics*, 17(7):841–856.

Cardot, H., Ferraty, F., Mas, A., and Sarda, P. (2003). Testing hypotheses in the functional linear model. *Scandinavian Journal of Statistics*, 30(1):241–255.

Cardot, H., Ferraty, F., and Sarda, P. (1999). Functional linear model. *Statistics & Probability Letters*, 45(1):11–22.

Celeux, G., Chrétien, S., Forbes, F., and Mkhadri, A. (2001). A component-wise em algorithm for mixtures. *Journal of Computational and Graphical Statistics*, 10(4):697–712.

Centofanti, F., Fontana, M., Lepore, A., and Vantini, S. (2022). Smooth lasso estimator for the function-on-function linear regression model. *Computational Statistics & Data Analysis*, 176:107556.

Chamroukhi, F., Pham, N. T., Hoang, V. H., and McLachlan, G. J. (2022). Functional mixtures-of-experts. *arXiv preprint arXiv:2202.02249*.

Chen, C., Choi, L., Veciana, G., Caramanis, C., Heath, R., and Bovik, A. (2014). Modeling the time—varying subjective quality of http video streams with rate adaptations. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 23:2206–21.

Chen, H. and Wang, Y. (2011). A penalized spline approach to functional mixed effects model analysis. *Biometrics*, 67(3):861–870.

Chernozhukov, V., Galichon, A., Hallin, M., and Henry, M. (2017). Monge–Kantorovich depth, quantiles, ranks and signs. *The Annals of Statistics*, 45(1):223 – 256.

Chrétien, S. and Hero, A. (1998). Acceleration of the em algorithm via proximal point iterations. In *Proceedings. 1998 IEEE International Symposium on Information Theory (Cat. No. 98CH36252)*, page 444. IEEE.

Chrétien, S., Hero, A., and Perdry, H. (2012). Space alternating penalized kullback proximal point algorithms for maximizing likelihood with nondifferentiable penalty. *Annals of the Institute of Statistical Mathematics*, 64:791–809.

Chrétien, S. and Hero, A. O. (2000). Kullback proximal algorithms for maximum-likelihood estimation. *IEEE transactions on information theory*, 46(5):1800–1810.

Cisco (2019). Global mobile data traffic forecast update 2017-2022 - cisco visual networking index white paper.

Crainiceanu, C. M. and Goldsmith, A. J. (2010). Bayesian functional data analysis using winbugs. *Journal of statistical software*, 32(11).

Crainiceanu, C. M., Staicu, A.-M., and Di, C.-Z. (2009). Generalized multilevel functional regression. *Journal of the American Statistical Association*, 104(488):1550–1561. PMID: 20625442.

Das, S., Demirer, R., Gupta, R., and Mangisa, S. (2019). The effect of global crises on stock market correlations: Evidence from scalar regressions via functional data analysis. *Structural Change and Economic Dynamics*, 50:132–147.

Dayton, C. M. and Macready, G. B. (1988). Concomitant-variable latent-class models. *Journal of the american statistical association*, 83(401):173–178.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22.

DeSarbo, W. S. and Cron, W. L. (1988). A maximum likelihood methodology for clusterwise linear regression. *Journal of classification*, 5(2):249–282.

Devijver, E. (2015). Finite mixture regression: A sparse variable selection by model selection for clustering. *Electronic Journal of Statistics*, 9(2):2642 – 2674.

Dobreff, G., Szalay, M., Ladóczki, B., Molnár, M., Varga, L., Báder, A., and Pašić, A. (2023). Data collection framework for end-to-end radio and transport network quality monitoring. In *2023 15th International Conference on Quality of Multimedia Experience (QoMEX)*, pages 127–130. IEEE.

Duanmu, Z., Zeng, K., Ma, K., Rehman, A., and Wang, Z. (2016). A quality-of-experience index for streaming video. *IEEE Journal of Selected Topics in Signal Processing*, PP:1–1.

Ericsson (2024). Mobile data traffic forecast – Mobility Report.

Ferraty, F., Hall, P., and Vieu, P. (2010). Most-predictive design points for functional data predictors. *Biometrika*, 97(4):807–824.

Ferraty, F. and Nagy, S. (2021). Scalar-on-function local linear regression and beyond. *Biometrika*, 109(2):439–455.

Ferraty, F., Rabhi, A., and Vieu, P. (2005). Conditional quantiles for dependent functional data with application to the climatic" el niño" phenomenon. *Sankhya: The Indian Journal of Statistics*, 67.

Fessler, J. A. and Hero, A. O. (1994). Space-alternating generalized expectation-maximization algorithm. *IEEE Transactions on signal processing*, 42(10):2664–2677.

Gauss, C. (1809). *Theoria motus corporum coelestium in sectionibus conicis solem ambientium*. Carl Friedrich Gauss Werke. Sumtibus F. Perthes et I.H. Besser.

Gertheiss, J., Rügamer, D., Liew, B. X. W., and Greven, S. (2023). Functional Data Analysis: An Introduction and Recent Developments. *arXiv preprint arXiv:2312.05523*.

Ghadiyaram, D., Chen, C., Inguva, S., and Kokaram, A. (2017). A no-reference video quality predictor for compression and scaling artifacts. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3445–3449. IEEE.

Ghadiyaram, D., Pan, J., and Bovik, A. C. (2015). A time-varying subjective quality model for mobile streaming videos with stalling events. In Tescher, A. G., editor, *Applications of Digital Image Processing XXXVIII*, volume 9599 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 959911.

Goldsmith, J., Bobb, J., Crainiceanu, C. M., Caffo, B., and Reich, D. (2011). Penalized functional regression. *Journal of Computational and Graphical Statistics*, 20(4):830–851. PMID: 22368438.

Goldsmith, J., Crainiceanu, C. M., Caffo, B., and Reich, D. (2012). Longitudinal penalized functional regression for cognitive outcomes on neuronal tract measurements. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 61(3):453–469.

Goldsmith, J., Huang, L., and Crainiceanu, C. M. (2014). Smooth scalar-on-image regression via spatial bayesian variable selection. *Journal of Computational and Graphical Statistics*, 23(1):46–64.

Goldsmith, J. and Scheipl, F. (2014). Estimator selection and combination in scalar-on-function regression. *Computational Statistics & Data Analysis*, 70:362–372.

Greven, S. and Scheipl, F. (2017). A general framework for functional regression modelling. *Statistical Modelling*, 17(1-2):1–35.

Grollemund, P.-M., Abraham, C., Baragatti, M., and Pudlo, P. (2016). Bayesian functional linear regression with sparse step functions. *Bayesian Analysis*, 14.

Grün, B. and Leisch, F. (2008a). Flexmix version 2: Finite mixtures with concomitant variables and varying and constant parameters. *Journal of Statistical Software*, 28(4):1–35.

Grün, B. and Leisch, F. (2008b). Identifiability of Finite Mixtures of Multinomial Logit Models with Varying and Fixed Effects. *Journal of Classification*, 25(2):225–247.

Guéguin, M. (2006). *Evaluation objective de la qualité vocale en contexte de conversation*. Theses, Université Rennes 1.

Gurer, S., Shang, H. L., Mandal, A., and Beyaztas, U. (2024). Locally sparse and robust partial least squares in scalar-on-function regression. *Statistics and Computing*, 34.

Hallin, M. (2022). Measure transportation and statistical decision theory. *Annual Review of Statistics and Its Application*, 9.

Hallin, M., del Barrio, E., Cuesta-Albertos, J., and Matrán, C. (2021). Distribution and quantile functions, ranks and signs in dimension d: A measure transportation approach. *The Annals of Statistics*, 49(2):1139 – 1165.

Harezlak, J., Coull, B. A., Laird, N. M., Magari, S. R., and Christiani, D. C. (2007). Penalized solutions to functional regression problems. *Computational statistics & data analysis*, 51(10):4911–4925.

Hastie, T. and Tibshirani, R. (1993). Varying-coefficient models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 55(4):757–796.

Hida, T., Hui-Hsiung, K., Potthoff, J., and Streit, L. (1993). *White Noise: An Infinite Dimensional Calculus*. Mathematics and its applications. Kluwer Academic Publishers.

Horváth, L. and Kokoszka, P. (2012). *Inference for Functional Data with Applications*. Springer Series in Statistics. Springer-Verlag, New York.

Huang, T., Saporta, G., Wang, H., and Wang, S. (2020). A robust spatial autoregressive scalar-on-function regression with t-distribution. *Advances in Data Analysis and Classification*.

Huang, T.-Y., Johari, R., McKeown, N., Trunnell, M., and Watson, M. (2014). A buffer-based approach to rate adaptation: evidence from a large video streaming service. In *Proceedings of the 2014 ACM Conference on SIGCOMM*, SIGCOMM '14, page 187–198, New York, NY, USA. Association for Computing Machinery.

Huber, P. J. and Ronchetti, E. M. (2011). *Robust statistics*. John Wiley & Sons.

ITU (1990). ITU-T Recommendation G.726: 40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM). Technical Report G.726, International Telecommunication Union, Geneva, Switzerland.

ITU (1996). ITU-T Recommendation P.800 : Méthodes d'évaluation subjective de la qualité de transmission. Technical Report P.800, International Telecommunication Union, Geneva, Switzerland.

ITU (2001). ITU-T Recommendation P.862: Perceptual Evaluation of Speech Quality (PESQ), an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs. Technical Report P.862, International Telecommunication Union, Geneva, Switzerland.

ITU (2003). ITU-T Recommendation G.722.2: Wideband coding of speech at around 16 kbit/s using Adaptive Multi-Rate Wideband (AMR-WB). Technical Report G.722.2, International Telecommunication Union, Geneva, Switzerland.

ITU (2008a). ITU-T Recommendation P.565.1: Single-ended method for objective speech quality assessment in narrow-band telephony applications. Technical Report P.565.1, International Telecommunication Union, Geneva, Switzerland.

ITU (2008b). ITU-T Recommendation P.910: Subjective video quality assessment methods for multimedia applications. Technical Report P.910, International Telecommunication Union, Geneva, Switzerland.

ITU (2011). ITU-T Recommendation P.863: Perceptual objective listening quality assessment (POLQA). Technical Report P.863, International Telecommunication Union, Geneva, Switzerland.

ITU (2012). ITU-T Recommendation P.1202: Parametric non-intrusive bitstream assessment of video media streaming quality. Technical Report P.1202, International Telecommunication Union, Geneva, Switzerland.

ITU (2015). ITU-T Recommendation G.107: The E-model, a computational model for use in transmission planning. Technical Report G.107, International Telecommunication Union, Geneva, Switzerland.

ITU (2017). ITU-T Recommendation P.1203: Parametric bitstream-based quality assessment of progressive download and adaptive audiovisual streaming services over reliable transport. Technical Report P.1203, International Telecommunication Union, Geneva, Switzerland.

Ivanescu, A., Staicu, A.-M., Scheipl, F., and Greven, S. (2015). Penalized function-on-function regression. *Computational Statistics*, 30(2):539–568.

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

Jacques, J. and Samardzic, S. (2022). Analyzing cycling sensors data through ordinal logistic regression with functional covariates. *Journal of the Royal Statistical Society*, 71(4):969–986.

Jacques, J. and Tamo Tchomgui, J. S. (2024). Mixture of function-on-funtion regression models. In *Royal Statistical Society 2024 International Conference*, Brighton (UK), United Kingdom.

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

Jank, W. and Shmueli, G. (2006). Functional Data Analysis in Electronic Commerce Research. *Statistical Science*, 21(2):155–166.

Jiang, W. and Tanner, M. A. ("1999"). Hierarchical mixtures-of-experts for exponential family regression models: Approximation and maximum likelihood estimation. *"Annals of Statistics"*, "27"("3"):"987–1011".

Jordan, M. and Jacobs, R. (1994). Hierarchical mixtures of experts and the. *Neural computation*, 6:181–.

Kalogridis, I. and Van Aelst, S. (2019). Robust functional regression based on principal components. *Journal of Multivariate Analysis*, 173:393–415.

Koenker, R. and Bassett, G. (1978). Regression quantiles. *Econometrica*, 46(1):33–50.

Kokoszka, P. and Reimherr, M. (2017). *Introduction to Functional Data Analysis*. Chapman and Hall/CRC.

Lei, J., G'Sell, M. G., Rinaldo, A., Tibshirani, R. J., and Wasserman, L. A. (2016). Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113:1094 – 1111.

Leurgans, S. E., Moyeed, R. A., and Silverman, B. W. (1993). Canonical correlation analysis when the data are curves. *Journal of the Royal Statistical Society*, 55(3):725 – 740.

Li, Y. and Ruppert, D. (2008). On the asymptotics of penalized splines. *Biometrika*, 95(2):415–436.

Li, Z. (2020). The Waterloo Streaming Quality-of-Experience Database-IV.

Lin, C.-H., Shieh, C.-K., Hwang, W.-S., and Liu, D.-Y. (2021). Throughput-based mapping algorithm for video streaming over ieee 802.11 e wlan. *Wireless Communications and Mobile Computing*, 2021:1–14.

Lindsay, B. (1995). *Mixture Models: Theory, Geometry, and Applications*. NSF-CBMS regional conference series in probability and statistics. Institute of Mathematical Statistics.

Lindstrom, M. J. and Bates, D. M. (1988). Newton—raphson and em algorithms for linear mixed-effects models for repeated-measures data. *Journal of the American Statistical Association*, 83(404):1014–1022.

Liu, Y., Li, M., and Morris, J. (2020). Function-on-scalar quantile regression with application to mass spectrometry proteomics data. *Annals of Applied Statistics*, 14(2):521–541.

Louppe, G., Wehenkel, L., Sutera, A., and Geurts, P. (2013). Understanding variable importances in forests of randomized trees. In Burges, C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Luo, R., Qi, X., and Wang, Y. (2016). Functional wavelet regression for linear function-on-function models. *Electronic Journal of Statistics*, 10(2):3179 – 3216.

Machado, V. A., Silva, C. N., Oliveira, R. S., Melo, A. M., Silva, M., Francês, C. R., Costa, J. C., Vijaykumar, N. L., and Hirata, C. M. (2011). A new proposal to provide estimation of qos and qoe over wimax networks: An approach based on computational intelligence and discrete-event simulation. In *2011 IEEE Third Latin-American Conference on Communications*, pages 1–6. IEEE.

Malfait, N. and Ramsay, J. O. (2003). The historical functional linear model. *The Canadian Journal of Statistics*, 31(2):115 – 128.

Marx, B. D. and Eilers, P. H. C. (1999). Generalized linear regression on sampled signals and curves: A p-spline approach. *Technometrics*, 41(1):1–13.

McLachlan, G. and Krishnan, T. (2007). *The EM Algorithm and Extensions*. Wiley Series in Probability and Statistics. Wiley.

McLachlan, G. J. and Peel, D. (2000). *Finite mixture models*, volume 299 of *Probability and Statistics – Applied Probability and Statistics Section*. Wiley, New York.

Meyer, M. J., Coull, B. A., Versace, F., Cinciripini, P., and Morris, J. S. (2015). Bayesian Function-on-Function Regression for Multilevel Functional Data. *Biometrics*, 71(3):563–574.

Miao, Z. and Wang, L. (2024). Robust estimation for function-on-scalar regression models. *Journal of Statistical Computation and Simulation*, 94(5):1035–1055.

Minnier, J., Tian, L., and Cai, T. (2011). A perturbation method for inference on regularized regression estimates. *Journal of the American Statistical Association*, 106(496):1371–1382.

Mittag, G. (2022). *Deep learning based speech quality prediction*. Springer.

Mittag, G., Naderi, B., Gopal, V., and Cutler, R. (2023). LSTM-Based Video Quality Prediction Accounting for Temporal Distortions in Videoconferencing Calls. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Mittal, A., Moorthy, A. K., and Bovik, A. C. (2012). No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, 21:4695–4708.

Morris, J. (2014). Functional regression. *Annual Review of Statistics and Its Application*, 2.

Mousavi, S. and Sørensen, H. (2018). Functional logistic regression: a comparison of three methods. *Journal of Statistical Computation and Simulation*, 88:1–19.

Muelas, D., López de Vergara Méndez, J., and Berrendero, J. (2015). Functional data analysis: A step forward in network management. In *Functional Data Analysis: A step forward in Network Management*, pages 882–885.

Neal, R. M. and Hinton, G. E. (1998). A view of the em algorithm that justifies incremental, sparse, and other variants. In *Learning in graphical models*, pages 355–368. Springer.

Nguyen, H. and Chamroukhi, F. (2018). Practical and theoretical aspects of mixture-of-experts modeling: An overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8:e1246.

Peng, F., Jacobs, R. A., and Tanner, M. A. (1996). Bayesian inference in mixtures-of-experts and hierarchical mixtures-of-experts models with an application to speech recognition. *Journal of the American Statistical Association*, 91(435):953–960.

Pinson, M. and Wolf, S. (2004). A new standardized method for objectively measuring video quality. *IEEE Transactions on Broadcasting*, 50(3):312–322.

Rahman, W. u., Hossain, M. D., and Huh, E.-N. (2021). Fuzzy-based quality adaptation algorithm for improving qoe from mpeg-dash video. *Applied Sciences*, 11(11).

Ramsay, J. and Silverman, B. W. (2005). *Functional Data Analysis*. Springer Series in Statistics. Springer-Verlag, New York, 2 edition.

Ramsay, J. O., Hooker, G., and Graves, S. (2009). *Functional Data Analysis with R and MATLAB*. Springer Publishing Company, Incorporated, 1st edition.

Ratcliffe, S. J., Leader, L. R., and Heller, G. Z. (2002). Functional data analysis with application to periodically stimulated foetal heart rate data. i: Functional regression. *Statistics in Medicine*, 21(8):1103–1114.

Reiss, P. T., Goldsmith, J., Shang, H. L., and Ogden, R. T. (2017). Methods for scalar-on-function regression. *International Statistical Review*, 85(2):228–249.

Reiss, P. T., Huang, L., and Mennes, M. (2010). Fast function-on-scalar regression with penalized basis expansions. *The international journal of biostatistics*, 6(1).

Reiss, P. T. and Ogden, R. T. (2007). Functional principal component regression and functional partial least squares. *Journal of the American Statistical Association*, 102(479):984–996.

Riiser, H., Vigmostad, P., Griwodz, C., and Halvorsen, P. (2013). Commute path bandwidth traces from 3g networks: Analysis and applications. In *Proceedings of the 4th ACM Multimedia Systems Conference*, pages 114–118.

Robitza, W., Göring, S., Raake, A., Lindegren, D., Heikkilä, G., Gustafsson, J., List, P., Feiten, B., Wüstenhagen, U., Garcia, M.-N., Yamagishi, K., and Broom, S. (2018). Http adaptive streaming qoe estimation with itu-t rec. p. 1203: open databases and software. In *Proceedings of the 9th ACM Multimedia Systems Conference*, MMSys '18, page 466–471, New York, NY, USA. Association for Computing Machinery.

Rodriguez, D. Z., Rosa, R. L., and Bressan, G. (2013). Predicting the quality level of a voip communication through intelligent learning techniques.

Romano, Y., Patterson, E., and Candes, E. (2019). Conformalized quantile regression. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Ruppert, D. (2002). Selecting the number of knots for penalized splines. *Journal of Computational and Graphical Statistics*, 11(4):735–757.

Saricam, S., Beyaztas, U., Asikgil, B., and Shang, H. L. (2022). On partial least-squares estimation in scalar-on-function regression models. *Journal of Chemometrics*, 36(12):e3452.

Scheipl, F. and Greven, S. (2016). Identifiability in penalized function-on-function regression models. *Electronic Journal of Statistics*, 10(1):495 – 526.

Schwarz, G. (1978). Estimating the Dimension of a Model. *The Annals of Statistics*, 6(2):461 – 464.

Schwarzmann, S., Cassales Marquezan, C., Bosk, M., Liu, H., Trivisonno, R., and Zinner, T. (2019). Estimating video streaming qoe in the 5g architecture using machine learning. In *Proceedings of the 4th Internet-QoE Workshop on QoE-based Analysis and Management of Data Communication Networks*, pages 7–12.

Schwarzmann, S., Marquezan, C. C., Trivisonno, R., Nakajima, S., Barriac, V., and Zinner, T. (2022). Ml-based qoe estimation in 5g networks using different regression techniques. *IEEE Transactions on Network and Service Management*, 19(3):3516–3532.

Seshadrinathan, K. and Bovik, A. C. (2010). Motion tuned spatio-temporal quality assessment of natural videos. *Image Processing, IEEE Transactions on*, 19:335 – 350.

Silverman, B. W. (1996). Smoothed functional principal components analysis by choice of norm. *The Annals of Statistics*, 24(1):1 – 24.

Stewart, G. W. (1990). Perturbation theory and least squares with errors in the variables. *Contemporary Mathematics*, 112:171–181.

Sun, X., Du, P., Wang, X., and Ma, P. (2018). Optimal penalized function-on-function regression under a reproducing kernel hilbert space framework. *Journal of the American Statistical Association*, 113(524):1601–1611.

Tamo Tchomgui, J. S. (2023). Penffr git repository, https://github.com/Stevens05/PenFFR.

Tamo Tchomgui, J. S., Barriac, V., Fraysse, G., Jacques, J., and Chrétien, S. (2024a). Functional Linear Regression for the prediction of streaming video QoE. *20th International Conference on Network and Service Management*.

Tamo Tchomgui, J. S., Jacques, J., Barriac, V., Fraysse, G., and Chrétien, S. (2023a). A Penalized Spline Estimator for Functional Linear Regression with Functional Response. working paper or preprint.

Tamo Tchomgui, J. S., Jacques, J., Barriac, V., Fraysse, G., and Chrétien, S. (2023b). A penalized spline estimator for functional linear regression with functional response. *in proceeding draft version*.

Tamo Tchomgui, J. S., Jacques, J., Chrétien, S., Fraysse, G., and Barriac, V. (2022a). Function-on-Function Mixture of Experts Regression Models. In *15th International Conference of the ERCIM WG on Computational and Methodological Statistics. 16th International Conference on Computational and Financial Econometrics*, London, United Kingdom.

Tamo Tchomgui, J. S., Jacques, J., Chrétien, S., Fraysse, G., and Barriac, V. (2022b). Prédiction de la Qualité d'Expérience dans les Réseaux Mobiles : Cas de la VoIP. In *JDS'22 53es journées de la Statistique de la Société Française de Statistique (SFdS)*, Lyon, France.

Tamo Tchomgui, J. S., Jacques, J., Chrétien, S., Fraysse, G., and Barriac, V. (2023c). Modèle de mélanges d'experts pour données fonctionnelles. In *54es Journées de la Statistique de la SFdS*, Bruxelles, Belgium. Société Française de Statistique (SFdS).

Tamo Tchomgui, J. S., Jacques, J., Chrétien, S., Fraysse, G., and Barriac, V. (2024b). Function-on-Function Mixture-of-Experts Regression. In *fda-lille : Functional Data Analysis Workshop*, Lille, France.

Tamo Tchomgui, J. S., Jacques, J., Fraysse, G., Barriac, V., and Chrétien, S. (2022c). Quality of Experience's Prediction in Mobile Networks : The case of VoIP service. Spring school "Statlearn: challenging problems in statistical learning". Poster.

Tamo Tchomgui, J. S., Jacques, J., Fraysse, G., Barriac, V., and Chretien, S. (2024c). A mixture of experts regression model for functional response with functional covariates. *Statistics and Computing*, 34(5):154.

Tasaka, S. and Watanabe, Y. (2007). Real-time estimation of user-level qos in audio-video ip transmission by using temporal and spatial quality. In *IEEE GLOBECOM 2007 - IEEE Global Telecommunications Conference*, pages 2661–2666.

Tibshirani, R. J., Foygel Barber, R., Candes, E., and Ramdas, A. (2019). Conformal prediction under covariate shift. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Titterington, D., Smith, A., and Makov, U. (1985). *Statistical Analysis of Finite Mixture Distributions*. Wiley, New York.

Ul Mustafa, R., Esteve Rothenberg, C., and Barakat, C. (2022). Youtube goes 5g: Benchmarking youtube in 4g vs 5g.

Ullah, S. and Finch, C. F. (2013). Applications of functional data analysis: A systematic review. *BMC medical research methodology*, 13:1–12.

Varga, A. (2001). The omnet++ discrete event simulation system. *Proc. ESM'2001*, 9.

Vovk, V., Gammerman, A., and Shafer, G. (2005). *Algorithmic Learning in a Random World*. Springer-Verlag, Berlin, Heidelberg.

Wang, J.-L., Chiou, J.-M., and Müller, H.-G. (2016). Functional data analysis. *Annual Review of Statistics and its application*, 3(1):257–295.

Wang, Z., Lu, L., and Bovik, A. C. (2004). Video quality assessment based on structural distortion measurement. *Signal Processing: Image Communication*, 19(2):121–132.

Wang, Z., Simoncelli, E. P., and Bovik, A. C. (2003). Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pages 1398–1402. Ieee.

Wood, S. N. (2006). On confidence intervals for generalized additive models based on penalized regression splines. *Australian & New Zealand Journal of Statistics*, 48(4):445–464.

Yao, F., Müller, H.-G., and Wang, J.-L. (2005a). Functional linear regression analysis for longitudinal data1. *The Annals of Statistics*, 33(6):2873–2903.

Yao, F., Müller, H.-G., and Wang, J.-L. (2005b). Functional data analysis for sparse longitudinal data. *Journal of the American Statistical Association*, 100(470):577–590.

Young, D. and Hunter, D. (2010). Mixtures of regressions with predictor-dependent mixing proportions. *Computational Statistics & Data Analysis*, 54(10):2253–2266.

Zhang, H., Dong, L., Gao, G., Hu, H., Wen, Y., and Guan, K. (2020). Deepqoe: A multimodal learning framework for video quality of experience (qoe) prediction. *IEEE Transactions on Multimedia*, PP:1–1.

Zhao, Y., Ogden, R., and Reiss, P. (2012). Wavelet-based lasso in functional linear regression. *Journal of computational and graphical statistics : a joint publication of American Statistical Association, Institute of Mathematical Statistics, Interface Foundation of North America*, 21:600–617.

Appendix A

Function-on-Function Linear Regression

A.1 Simulation parameters

The values of the chosen constants is drawn from uniform law between -5 and 5. The values is given by: $\rho_0 = 0.439$, $\rho_1^1 =$ -3.562, $\rho_1^2 =$ -1.058, $\rho_1^3 =$ -2.955, $\rho_1^4 =$ -0.585, $\rho_1^5 =$ -0.298, $\rho_2^1 = 0.228$, $\rho_2^2 = 2.641$, $\rho_2^3 = 4.462$, $\rho_2^4 = 2.757$ and $\rho_2^5 = 2.283$.

A.2 Mixed model estimator

We first rewrite the model in the form :

$$\mathrm{Y} = \mathrm{R}^\top b + \varepsilon^*, \tag{A.2.1}$$

with $\varepsilon^* = \mathrm{Z}\,\mathrm{U} + \eta$, from which we get $\mathrm{V} = \mathbb{V}\mathrm{ar}(\varepsilon^*) = \mathrm{Z}\,\Gamma\,\mathrm{Z}^\top + \sigma^2\mathrm{I}$. We aim to estimate the fixed effects b and the error variance V from the observed data. The most popular estimation methods for the parameters in Model (3.3.8) are maximum likelihood (ML) and restricted maximum likelihood (ReML) as described in Lindstrom and Bates (1988). The log-likelihood of the model is written as:

$$\mathcal{L}_{pen}(b,\,\mathrm{V}) \quad = \quad nm\log(2\pi) + \log|\mathrm{V}| + (\mathrm{Y} - \mathrm{R}^\top b)^\top \mathrm{V}^{-1}(\mathrm{Y} - \mathrm{R}^\top b) \;+\; b^\top(\lambda\mathrm{P})\; b \tag{A.2.2}$$

First order condition: $\dfrac{\partial}{\partial b}\Big(\mathcal{L}_{pen}(b,\,\mathrm{V})\Big) = 0.$

$$\begin{aligned}
\frac{\partial \mathcal{L}_{pen}}{\partial b} &= \frac{\partial}{\partial b}\left(\Big(\mathrm{Y}^\top - (\mathrm{R}^\top b)^\top\Big)\mathrm{V}^{-1}\Big(\mathrm{Y} - (\mathrm{R}^\top b)\Big) \;+\; b^\top(\lambda\mathrm{P})\; b\right) \\
&= \frac{\partial}{\partial b}\left(\Big(\mathrm{Y}^\top\mathrm{V}^{-1}\mathrm{Y} - \mathrm{Y}^\top\mathrm{V}^{-1}\mathrm{R}^\top b - (\mathrm{R}^\top b)^\top\mathrm{V}^{-1}\mathrm{Y} + (\mathrm{R}^\top b)^\top\mathrm{V}^{-1}\mathrm{R}^\top b\Big) \;+\; b^\top(\lambda\mathrm{P})\; b\right) \\
&= -\,(\mathrm{Y}^\top\mathrm{V}^{-1}\mathrm{R}^\top)^\top - \mathrm{R}\,\mathrm{V}^{-1}\mathrm{Y} + 2\,\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top b \;+\; 2\,(\lambda\mathrm{P})\; b \\
&= -\,2\,\mathrm{R}\,\mathrm{V}^{-1}\mathrm{Y} \;+\; 2\,(\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top \;+\; \lambda\mathrm{P})\,b.
\end{aligned}$$

and by equalizing to 0, i.e. $\dfrac{\partial \mathcal{L}_{pen}}{\partial b} = 0$, we get:

$$\widehat{b}(\mathrm{V}) \;=\; \big(\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top \;+\; \lambda \mathrm{P}\big)^{-1}\mathrm{R}\,\mathrm{V}^{-1}\mathrm{Y}. \tag{A.2.3}$$

By replacing b by its estimator in the likelihood expression, we get the profiled log-likelihood given by:

$$\begin{aligned}
\mathcal{L}_p(\mathrm{V}) &= -\frac{1}{2}\Bigg(N\log(2\pi) + \log|\mathrm{V}| + \Big(\mathrm{Y} - \mathrm{R}^\top\big(\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top\big)^{-1}\mathrm{R}\,\mathrm{V}^{-1}\mathrm{Y}\Big)^\top \mathrm{V}^{-1} \\
&\qquad \Big(\mathrm{Y} - \mathrm{R}^\top\big(\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top\big)^{-1}\mathrm{R}\,\mathrm{V}^{-1}\mathrm{Y}\Big)\Bigg) \\
&= -\frac{1}{2}\Bigg(N\log(2\pi) + \log|\mathrm{V}| + \Big(\mathrm{Y}^\top\mathrm{V}^{-1} - \mathrm{Y}^\top\mathrm{V}^{-1}\mathrm{R}^\top\big(\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top\big)^{-1}\mathrm{R}\,\mathrm{V}^{-1}\Big) \\
&\qquad \Big(\mathrm{Y} - \mathrm{R}^\top\big(\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top\big)^{-1}\mathrm{R}\,\mathrm{V}^{-1}\mathrm{Y}\Big)\Bigg) \\
&= -\frac{1}{2}\Bigg(N\log(2\pi) + \log|\mathrm{V}| + \mathrm{Y}^\top\mathrm{V}^{-1}\mathrm{Y} - \mathrm{Y}^\top\mathrm{V}^{-1}\mathrm{R}^\top\big(\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top\big)^{-1}\mathrm{R}\,\mathrm{V}^{-1}\mathrm{Y} - \\
&\qquad \mathrm{Y}^\top\mathrm{V}^{-1}\mathrm{R}^\top\big(\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top\big)^{-1}\mathrm{R}\,\mathrm{V}^{-1}\mathrm{Y} + \\
&\qquad \mathrm{Y}^\top\mathrm{V}^{-1}\mathrm{R}^\top\big(\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top\big)^{-1}\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top\big(\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top\big)^{-1}\mathrm{R}\,\mathrm{V}^{-1}\mathrm{Y}\Bigg) \\
&= -\frac{1}{2}\Bigg(N\log(2\pi) + \log|\mathrm{V}| + \mathrm{Y}^\top\mathrm{V}^{-1}\mathrm{Y} - \mathrm{Y}^\top\mathrm{V}^{-1}\mathrm{R}^\top\big(\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top\big)^{-1}\mathrm{R}\,\mathrm{V}^{-1}\mathrm{Y}\Bigg) \\
\mathcal{L}_p(\mathrm{V}) &= -\frac{1}{2}\Bigg(N\log(2\pi) + \log|\mathrm{V}| + \mathrm{Y}^\top\mathrm{V}^{-1}\Big(\mathrm{I} - \mathrm{R}^\top\big(\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top\big)^{-1}\mathrm{R}\,\mathrm{V}^{-1}\Big)\mathrm{Y}\Bigg).
\end{aligned}$$

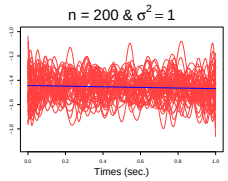
On the other hand, there holds $\mathrm{V} = \mathbb{V}\mathrm{ar}(\varepsilon^*) = \sigma_{\mathrm{U}}^2\,\mathrm{Z}\,\mathrm{Z}^\top + \sigma^2\mathrm{I}$, and thus, $\mathcal{L}_p(\mathrm{V}) = \mathcal{L}_p(\sigma_{\mathrm{U}}^2, \sigma^2)$. It is obviously not easy to derive this likelihood which no longer depends on b. Moreover, maximizing this last function gives the MLE which is nevertheless biased. For these reasons, and in order to account for the degrees of freedom of the fixed effects in the model, we propose to use the Restricted Maximum Likelihood (ReML) which reads:

$$\mathcal{L}_R(\mathrm{V}) \;=\; \mathcal{L}_p(\mathrm{V}) - \frac{1}{2}\,\log|\mathrm{R}\,\mathrm{V}^{-1}\mathrm{R}^\top| \tag{A.2.4}$$

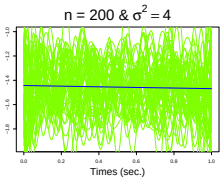
From a numerical viewpoint, we obtain the estimator $\hat{\mathrm{V}}$ of the variance V by maximizing this last likelihood from which we finally deduce the value of $\hat{\mathrm{U}}$ given by:

$$\begin{cases} \widehat{b} &= (\mathrm{R}^\top\hat{\mathrm{V}}^{-1}\mathrm{R} \;+\; \lambda\mathrm{P})^{-1}\mathrm{R}^\top\hat{\mathrm{V}}^{-1}\mathrm{Y}, \\ \widehat{\mathrm{U}} &= \sigma^2\mathrm{Z}^\top\hat{\mathrm{V}}^{-1}(\mathrm{Y} - \mathrm{R}^\top\widehat{b}). \end{cases} \tag{A.2.5}$$

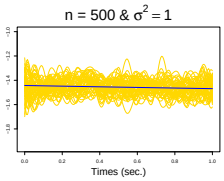
A.3 Parameter representation on simulated data



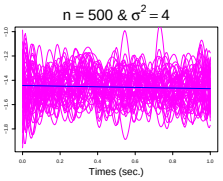
(a) $\beta_0(t)$, case 1



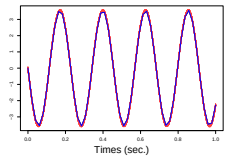
(b) $\beta_0(t)$, case 2



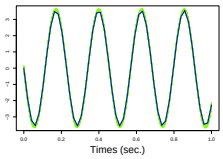
(c) $\beta_0(t)$, case 3



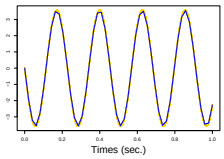
(d) $\beta_0(t)$, case 4



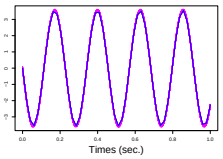
(e) $\beta_1(t)$, case 1



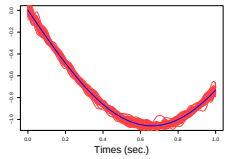
(f) $\beta_1(t)$, case 2



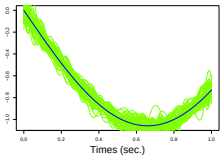
(g) $\beta_1(t)$, case 3



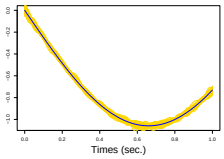
(h) $\beta_1(t)$, case 4



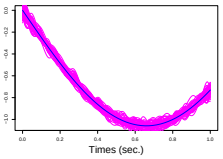
(i) $\beta_2(t)$, case 1



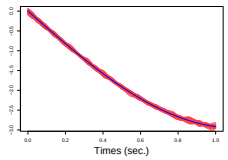
(j) $\beta_2(t)$, case 2



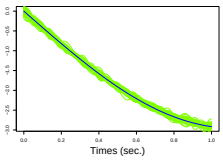
(k) $\beta_2(t)$, case 3



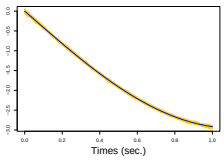
(l) $\beta_2(t)$, case 4



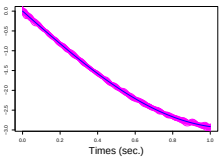
(m) $\beta_3(t)$, case 1



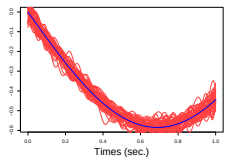
(n) $\beta_3(t)$, case 2



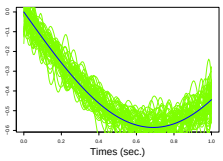
(o) $\beta_3(t)$, case 3



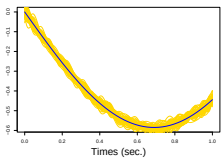
(p) $\beta_3(t)$, case 4



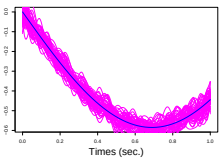
(q) $\beta_4(t)$, case 1



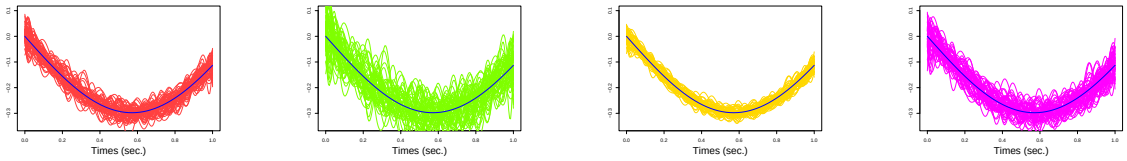
(r) $\beta_4(t)$, case 2



(s) $\beta_4(t)$, case 3



(t) $\beta_4(t)$, case 4

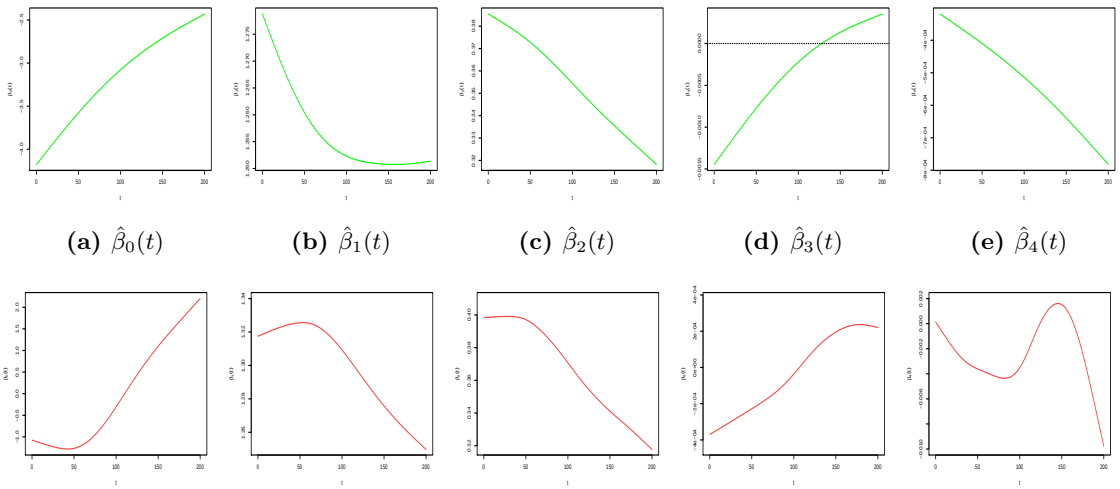


(u) $\beta_5(t)$, case 1 **(v)** $\beta_5(t)$, case 2 **(w)** $\beta_5(t)$, case 3 **(x)** $\beta_5(t)$, case 4

Figure A.1: Estimated and actual parameters for the concurrent model over the 4 scenarios of simulation.

In each scenario, we estimate the functional parameters with cubic B-splines basis, regular knots over the grid and $L_{\beta^l} = 50$ basis functions. The parameters we obtain with our model are close to the true parameters. However, we note that estimation of $\beta_0(t)$ is noised by the two large number of basis functions considered. This confirms the previously mentioned concerns about interpretability (smoothness) of the estimated parameters without regularization. Figure A.1 also confirms that estimation accuracy increases with the number of observations.

A.4 Parameters estimation for concurrent models on Hawaii Ocean Data

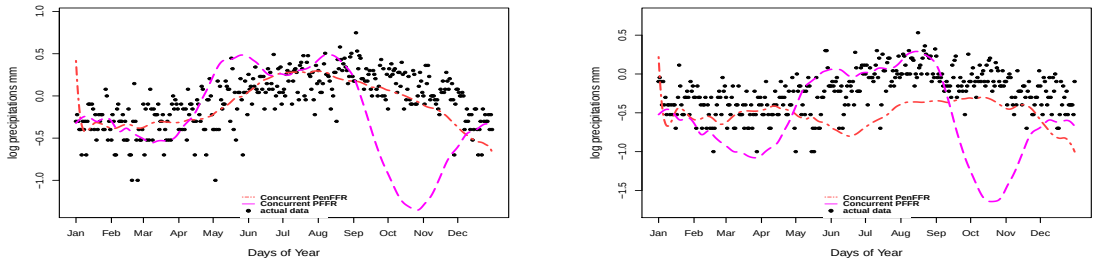


(a) $\hat{\beta}_0(t)$ **(b)** $\hat{\beta}_1(t)$ **(c)** $\hat{\beta}_2(t)$ **(d)** $\hat{\beta}_3(t)$ **(e)** $\hat{\beta}_4(t)$

(f) $\hat{\beta}_0(t)$ **(g)** $\hat{\beta}_1(t)$ **(h)** $\hat{\beta}_2(t)$ **(i)** $\hat{\beta}_3(t)$ **(j)** $\hat{\beta}_4(t)$

Figure A.2: Estimates $\hat{\beta}_j(t)$, $0 \leq j \leq 4$ for the two methods (pffr and PenFFR) on concurrent model. The first row shows the estimation provided by our PenFFR method. The second row shows the estimation provided by the pffr method.

A.5 Prediction on concurrent models for Canadian Weather data



(a) Churchill station **(b)** Inuvik station

Figure A.3: Prediction on two randomly chosen stations. For each figure, the black points are the actual data, the red two-dashed line is the prediction given by our concurrent PenFFR and the magenta dashed line is the prediction given by the concurrent PFFR method.

Appendix B

Function-on-Function Linear Mixture-of-Experts

B.1 EM for the FFMoE

Given the complete data log-likelihood and the parameters at current iteration l, we define the Q function for the EM algorithm defined by:

$$\mathrm{Q}(\Psi^{(l+1)} \,|\, \Psi^{(l)}) \quad = \quad \mathbb{E}(\mathcal{L}_c(\Psi^{(l+1)}; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) \,|\, \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j};\ \Psi^{(l)})$$

Now we are going to describe the EM algorithm for maximizing (3.3.11):

- **E-step:**
  At this step, we compute the conditional expectation of the log-likelihood given the observed data and the current parameter (at iteration l) estimation $\Psi^{(l)}$. This is equivalent to update the posterior probabilities $p_{ik}^{(l)}$ that the curves $\mathrm{x}_i(t)$ belongs to the k^{th} component of the mixture under the current model:

  $$p_{ik}^{(l)} \quad = \quad \mathbb{E}(z_{ik} \,|\, \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}; \Psi^{(l)}) = \mathbb{P}(z_{ik} = 1 \,|\, \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}; \Psi^{(l)}).$$

  Using Bayes' theorem, the conditional probability $p_{ik}^{(l)}$ can be expressed as:

  $$\begin{aligned} p_{ik}^{(l)} &= \frac{\mathbb{P}(z_{ik} = 1)\, \mathbb{P}(\{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j};\ \Psi^{(l)} \,|\, z_{ik} = 1)}{\mathbb{P}(\{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j};\ \Psi^{(l)})} \\ &= \frac{\mathbb{P}(z_{ik} = 1)\, \mathbb{P}(\{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j};\ \Psi^{(l)} \,|\, z_{ik} = 1)}{\sum_{u=1}^{\mathrm{K}} \mathbb{P}(z_{iu} = 1) \mathbb{P}(\{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j};\ \Psi^{(l)} \,|\, z_{iu} = 1)} \\ p_{ik}^{(l)} &= \frac{\pi_k(\mathrm{x}_i(t), \alpha_k^{(l)}(t))\, \Phi_m(\mathbf{y}_i\,;\, b_k^{\top(l)}\mathrm{R}_i, \mathrm{V}_{k,i}^{(l)})}{\sum_{u=1}^{\mathrm{K}} \pi_u(\mathrm{x}_i(t), \alpha_u^{(l)}(t))\, \Phi_m(\mathbf{y}_i\,;\, b_u^{\top(l)}\mathrm{R}_i, \mathrm{V}_{u,i}^{(l)})} \end{aligned} \tag{B.1.1}$$

- **M-step:**
  Given the previous posterior probability and the observed data, this step updates the current

parameters $\Psi^{(l)}$ by maximizing the complete (data) log-likelihood, that is $\Psi^{(l+1)}$:

$$\begin{aligned}
\mathrm{Q}(\Psi^{(l+1)} \,|\, \Psi^{(l)}) &= \mathbb{E}(\mathcal{L}_c(\Psi^{(l+1)}; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) \,|\, \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j};\ \Psi^{(l)}) \\
&= \mathbb{E}(\sum_{i=1}^{n}\sum_{k=1}^{\mathrm{K}} z_{ik}\ \log(\frac{\exp({a_k^{(l+1)}}^\top r_i)}{1+\sum_{u=1}^{\mathrm{K}-1}\exp({a_u^{(l+1)}}^\top r_i)} \cdot \frac{1}{\sqrt{(2\pi)^m|\mathrm{V}_{k,i}^{(l+1)}|}} \\
&\quad \exp(-\frac{1}{2}(\mathbf{y}_i - {b_k^{(l+1)}}^\top \mathrm{R}_i)^\top \mathrm{V}_{k,i}^{-1^{(l+1)}}(\mathbf{y}_i - {b_k^{(l+1)}}^\top \mathrm{R}_i))) \\
&\quad \Big| \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j};\ \Psi^{(l)}) \\
&= \sum_{i=1}^{n}\sum_{k=1}^{\mathrm{K}} \mathbb{E}(z_{ik} \,|\, \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}; \Psi^{(l)})\ \log(\frac{\exp({a_k^{(l+1)}}^\top r_i)}{1+\sum_{u=1}^{\mathrm{K}-1}\exp({a_u^{(l+1)}}^\top r_i)} \\
&\quad \frac{1}{\sqrt{(2\pi)^m|\mathrm{V}_{k,i}^{(l+1)}|}}\exp(-\frac{1}{2}(\mathbf{y}_i - {b_k^{(l+1)}}^\top \mathrm{R}_i)^\top \mathrm{V}_{k,i}^{-1^{(l+1)}}(\mathbf{y}_i - {b_k^{(l+1)}}^\top \mathrm{R}_i))))
\end{aligned}$$

$$\begin{aligned}
= &\underbrace{\sum_{i=1}^{n}\sum_{k=1}^{\mathrm{K}} p_{ik}^{(l)}\ \log(\frac{\exp({a_k^{(l+1)}}^\top r_i)}{1+\sum_{u=1}^{\mathrm{K}-1}\exp({a_u^{(l+1)}}^\top r_i)})}_{\mathrm{Q}_1(a_k^{(l+1)} \,|\, \Psi^{(l)})} \quad + \\
&\underbrace{\sum_{i=1}^{n}\sum_{k=1}^{\mathrm{K}} p_{ik}^{(l)}\ \log(\frac{1}{\sqrt{(2\pi)^m|\mathrm{V}_{k,i}^{(l+1)}|}}\exp(-\frac{1}{2}(\mathbf{y}_i - {b_k^{(l+1)}}^\top \mathrm{R}_i)^\top \mathrm{V}_{k,i}^{-1^{(l+1)}}(\mathbf{y}_i - {b_k^{(l+1)}}^\top \mathrm{R}_i)))}_{\mathrm{Q}_2(b_k^{(l+1)},\ \mathrm{V}_k^{(l+1)} \,|\, \Psi^{(l)})}
\end{aligned}$$

$$\mathrm{Q}(\Psi^{(l+1)} \,|\, \Psi^{(l)}) \quad = \quad \mathrm{Q}_1(a_k^{(l+1)} \,|\, \Psi^{(l)}) \ +\ \mathrm{Q}_2(b_k^{(l+1)},\ \mathrm{V}_k^{(l+1)} \,|\, \Psi^{(l)}).$$

The global maximization problem is split onto two separate maximization problems: the updating of gated network parameters via the maximization of the function $\mathrm{Q}_1(a_k^{(l+1)} \,|\, \Psi^{(l)})$ and the updating of experts parameters via the maximization of the function $\mathrm{Q}_2(b_k^{(l+1)},\ \sigma_k^{2\ (l+1)} \,|\, \Psi^{(l)})$. It obsvious to recognise in each of these two expressions the likelihood of the multinomial logistic model $\mathrm{Q}_1(.)$ and the linear gaussian model $\mathrm{Q}_2(.)$ for which we know how to calculate (at least numerically through Newton-Raphson method for e.g) MLEs.

B.2 EM for PenFFMoE

- **E-step:**
  Same as in non penalize case
- **M-step:**
  Given the previous posterior probability and the observed data, this step updates the current

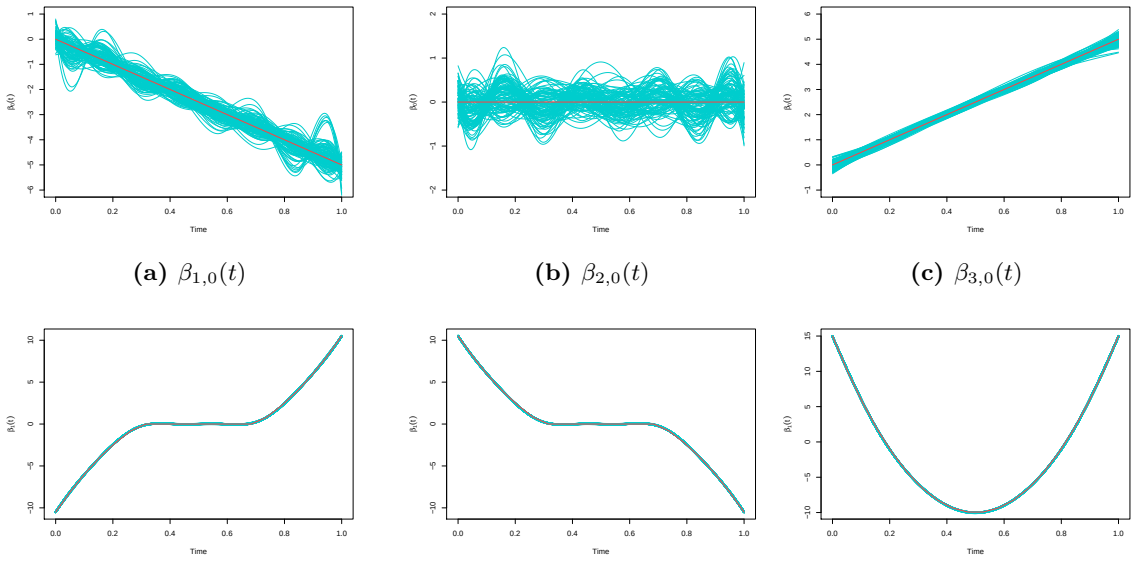
parameters $\Psi^{(l)}$ by maximizing the penalized complete (data) log-likelihood, that is $\Psi^{(l+1)}$. We define:

$$
\begin{aligned}
\mathrm{Q}_{pen}(\Psi^{(l+1)} \mid \Psi^{(l)}) &= \mathbb{E}(\mathcal{L}^c_{pen}(\Psi^{(l+1)}; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) \Big| \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j};\ \Psi^{(l)}) \\
&= \mathbb{E}(\sum_{i=1}^{n}\sum_{k=1}^{\mathrm{K}} z_{ik}\ \log(\frac{\exp({a_k^{(l+1)}}^\top r_i)}{1+\sum_{u=1}^{\mathrm{K}-1}\exp({a_u^{(l+1)}}^\top r_i)} \cdot \frac{1}{\sqrt{(2\pi)^m|\mathrm{V}_{k,i}^{(l+1)}|}} \\
&\quad \exp(-\frac{1}{2}(\mathbf{y}_i - {b_k^{(l+1)}}^\top \mathrm{R}_i)^\top \mathrm{V}_{k,i}^{-1^{(l+1)}}(\mathbf{y}_i - {b_k^{(l+1)}}^\top \mathrm{R}_i))) - \\
&\quad \sum_{k=1}^{\mathrm{K}} {b_k^{(l+1)}}^\top(\lambda_k \mathrm{P})\ b_k^{(l+1)} - \sum_{k=1}^{\mathrm{K}-1} {a_k^{(l+1)}}^\top(\gamma_k \mathrm{Q})\ a_k^{(l+1)} \\
&\quad \Big| \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j};\ \Psi^{(l)}) \\
&= \sum_{i=1}^{n}\sum_{k=1}^{\mathrm{K}} \mathbb{E}(z_{ik} \mid \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}; \Psi^{(l)})\ \log(\frac{\exp({a_k^{(l+1)}}^\top r_i)}{1+\sum_{u=1}^{\mathrm{K}-1}\exp({a_u^{(l+1)}}^\top r_i)} \\
&\quad \frac{1}{\sqrt{(2\pi)^m|\mathrm{V}_{k,i}^{(l+1)}|}}\exp(-\frac{1}{2}(\mathbf{y}_i - {b_k^{(l+1)}}^\top \mathrm{R}_i)^\top \mathrm{V}_{k,i}^{-1^{(l+1)}}(\mathbf{y}_i - {b_k^{(l+1)}}^\top \mathrm{R}_i)))) \\
&\quad -\sum_{k=1}^{\mathrm{K}} {b_k^{(l+1)}}^\top(\lambda_k \mathrm{P})\ b_k^{(l+1)} - \sum_{k=1}^{\mathrm{K}-1} {a_k^{(l+1)}}^\top(\gamma_k \mathrm{Q})\ a_k^{(l+1)}
\end{aligned}
$$

$$
\begin{aligned}
= \underbrace{\sum_{i=1}^{n}\sum_{k=1}^{\mathrm{K}} p_{ik}^{(l)}\ \log(\frac{\exp({a_k^{(l+1)}}^\top r_i)}{1+\sum_{u=1}^{\mathrm{K}-1}\exp({a_u^{(l+1)}}^\top r_i)})}_{\mathrm{Q}_1(a_k^{(l+1)} \mid \Psi^{(l)})} &- \sum_{k=1}^{\mathrm{K}-1} {a_k^{(l+1)}}^\top(\gamma_k \mathrm{Q})\ a_k^{(l+1)} - \\
&\sum_{k=1}^{\mathrm{K}} {b_k^{(l+1)}}^\top(\lambda_k \mathrm{P})\ b_k^{(l+1)} + \\
\underbrace{\sum_{i=1}^{n}\sum_{k=1}^{\mathrm{K}} p_{ik}^{(l)}\ \log(\frac{1}{\sqrt{(2\pi)^m|\mathrm{V}_{k,i}^{(l+1)}|}}\exp(-\frac{1}{2}(\mathbf{y}_i - {b_k^{(l+1)}}^\top \mathrm{R}_i)^\top \mathrm{V}_{k,i}^{-1^{(l+1)}}(\mathbf{y}_i - {b_k^{(l+1)}}^\top \mathrm{R}_i)))}_{\mathrm{Q}_2(b_k^{(l+1)},\ \mathrm{V}_k^{(l+1)} \mid \Psi^{(l)})}
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{Q}(\Psi^{(l+1)} \mid \Psi^{(l)}) &= \underbrace{\mathrm{Q}_1(a_k^{(l+1)} \mid \Psi^{(l)}) - \sum_{k=1}^{\mathrm{K}-1} {a_k^{(l+1)}}^\top(\gamma_k \mathrm{Q})\ a_k^{(l+1)}}_{} + \\
&\quad \underbrace{\mathrm{Q}_2(b_k^{(l+1)},\ \mathrm{V}_k^{(l+1)} \mid \Psi^{(l)}) - \sum_{k=1}^{\mathrm{K}} {b_k^{(l+1)}}^\top(\lambda_k \mathrm{P})\ b_k^{(l+1)}}_{} \\
&= \mathrm{Q}_{1,\,pen}(a_k^{(l+1)} \mid \Psi^{(l)})\ +\ \mathrm{Q}_{2,\,pen}(b_k^{(l+1)},\ \sigma_k^{2\ (l+1)} \mid \Psi^{(l)}).
\end{aligned}
$$

B.3 Parameters in simulation study

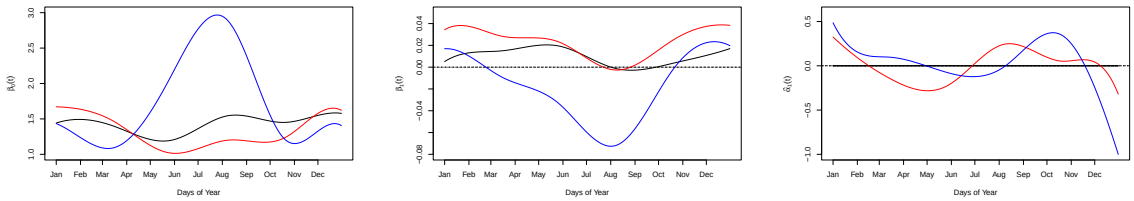


(a) $\beta_{1,0}(t)$ **(b)** $\beta_{2,0}(t)$ **(c)** $\beta_{3,0}(t)$

(d) $\beta_{1,1}(t)$ **(e)** $\beta_{2,1}(t)$ **(f)** $\beta_{3,1}(t)$

Figure B.1: Estimation of the regression coefficients for Scenario S3 with PenFFMoE. The red curves are the actual parameters, the cyan curves are the estimation.

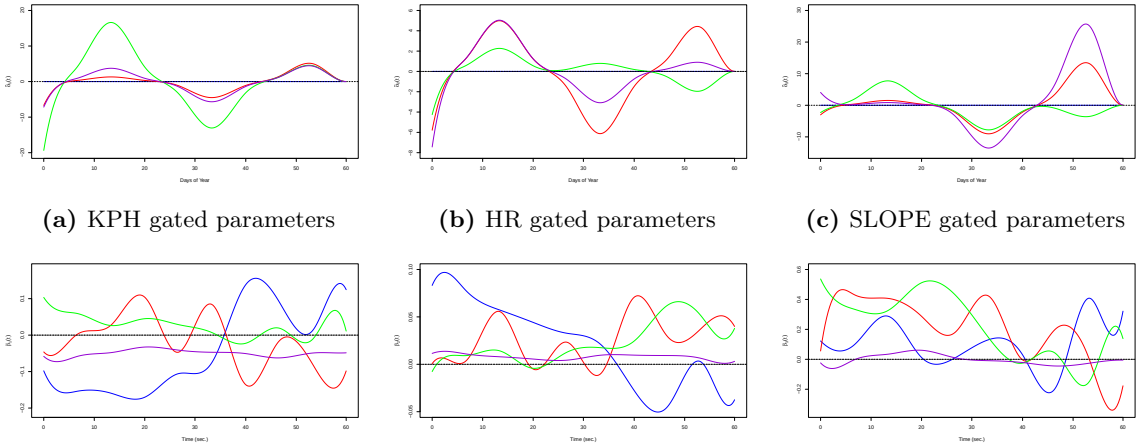
B.4 Estimators for Canadian weather data



(a) Intercepts **(b)** Expert parameter **(c)** gated parameters

Figure B.2: Functional coefficients and gated network parameters obtained by FFMoE on Canadian Weather data. Color depends on group membership.

B.5 Estimators for Cycling data



(a) KPH gated parameters **(b)** HR gated parameters **(c)** SLOPE gated parameters

(d) KPH's expert parameters **(e)** HR's expert parameters **(f)** SLOPE's exp. parameters

Figure B.3: Functional gated (first row) and expert (second row) parameters obtained by FFMoE on Cycling data. corresponding colors matched for the same cluster.

List of Figures

List of Tables

Résumé Long en Français

Contexte scientifique et industriel :
La croissance exponentielle du trafic Internet et le développement rapide des services dépendants dus à l'évolution continue des réseaux de télécommunications ont conduit à une demande sans cesse croissante des performances de qualité du réseau. Dans ce contexte, les opérateurs de réseau mobile (ORM) doivent se différencier de leurs concurrents en prêtant une attention particulière à garantir une expérience utilisateur optimale. Ce qui signifie passer de simples mesures "objectives" de la performance du réseau, ou Qualité de Service (QoS), aux mesures "subjectives" de l'expérience utilisateur, ou Qualité d'Expérience (QoE). La QoE, souvent exprimée en termes de score moyen d'opinion ou note MOS, est une valeur scalaire allant de 1 (pour très mauvais) à 5 (pour excellent), initialement résultant d'une moyenne des scores individuels dans un test subjectif de qualité formel, et aujourd'hui souvent calculée à l'aide de modèles objectifs utilisant des signaux audio ou vidéo comme entrées. Nous utiliserons largement cet acronyme MOS dans ce document chaque fois que la QoE sera abordée. Dans ce contexte, la capacité à prédire avec précision la MOS est devenue un défi pour les opérateurs de réseau. L'un des principaux défis dans le cadre de la prédiction de la QoE est le fait que, pour certains services, l'expérience utilisateur (c'est-à-dire la QoE en tant que variable de réponse) et les facteurs qui l'influencent (c'est-à-dire les paramètres réseau en tant que covariables) sont enregistrés au fil du temps (fonctions du temps), ce qui nécessite de nouvelles méthodes pour traiter ces données. C'est dans ce contexte que s'inscrit l'Analyse de Données Fonctionnelles (ADF) qui est devenue très populaire dans un nombre croissant d'applications industrielles, sociétales et médicales. L'ADF est une branche des statistiques qui traite des données pouvant être représentées sous forme de fonctions. Sa flexibilité à gérer des données complexes, multidimensionnelles et structurées la rend applicable à un large éventail de problèmes scientifiques et pratiques, fournissant des perspectives que les méthodes d'analyse de données traditionnelles ne parviennent pas toujours à combler. Parmi les applications récentes les plus remarquables, on trouve notamment la Santé et la médecine (surveillance de la santé des patients au fil du temps, données d'IRM), les sciences environnementales (tendances des températures ou des précipitations dans le temps), l'économie et la finance (évolution des actions ou des matières premières, modélisation du comportement des consommateurs dans le temps), les sciences du sport (analyse des performances des athlètes au fil du temps ou lors d'événements pour optimiser l'entraînement et les performances), la chimiométrie (analyse des données de spectroscopie pour identifier et quantifier des substances chimiques), la Génomique et la Bioinformatique (analyse des données d'expression génétique au fil du temps), ainsi que l'Analyse du Trafic et la Planification Urbaine. L'extension de la régression linéaire au cadre fonctionnel est devenue naturellement un domaine majeur de recherche en ADF. Cette thèse de doctorat vise à contribuer au domaine de l'ADF en proposant des solutions pour la prédiction de réponses fonctionnelles à partir de covariables fonc-

tionnelles en utilisant la régression linéaire, en relevant les principaux défis méthodologiques et en proposant des solutions innovantes. Les approches proposées s'appuient sur l'expansion en base de fonctions, des techniques de régression pénalisée et des méthodes adaptées aux données hétérogènes.

Modèles homogènes de prédiction de QoE :
Le problème mathématique posé est d'estimer la relation linéaire entre une réponse fonctionnelle et des covariables fonctionnelles à partir du n-échantillon:

$$\left\{ \mathrm{Y}_i(t),\ \mathrm{X}_i(t) = \left(\mathrm{X}_i^1(t), \ldots, \mathrm{X}_i^p(t)\right)^\top,\ t \in [0, \mathrm{T}],\ i = 1, \ldots, n \right\}$$

où on suppose que les p variables d'entrées et la variables de sortie $\mathrm{Y}(t)$ appartiennent à l'espace de Hilbert séparable des fonctions de carré intégrable $\mathrm{L}^2([0, \mathrm{T}])$.
Les modèles sur lesquelles nous nous focaliserons initialement définis par Ramsay and Silverman (2005) sont donnés par:

$$\mathrm{Y}_i(t) \quad = \quad \beta_0(t) + \sum_{l=1}^{p} \beta_l(t)\, \mathrm{X}_i^l(t) + \varepsilon_i(t) \quad = \quad \left(1,\, \mathrm{X}_i(t)\right)^\top \beta(t) + \varepsilon_i(t)\ , \tag{B.5.1}$$

$$\mathrm{Y}_i(t) \quad = \quad \gamma_0(t) + \sum_{l=1}^{p} \int_0^t \gamma_l(s,t)\, \mathrm{X}_i^l(s)\, \mathrm{d}s + \varepsilon_i(t) = \gamma_0(t) + \int_0^t \mathrm{X}_i(s)^\top \gamma(s,t)\, \mathrm{d}s + \varepsilon_i(t) \tag{B.5.2}$$

Où $\beta(t) = \left(\beta_0(t),\quad \beta_1(t),\quad \ldots,\quad \beta_p(t)\right)^\top$, $\gamma(s,t) = \left(\gamma_0(t),\quad \gamma_1(s,t),\quad \gamma_2(s,t),\quad \ldots,\quad \gamma_p(s,t)\right)^\top$ sont les paramètres à estimer et sont supposés de carré intégrable. $\varepsilon_i(t)$ l'erreur du modèle distribué selon une loi normale centrée et de variance σ_i^2.

Le modèle (B.5.1) appelé modèle concurrent suppose la fonction réponse à un instant t dépend des covariables au même instant t. Le second modèle ou modèle intégral (B.5.2) quant à lui prend en compte l'influence des covariables à partir de l'instant initial jusqu'à l'instant t pour expliquer la réponse à l'instant t.

Avant d'estimer les différents paramètres de ces modèles, nous allons au préalable reconstruire la nature fonctionnelle des données. En effet dans la pratique, les données observées ne sont pas des courbes. La donnée fonctionnelle se présente généralement sous forme vecteur correspondant aux observations (bruitées) de la fonction originelle à différents instants. Pour reconstruire cette fonction continue qui appartient généralement à un espace de dimension infinie, un moyen efficace est de l'approximer dans une base de fonctions comme combinaison linéaire infinie des coefficients et fonctions de la base de fonctions. L'avantage de cette méthode est qu'en tronquant la somme à un niveau finie q, on obtient une approximation de la fonction dans un espace de dimension fini. Ainsi, les p covariables $\mathrm{X}_i^l(t)$ peuvent s'exprimer sous la forme :

$$\mathrm{X}_i^l(t) \quad = \quad \sum_{j=1}^{q_{\mathrm{x}^l}} x_{ij}^l\, \mathrm{B}_j^l(t) = \mathrm{B}^l(t)^\top x_i^l \qquad \text{with } 1 \leq l \leq p. \tag{B.5.3}$$

Tout comme les covariables, pour pouvoir estimer les paramètres fonctionnels nous supposons également qu'ils se décomposent dans une base de fonctions. Ainsi,

Pour le modèle concurrent (B.5.1) :
les paramètres s'écrivent:

$$\beta_l(t) = \sum_{j=1}^{q_{\beta^l}} b_j^l \, \phi_j^l(t) = \phi^l(t)^\top \, b^l \qquad \text{avec } 0 \leq l \leq p. \tag{B.5.4}$$

En utilisant les expressions (B.5.3) et (B.5.4), les éléments du modèle (B.5.1) deviennent:

$$\beta(t) = \begin{pmatrix} \beta_0(t) \\ \beta_1(t) \\ \vdots \\ \beta_p(t) \end{pmatrix} = \begin{pmatrix} \phi^0(t)^\top b^0 \\ \phi^1(t)^\top b^1 \\ \vdots \\ \phi^p(t)^\top b^p \end{pmatrix} = \underbrace{\begin{pmatrix} \phi^0(t)^\top & 0 & \dots & 0 \\ 0 & \phi^1(t)^\top & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \phi^p(t)^\top \end{pmatrix}}_{(p+1,\, \sum_l q_{\beta^l})} \underbrace{\begin{pmatrix} b^0 \\ b^1 \\ \vdots \\ b^p \end{pmatrix}}_{\sum_l q_{\beta^l}} = \Phi(t)\, b\,,$$

et

$$\mathrm{X}_i(t) = \begin{pmatrix} 1 \\ \mathrm{X}_i^1(t) \\ \vdots \\ \mathrm{X}_i^p(t) \end{pmatrix} = \begin{pmatrix} 1 \\ \mathrm{B}^1(t)^\top x_i^1 \\ \vdots \\ \mathrm{B}^p(t)^\top x_i^p \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & \mathrm{B}^1(t)^\top & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathrm{B}^p(t)^\top \end{pmatrix}}_{(p+1,\, \sum_l q_{\mathrm{X}^l})} \underbrace{\begin{pmatrix} 1 \\ x_i^1 \\ \vdots \\ x_i^p \end{pmatrix}}_{\sum_l q_{\mathrm{X}^l}} = \mathrm{B}(t)\, x_i.$$

En intégrant ces expressions dans le modèle (B.5.1), on obtient :

$$\mathrm{Y}_i(t) \quad = \quad x_i^\top \mathrm{B}(t)^\top \Phi(t)\; b + \varepsilon_i(t) = \mathrm{R}_i(t)^\top b + \varepsilon_i(t) \tag{B.5.5}$$

avec $\mathrm{R}_i(t) = \Phi(t)^\top \mathrm{B}(t)\, x_i$ qui constituera les éléments de la matrice de design dont le vecteur b est le vecteur de paramètres à estimer.

Pour le modèle intégral (B.5.2) :
les paramètres bivariés s'expriment dans la base de fonctions sous la forme :

$$\gamma_l(t,s) = \sum_{j,k=1}^{q_{\gamma^l}} a_{jk}^l \, \mathrm{B}_{1j}^l(t)\, \mathrm{B}_{2k}^l(s), \tag{B.5.6}$$

où $\left\{\mathrm{B}_{1j}^l(t)\right\}_{1\leq j\leq q_{\gamma^l}}$ et $\left\{\mathrm{B}_{2j}^l(t)\right\}_{1\leq j\leq q_{\gamma^l}}$ sont les deux ensembles de fonctions de bases de fonctions pour chaque dimension et $(a_{jk}^l)_{1\leq j,k\leq q_{\gamma^l}}$ les paramètres inconnus à estimer. Les paramètres étant sous forme matricielle, on peut réecrire cette expression en utilisant une vecteur pour les paramètres sous la forme :

$$\gamma_l(t,s) \quad = \quad a^{l^\top} \; \mathbf{B}_1^l(t)\, \mathrm{B}_2^l(s) \tag{B.5.7}$$

avec

$$\begin{aligned} a^l &= \left(a_{11}^l \quad \dots \quad a_{1q_{\gamma^l}}^l \quad a_{21}^l \quad \dots \quad \dots \quad a_{q_{\gamma^l}1}^l \quad \dots \quad a_{q_{\gamma^l}q_{\gamma^l}}^l\right)^\top, \\ \mathbf{B}_1^l(t) &= \mathbf{diag}\left(\mathrm{B}_{11}^l(t), \dots, \mathrm{B}_{1q_{\gamma^l}}^l(t), \dots, \dots, \mathrm{B}_{11}^l(t), \dots, \mathrm{B}_{1q_{\gamma^l}}^l(t)\right), \\ \mathrm{B}_2^l(s) &= \left(\mathrm{B}_{21}^l(s) \quad \dots \quad \mathrm{B}_{21}^l(s) \quad \dots \quad \dots \quad \mathrm{B}_{2q_{\gamma^l}}^l(s) \quad \dots \quad \mathrm{B}_{2q_{\gamma^l}}^l(s)\right)^\top. \end{aligned}$$

La constante fonctionnelle $\gamma_0(t)$ étant univariée, elle peut s'écrire comme avec le modèle concurrent (B.5.4) sous la forme :

$$\gamma_0(t) = \sum_{j=1}^{q_{\gamma^0}} a_j^0 \, \mathrm{B}_j^0(t) = \mathrm{B}^0(t)^\top a^0.$$

En intégrant les expressions des covariables et des paramètres fonctionnels dans le modèle intégral (B.5.2), on obtient :

$$\begin{aligned} \mathrm{Y}_i(t) &= \gamma_0(t) + \sum_{l=1}^{p} \int_0^t x_i^{l\top} \mathrm{B}^l(s) \, \mathrm{B}_2^l(s)^\top \, \mathbf{B}_1^l(t)^\top \, a^l \, \mathrm{d}s + \varepsilon_i(t) \\ &= \mathrm{B}^0(t)^\top a^0 + \sum_{l=1}^{p} \mathrm{Q}_i^l(t)^\top a^l + \varepsilon_i(t)\,, \end{aligned}$$

avec $\mathrm{Q}_i^l(t) = \mathbf{B}_1^l(t)^\top \mathbf{B}_2^l(t)^\top x_i^l$.

Nous avons à la fin :

$$\mathrm{Y}_i(t) = \mathrm{Q}_i(t)^\top a + \varepsilon_i(t) \tag{B.5.8}$$

avec $a = (a^0, a^1, a^2, \ldots, a^p)^\top$ et $\mathrm{Q}_i(t) = \left(\mathrm{B}_0(t)^\top, \mathrm{Q}_i^1(t)^\top, \mathrm{Q}_i^2(t)^\top, \ldots, \mathrm{Q}_i^p(t)^\top\right)^\top$ deux vecteurs de taille $q_\gamma = q_{\gamma^0} + \sum_{l=1}^{p} q_{\gamma^l}^2$.

Les deux modèles fonctionnels étant transformés en modèles matriciels, l'estimation peut aisément se faire à l'aide des méthodes algorithmes développés dans la littérature. Toutefois, un des problèmes fondamental des méthodes d'estimation basées sur la décomposition en base de fonctions est le choix du nombre de fonctions de la base de fonctions pour les paramètres. Ce choix est d'autant plus important qu'il permet non seulement de contrôler le sur- ou le sous- ajustement du modèle mais aussi l'interprétabilité des paramètres. Pour contrôler la forme des paramètres à estimer, la littérature préconise une approche de pénalisation en imposant des contraintes de nullité sur les dérivées. Nous avons dans cette optique pénaliser l'estimation de nos paramètres en imposant une contrainte de régularité de type Ridge sur la dérivée seconde des paramètres. Ce qui donne,

Pour le modèle concurrent,

$$\begin{aligned} \mathrm{Pen}(\beta_l) &= \lambda_l \int \beta_l^{''}(t)^2 \, dt = \lambda_l \int \left[\sum_{j=1}^{q_{\beta^l}} b_j^l \phi_j^{l^{"}}(t) \right]^2 dt = \lambda_l \sum_{s,k=1}^{q_{\beta^l}} b_s^l b_k^l \Phi_{sk}^l, \\ \text{avec} \quad \Phi_{sk}^l &= \int \phi_s^{l^{"}}(t) \, \phi_k^{l^{"}}(t) \, dt. \end{aligned}$$

Lorsque λ_l est grand, la forme lisse du paramètre $\beta_l(.)$ est préférée par rapport à son adéquation aux données et on peut donc être en présence de sous-ajustement du modèle. Si au contraire λ_l est petit, c'est l'adéquation aux données qui est préférée et on peut être en présence de sur-ajustement.

Ainsi, pour une valeur donnée de λ_l, l'estimation de $\big(\beta_l(t)\big)_{0\leq l\leq p}$ est obtenue en résolvant :

$$\begin{aligned}\min_{b,\,\Gamma}\ \mathcal{L}_{pen}(b,\ \Gamma) &= \min_{b,\,\Gamma}\ -2\,\mathcal{L}(b,\ \Gamma\,|\,\mathrm{Y}) \ +\ \sum_{l=0}^{p}\lambda_l\sum_{s,k=1}^{q_{\beta l}} b_s^l b_k^l \Phi_{sk}^l \\ &= \min_{b,\,\Gamma}\ -2\,\mathcal{L}(b,\ \Gamma\,|\,\mathrm{Y}) \ +\ b^\top(\lambda\mathrm{P})\ b\,, \end{aligned} \tag{B.5.9}$$

où $\lambda\mathrm{P}\in\mathbb{R}^{q_\beta\times q_\beta}$ est donné par :

$$\lambda\mathrm{P}=\begin{pmatrix} \lambda_0\Psi^0 & 0_{q_{\beta^0}\times q_{\beta^1}} & \dots & 0_{q_{\beta^0}\times q_{\beta^p}} \\ 0_{q_{\beta^1}\times q_{\beta^0}} & \lambda_1\Psi^1 & \dots & 0_{q_{\beta^1}\times q_{\beta^p}} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{q_{\beta^p}\times q_{\beta^0}} & 0_{q_{\beta^p}\times q_{\beta^1}} & \dots & \lambda_p\Psi^p \end{pmatrix} \quad \text{avec } \Psi^l=\begin{pmatrix} \Phi_{11}^l & \Phi_{12}^l & \dots & \Phi_{1q_{\beta l}}^l \\ \Phi_{21}^l & \Phi_{22}^l & \dots & \Phi_{2q_{\beta l}}^l \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{q_{\beta l}1}^l & \Phi_{q_{\beta l}2}^l & \dots & \Phi_{q_{\beta l}q_{\beta l}}^l \end{pmatrix}.$$

Ici, $0_{q_1\times q_2}$ est la notation standard pour la matrice nulle de dimensions $q_1\times q_2$. Comme Ψ^l est une matrice symétrique définie positive pour tout $0\leq l\leq p$, on peut implémenter facilement cette contrainte grâce à sa décomposition de cholesky.

Pour le modèle intégral,

les paramètres étant bivariés la contrainte de régularité change par rapport au modèle concurrent et on a :

$$\mathrm{Pen}(\gamma_l)=\lambda_l\int\int\left\|\mathbf{H}_{\gamma_l}(t,s)\right\|^2 ds\,dt=\lambda_l\int\int\left\|\begin{bmatrix}\frac{\partial^2\gamma_l(t,s)}{\partial t^2} & \frac{\partial^2\gamma_l(t,s)}{\partial t\,\partial s}\\ \frac{\partial^2\gamma_l(t,s)}{\partial s\,\partial t} & \frac{\partial^2\gamma_l(t,s)}{\partial s^2}\end{bmatrix}\right\|^2 ds\,dt.$$

Où $\mathbf{H}_f(t,s)$ est la matrice hessienne de la fonction bivarié f et $\|\cdot\|$ la norme de Frobenius.

Pour simplifier les expressions, nous pouvons utiliser les notations: $\dfrac{\partial^2\gamma_l(t,s)}{\partial t\,\partial s}\equiv\gamma_l^{ts}(t,s)$, et on a ainsi

$$\mathrm{Pen}(\gamma_l) \ = \ \lambda_l\int\int\left(\gamma_l^{tt}(t,s)^2+2\,\gamma_l^{ts}(t,s)^2+\gamma_l^{ss}(t,s)^2\right)ds\,dt. \tag{B.5.10}$$

On sait d'après (B.5.7) que

$$\begin{aligned}\gamma_l(t,s)^2 &= \left(a^{l^\top}\mathbf{B}_1(t)\mathrm{B}_2(s)\right)^2=\left(\sum_{i,j=1}^{q_{\gamma l}}a_{ij}^l\mathbf{B}_{1i}^l(t)\mathrm{B}_{2j}^l(s)\right)^2 \\ &= \sum_{i,j,k,m=1}^{q_{\gamma l}}a_{ij}^l a_{km}^l\mathbf{B}_{1i}^l(t)\mathbf{B}_{1k}^l(t)\,\mathrm{B}_{2j}^l(s)\mathrm{B}_{2m}^l(s),\end{aligned}$$

On a alors les expressions suivantes pour les dérivées partielles :

$$\begin{cases} \int\int \gamma_l^{tt}(t,s)^2\,ds\,dt &= \sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l \int \mathbf{B}_{1i}^{l''}(t)\mathbf{B}_{1k}^{l''}(t)\,dt \int \mathrm{B}_{2j}^l(s)\mathrm{B}_{2m}^l(s)\,ds \\ &= \sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l\, \Phi_{1,ik}^{l''}\, \Phi_{2,jm}^l\,; \\ \int\int \gamma_l^{ts}(t,s)^2\,ds\,dt &= \sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l \int \mathbf{B}_{1i}^{l'}(t)\mathbf{B}_{1k}^{l'}(t)\,dt \int \mathrm{B}_{2j}^{l'}(s)\mathrm{B}_{2m}^{l'}(s)\,ds \\ &= \sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l\, \Phi_{1,ik}^{l'}\, \Phi_{2,jm}^{l'}\,; \\ \int\int \gamma_l^{ss}(t,s)^2\,ds\,dt &= \sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l \int \mathbf{B}_{1i}^{l}(t)\mathbf{B}_{1k}^{l}(t)\,dt \int \mathrm{B}_{2j}^{l''}(s)\mathrm{B}_{2m}^{l''}(s)\,ds \\ &= \sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l\, \Phi_{1,ik}^{l}\, \Phi_{2,jm}^{l''}\,. \end{cases}$$

avec la notation $\Phi_{u,sk}^{l'} = \int \mathrm{B}_{us}^{l'}(t)\mathrm{B}_{uk}^{l'}(t)\,dt.$

L'estimation des paramètres $\big(\gamma_0(t), \gamma_1(t,s),\ \ldots,\ \gamma_p(t,s)\big)$ est obtenu en résolvant le problème :

$$\begin{aligned} \min_{a,\ \Gamma} \mathcal{L}_{pen}(a,\ \Gamma) \quad &= \quad \min_{a,\ \Gamma} \text{-}\,2\,\mathcal{L}(a,\ \Gamma \,|\, \mathrm{Y}) + \ \lambda_0 \sum_{s,k=1}^{q_{\gamma^0}} a_s^0 a_k^0 \Phi_{sk}^0 + \\ &\sum_{l=1}^{p} \lambda_l \sum_{i,j,k,m=1}^{q_{\gamma^l}} a_{ij}^l a_{km}^l \Big(\Phi_{1,ik}^{l''}\, \Phi_{2,jm}^l + 2\,\Phi_{1,ik}^{l'}\, \Phi_{2,jm}^{l'} + \Phi_{1,ik}^{l}\, \Phi_{2,jm}^{l''} \Big). \end{aligned}$$

Intervalles de confiance :
Une fois paramètres estimés, l'étape suivante les constructions des intervalles de confiance des prédictions donnés par les différents modèles. Dans l'optique de construire des intervalles robustes pour des prédictions fonctionnelles, nous nous sommes plongés dans le cadre de la prédiction conforme qui est un cadre fournissant des mesures valides de confiance aux prédictions individuelles. L'idée principale de la méthode est de construire des régions autour des prédictions individuelles, de sorte qu'avec un certain niveau de confiance, la véritable valeur de la réponse se trouve dans ces régions. L'intervalle résultant présente des propriétés intéressantes, telles que l'indépendance de la distribution et la non-asymptoticité. Mais malgré ces bonnes propriétés, les méthodes conformes classiques conduisent à des intervalles de longueur constante ou faiblement variable dans l'espace des entrées. C'est dans ce contexte que Romano et al. (2019) ont proposé une méthode entièrement adaptative à l'hétéroscédasticité, appelée régression quantile conforme (CQR), qui combine la prédiction conforme à la régression quantile, héritant ainsi des avantages des deux méthodes. Nous adaptons la méthode CQR à la régression fonction-sur-fonction selon le processus suivant :

Pour un échantillon $\Big\{(\mathrm{X}_i(t_j), \mathrm{Y}_i(t_j)), 1 \le i \le n\ \ 1 \le j \le m\Big\}$, :

Étape 1 : Construction de deux régressions quantiles fonctionnelles $\widehat{\mathrm{q}}_{\tau_0}\Big(\mathrm{X}_i(t_j)\Big)$ et $\widehat{\mathrm{q}}_{\tau_1}\Big(\mathrm{X}_i(t_j)\Big)$ avec $\tau_0 < \tau_1$ sur l'échantillon d'entraînement $\mathcal{A}_1$. Cette régression est obtenu en combinant la perturbation de la régression linéaire standard (Minnier et al., 2011) et le calcul des quantiles multivariés par transport optimal. Le détail est donné dans la suite.

Étape 2 : Sur l'échantillon de validation $\mathcal{A}_2$, on calcule les scores de conformité E_i définis par :

$$E_i := \left\{ \max\Big(\widehat{\mathrm{q}}_{\tau_0}\big(\mathrm{X}_i(t_j)\big) - \mathrm{Y}_i(t_j)\,;\, \mathrm{Y}_i(t_j) - \widehat{\mathrm{q}}_{\tau_1}\big(\mathrm{X}_i(t_j)\big)\Big) \right\}_{1\le j\le m} \in \mathbb{R}^m \qquad \text{pour } i \in \mathcal{A}_2$$

Étape 3 : L'intervalle de prédiction d'une nouvelle observation $\widehat{\mathrm{Y}}_{n+1}(t_j)$ d'entrée $\mathrm{X}_{n+1}(t_j)$ est donnée par :

$$C\Big(\mathrm{X}_{n+1}(t_j)\Big) \quad = \quad \Big[\widehat{\mathrm{q}}_{\tau_0}\Big(\mathrm{X}_{n+1}(t_j)\Big) - Q_{1-\tau}(\mathcal{A}_2)\,;\, \widehat{\mathrm{q}}_{\tau_1}\Big(\mathrm{X}_{n+1}(t_j)\Big) + Q_{1-\tau}(\mathcal{A}_2)\Big] \quad \text{(B.5.11)}$$

Où $Q_{1-\tau}(\mathcal{A}_2)$ est le quantile multivarié au niveau $\Big((1-\tau)(1+\frac{1}{|\mathcal{A}_2|})\Big)$ des erreurs E_i.

Construction de la régression quantile fonctionnelle :
En ADF, la principale difficulté relative à la construction de la régression quantile est la définition de l'équivalent de la fonction de perte quantile ("pinball" loss) lorsque la variable réponse est fonctionnelle. Bien que des travaux dans la littérature ont proposé des fonctions de perte point-par-point, nous avons préconisé une approche plus robuste utilisant la théorie de la perturbation. Ainsi, la régression quantile que nous avons développé consiste à perturber la régression linéaire standard. Ensuite à partir des prédictions données par chaque régression perturbée, nous calculons leur quantile multivarié afin d'obtenir la régression quantile fonctionnelle. Il reste toutefois le problème de calcul quantile multivarié qui est encore à ce jour un problème ouvert. Hallin (2022) a proposé une définition du quantile multivarié basé sur le transport optimal introduit par Chernozhukov et al. (2017) dont le processus est schématiquement décrit par les trois étapes suivantes :

Étape 1: Construction d'une grille uniforme dans la boule unité

Puisqu'en dimension $d > 1$, la majeure partie du volume d'une boule unité est concentrée dans une couche fine près de sa surface, il n'est pas trivial de générer des points uniformément dans la boule unité. Pour cela, il est nécessaire de s'assurer que la distribution résultante corresponde à la distribution des volumes des couronnes. Dans l'espace de dimension d, les volumes de la boule de rayon r et de la boule unité (rayon 1) sont respectivement donnés par

$$\mathrm{V}_{d,r} = \frac{\pi^{d/2}}{\Gamma\left(\frac{d}{2}+1\right)}\, r^d \quad \text{et} \quad \mathrm{V}_{d,1} = \frac{\pi^{d/2}}{\Gamma\left(\frac{d}{2}+1\right)}.$$

Ainsi, le volume de la couronne sera :

$$\mathrm{V}_{\mathcal{C}_r} = \mathrm{V}_{d,1} - \mathrm{V}_{d,r} = \mathrm{V}_{d,1}(1 - r^d).$$

La probabilité d'être dans cette couronne, pour la distribution générée, est

$$p_{\mathcal{C}_r} = \frac{\mathcal{V}_r}{\mathrm{V}_{d,1}} = 1 - r^d.$$

Avec ce résultat, nous pouvons déduire la fonction de répartition (et la fonction de densité) de la variable aléatoire R des rayons des boules devant se trouver dans la couronne :

$$\mathrm{F}_{\mathrm{R}}(r) = 1 - \mathbb{P}(R > r) = 1 - (1 - r^d) = r^d \quad \text{et} \quad f_{\mathrm{R}}(r) = d\, r^{d-1}.$$

Nous proposons donc l'algorithme suivant pour générer une grille uniforme dans la boule unité :

1. Simuler un ensemble de points en dimension d selon une distribution normale.
2. Normaliser les points simulés.
3. Simuler autant de rayons r que de points générés selon la distribution F_{R}. Ensuite,
$$r = \exp\left(\frac{\log(u)}{d}\right) \quad \text{avec } u \sim \mathcal{U}([0;1]).$$
4. Enfin, chaque point normalisé est multiplié par un rayon simulé.

Avec ce processus, nous garantissons que la distribution uniforme générée respectera la géométrie et la distribution des volumes dans des espaces de haute dimension.

Étape 2 : Trouver le rayon r qui exclut une proportion τ du volume de la boule

Nous répondons dans cette partie à la question : "De combien le rayon de la boule unité doit-il être réduit pour que le volume exclu représente τ% du volume total ?"
Dans ce contexte, τ représente le rapport du volume de la couronne formée par la boule unité et la boule de rayon r par rapport au volume de la boule unité.

$$\tau \quad = \quad \frac{\mathrm{V}_{\mathcal{C}_r}}{\mathrm{V}_{d,1}} = 1 - r^d \quad \Longleftrightarrow \quad r = \exp\left(\frac{\log(1-\tau)}{d}\right) \tag{B.5.12}$$

Ensuite, pour exclure τ% du volume total, nous pouvons réduire le rayon de la boule unité de 1 à r.

Étape 3: Définition du quantile multivarié

Une fois que nous avons compris comment construire correctement une grille uniforme sur la boule unité de dimension d et comment réduire le rayon de cette boule pour extraire un volume bien défini, la dernière étape consiste à déterminer le quantile multivarié empirique τ du jeu de points $Z_1^{(n)}, \ldots, Z_n^{(n)}$ dans $\mathbb{R}^d$. Le quantile empirique τ est obtenu via un transport optimal de la distribution d'échantillon vers la grille uniforme déterministe sur la boule unité. En réduisant le rayon de cette boule de 1 à r selon la formule (B.5.12), le quantile τ est donné par l'enveloppe convexe des observations exclues par la réduction du rayon.

Romano et al. (2019) établissent que si les éléments de l'échantillon $\{(\mathrm{X}_i(t_j), \mathrm{Y}_i(t_j)), 1 \leq i \leq n+1\}$ sont interchangeables et si les scores de conformité E_i sont presque sûrement distinct, alors l'intervalle de prédiction (B.5.11) vérifie

$$1 - \tau \leq \mathbb{P}\Big\{\widehat{\mathrm{Y}}_{n+1}(t_j) \in C\Big(\mathrm{X}_{n+1}(t_j)\Big)\Big\} \leq 1 - \tau + \frac{1}{|\mathcal{A}_2| + 1}.$$

Des expériences sur des données simulées et réelles ont montré l'efficacité de notre approche. Une autre application de cette méthode au cas d'utilisation de la prédiction de la QoE du streaming vidéo a également été effectué et a montré que notre approche est capable de challenger des modèles d'apprentissage profond. Toutefois et pour améliorer la capacité de prédiction de notre modèle, nous avons ensuite développer un modèle capable de s'adapter à des hétérogènes. C'est le cadre des modèles de mélange et plus spécifiquement des modèles de mélange d'experts.

Modèles hétérogènes de prédiction de QoE :
Les modèles de mélanges et particulièrement les modèles de mélange d'experts supposent qu'il existe $\mathrm{K} \in \mathbb{N}^\star$ groupes au sein de la population. Chacun d'eux représenté par une variable catégorielle représenté sous forme binaire par : $\mathrm{Z} = (\mathrm{z}_1, \ldots, \mathrm{z}_\mathrm{K})$ où $\mathrm{z}_k = 1$ si l'observation appartient au groupe k et 0 sinon. Le modèle s'écrit donc

$$\mathrm{MR}(\mathrm{Y}|\mathrm{X}) \quad = \quad \sum_{k=1}^{\mathrm{K}} \pi_k \mathbb{E}_k[\mathrm{Y}|\mathrm{X}, \mathrm{z}_k = 1] \tag{B.5.13}$$

où π_k est le poids du groupe k associé à l'expert k $\mathbb{E}_k[\mathrm{Y}|\mathrm{X}]$. On suppose un modèle concurrent pour l'expert k qui s'écrit:

$$\mathbb{E}_k[\mathrm{Y}(t)|\mathrm{X}(t), \mathrm{z}_k = 1] \quad = \quad \mathrm{X}(t)^\top \beta_k(t) \tag{B.5.14}$$

où $\beta_k(t) = (\beta_{k,0}(t), \beta_{k,1}(t), \ldots, \beta_{k,p}(t))$ sont les paramètres fonctionnels de l'expert k.

Dans le modèle de mélange d'experts, on suppose que la valeur de Z (l'appartenance à un groupe) dépend des covariables X soit : $\pi_k = \pi_k(\mathrm{X}) = \mathbb{P}(\mathrm{z}_k = 1 \mid \mathrm{X})$.

Ainsi, la densité conditionnelle $\mathrm{Y}(t)$ dans le modèle de mélange d'expert fonctionnel (FFMoE) est donné par :

$$f(\mathrm{Y}(t)|\mathrm{X}(t), \Psi(t)) \quad = \quad \sum_{k=1}^{\mathrm{K}} \pi_k(\mathrm{X}(t), \alpha_k(t)) \Phi(\mathrm{Y}(t); \mathrm{X}(t)\beta_k(t), \sigma_k^2), \tag{B.5.15}$$

avec

- $\pi_k(\mathrm{X}(t),\ \alpha_k(t))$ la probabilité d'appartenir au groupe k ou $k^{\text{ème}}$ fonction d'activation, dépendant des covariables $\mathrm{X}(t)$ à travers les paramètres $\alpha_k(t)$.
- $\Psi_k(t) = (\beta_k(t), \alpha_k(t))$ les paramètres fonctionnels à estimer ;
- $\Phi(\mathrm{Y}(t); \mathrm{X}_i(t)\beta_k(t), \sigma_k^2)$ la densité de la loi gaussienne d'espérance $\mathrm{X}(t)\beta_k(t)$ et de variance σ_k^2.

Pour modéliser la fonction d'activation, on a considéré le modèle logistique multinomial donné par :

$$\pi_k(\mathrm{x}_i(t), \alpha_k(t)) = \frac{\exp(h_k(\mathrm{x}_i(t),\ \alpha_k(t)))}{1 + \sum_{k'=1}^{\mathrm{K}-1} \exp(h_{k'}(\mathrm{x}_i(t),\ \alpha_{k'}(t)))}, \tag{B.5.16}$$

où

$$h_k(\mathrm{x}_i(t),\ \alpha_k(t)) \quad = \quad \int_{\mathrm{T}} \alpha_k^\top(s)\mathrm{x}_i(s)ds \tag{B.5.17}$$

avec $\alpha_k(t) = (\alpha_{k,0}(t), \alpha_{k,1}(t), \ldots, \alpha_{k,p}(t))^\top$.

Comme avec les covariables fonctionnels, on suppose que $\alpha_k(t)$ s'exprime dans une base de fonctions de la forme :

$$\alpha_{k,\ell}(t) = \sum_{j=1}^{\mathrm{L}_{\alpha^\ell}} a_{k,j}^\ell \varrho_j^\ell(t) = \varrho^\ell(t)^\top a_k^\ell = \varrho(t) a_k.$$

Dans ce cas, l'équation ((B.5.17)) devient :

$$h_k(\mathrm{x}_i(t), \alpha_k(t)) = \int_{\mathrm{T}} a_k^\top \varrho(s)^\top \mathrm{B}(s)\ x_i ds = a_k^\top \underbrace{\int_{\mathrm{T}} \varrho(s)^\top \mathrm{B}(s) dt x_i}_{r_i} = a_k^\top r_i,$$

Le Modèle (B.5.16) peut donc se réécrire sous la forme :

$$\pi_k(\mathrm{x}_i(t), \alpha_k(t)) \quad = \quad \frac{\exp(a_k^\top r_i)}{1 + \sum_{k'=1}^{\mathrm{K}-1} \exp(a_{k'}^\top r_i)}. \tag{B.5.18}$$

Enfin, pour garantir l'identifiabilité des paramètres $\alpha_k(t) \in \mathrm{L}^2(\mathbb{R}^{p+1})$, $k = 1, \ldots, \mathrm{K}$, on fixe $\alpha_{\mathrm{K}}(t)$ égale à la fonction nulle (par conséquent a_{K} fixé au vecteur null) (Jiang and Tanner, 1999).

En utilisant la représentation matricielle des termes fonctionnels, on peut écrire la vraisemblance incomplète du modèle par :

$$\mathcal{L}(\Psi; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) \quad = \quad \sum_{i=1}^{n} \log(\sum_{k=1}^{\mathrm{K}} \pi_k(\mathrm{x}_i(t), \alpha_k(t)) \Phi_m(\mathbf{y}_i; b_k^\top \mathrm{R}_i, \mathrm{V}_{k,i})) \tag{B.5.19}$$

où $\mathbf{y}_i$ est le vecteur de taille m contenant toutes les mesure de l'observation i, R_i et $\mathrm{V}_{k,i}$ sont respectivement la matrice de design et le bloc matrice de covariance de V_k associé à l'observation i.

La log-vraisemblance complétée peut donc s'écrire:

$$\begin{aligned}\mathcal{L}_c(\Psi; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) \quad = \quad & \sum_{i=1}^{n} \sum_{k=1}^{\mathrm{K}} z_{ik} \log(\frac{\exp(a_k^\top r_i)}{1 + \sum_{u=1}^{\mathrm{K}-1} \exp(a_u^\top r_i)} . \frac{1}{\sqrt{(2\pi)^m |\mathrm{V}_{k,i}|}} \\ & \exp(-\frac{1}{2}(\mathbf{y}_i - b_k^\top \mathrm{R}_i)^\top \mathrm{V}_{k,i}^{-1}(\mathbf{y}_i - b_k^\top \mathrm{R}_i))).\end{aligned} \tag{B.5.20}$$

Avec $\Psi^{(0)} = ((a_1^{(0)}, b_1^{(0)}, \sigma_1^{2^{(0)}}), \ldots, (a_{\mathrm{K}}^{(0)}, b_{\mathrm{K}}^{(0)}, \sigma_{\mathrm{K}}^{2^{(0)}}), \mathrm{U}^{(0)}, \Gamma^{(0)})$ la valeur initiale de Ψ, on estime les paramètres à partir de l'algorithme EM (Dempster et al., 1977).

Une approche régularisée est ensuite proposée, basée sur une pénalité de type Lasso sur les dérivées secondes des paramètres fonctionnels, à la fois des experts et des fonctions d'activation. Cette régularité pénalise la forme des paramètres fonctionnels afin de fournir des estimateurs interprétables. Elle est formulée mathématiquement par:

$$\mathrm{Pen}(\Psi) = \sum_{k=1}^{\mathrm{K}} \sum_{\ell=0}^{p} \lambda_{k,\ell} \int \beta_{k,\ell}^{''}(t)^2 dt + \sum_{k=1}^{\mathrm{K}-1} \sum_{\ell=1}^{p} \gamma_{k,\ell} \int \alpha_{k,\ell}^{''}(t)^2 dt$$

où

$$\int \beta_{k,\ell}^{''}(t)^2 dt = \int \left[\sum_{j=1}^{\mathrm{L}_{\beta\ell}} b_{k,j}^{\ell} \phi_j^{\ell^{''}}(t) \right]^2 dt = \sum_{s,u=1}^{\mathrm{L}_{\beta\ell}} b_{k,s}^{\ell} b_{k,u}^{\ell} \Gamma_{su}^{\ell}$$

avec $\Gamma_{su}^{\ell} = \int \phi_s^{\ell^{''}}(t) \phi_u^{\ell^{''}}(t) dt$, et

$$\int \alpha_{k,\ell}^{''}(t)^2 dt = \int \left[\sum_{j=1}^{\mathrm{L}_{\alpha\ell}} a_{k,j}^{\ell} \varrho_j^{\ell^{''}}(t) \right]^2 dt = \sum_{s,u=1}^{\mathrm{L}_{\beta\ell}} a_{k,s}^{\ell} a_{k,u}^{\ell} \Upsilon_{su}^{\ell}$$

avec $\Upsilon_{su}^{\ell} = \int \varrho_s^{\ell^{''}}(t) \varrho_u^{\ell^{''}}(t) dt$.
Et donc,

$$\mathrm{Pen}(\Psi) = \sum_{k=1}^{\mathrm{K}} \sum_{\ell=0}^{p} \lambda_{k,\ell} \sum_{s,u=1}^{\mathrm{L}_{\beta\ell}} b_{k,s}^{\ell} b_{k,u}^{\ell} \Gamma_{su}^{\ell} + \sum_{k=1}^{\mathrm{K}-1} \sum_{\ell=1}^{p} \gamma_{k,\ell} \sum_{s,u=1}^{\mathrm{L}_{\beta\ell}} a_{k,s}^{\ell} a_{k,u}^{\ell} \Upsilon_{su}^{\ell} \qquad \text{(B.5.21)}$$

où $\lambda_{k,\ell}$ et $\gamma_{k,\ell}$ sont les paramètres de calibrage de la pénalité.

En utilisant l'expression matricielle, on obtient :

$$\mathcal{L}_{pen}(\Psi; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) = \mathcal{L}(\Psi; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) - \sum_{k=1}^{\mathrm{K}} b_k^{\top} (\lambda_k \mathrm{P})\, b_k - \sum_{k=1}^{\mathrm{K}-1} a_k^{\top} (\gamma_k \mathrm{Q})\, a_k$$

où $(\lambda_k \mathrm{P}) \in \mathbb{R}^{\mathrm{L}_\beta \times \mathrm{L}_\beta}$ est donné par :

$$(\lambda_k \mathrm{P}) = \begin{pmatrix} \lambda_{k,0}\Gamma^0 & 0_{\mathrm{L}_{\beta^0} \times \mathrm{L}_{\beta^1}} & \ldots & 0_{\mathrm{L}_{\beta^0} \times \mathrm{L}_{\beta^p}} \\ 0_{\mathrm{L}_{\beta^1} \times \mathrm{L}_{\beta^0}} & \lambda_{k,1}\Gamma^1 & \ldots & 0_{\mathrm{L}_{\beta^1} \times \mathrm{L}_{\beta^p}} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{\mathrm{L}_{\beta^p} \times \mathrm{L}_{\beta^0}} & 0_{\mathrm{L}_{\beta^p} \times \mathrm{L}_{\beta^1}} & \ldots & \lambda_{k,p}\Gamma^p \end{pmatrix} \quad \text{avec } \Gamma^{\ell} = \begin{pmatrix} \Gamma_{11}^{\ell} & \Gamma_{12}^{\ell} & \ldots & \Gamma_{1\mathrm{L}_{\beta\ell}}^{\ell} \\ \Gamma_{21}^{\ell} & \Gamma_{22}^{\ell} & \ldots & \Gamma_{2\mathrm{L}_{\beta\ell}}^{\ell} \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma_{\mathrm{L}_{\beta\ell}1}^{\ell} & \Gamma_{\mathrm{L}_{\beta\ell}2}^{\ell} & \ldots & \Gamma_{\mathrm{L}_{\beta\ell}\mathrm{L}_{\beta\ell}}^{\ell} \end{pmatrix};$$

et $(\gamma_k \mathrm{Q}) \in \mathbb{R}^{\mathrm{L}_\alpha \times \mathrm{L}_\alpha}$ est donné par :

$$(\gamma_k \mathrm{Q}) = \begin{pmatrix} \gamma_{k,0}\Upsilon^0 & 0_{\mathrm{L}_{\alpha^0} \times \mathrm{L}_{\alpha^1}} & \ldots & 0_{\mathrm{L}_{\alpha^0} \times \mathrm{L}_{\alpha^p}} \\ 0_{\mathrm{L}_{\alpha^1} \times \mathrm{L}_{\alpha^0}} & \gamma_{k,1}\Upsilon^1 & \ldots & 0_{\mathrm{L}_{\alpha^1} \times \mathrm{L}_{\alpha^p}} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{\mathrm{L}_{\alpha^p} \times \mathrm{L}_{\alpha^0}} & 0_{\mathrm{L}_{\alpha^p} \times \mathrm{L}_{\alpha^1}} & \ldots & \gamma_{k,p}\Upsilon^p \end{pmatrix} \quad \text{avec } \Upsilon^{\ell} = \begin{pmatrix} \Upsilon_{11}^{\ell} & \Upsilon_{12}^{\ell} & \ldots & \Upsilon_{1q_{\alpha\ell}}^{\ell} \\ \Upsilon_{21}^{\ell} & \Upsilon_{22}^{\ell} & \ldots & \Upsilon_{2\mathrm{L}_{\alpha\ell}}^{\ell} \\ \vdots & \vdots & \ddots & \vdots \\ \Upsilon_{\mathrm{L}_{\alpha\ell}1}^{\ell} & \Upsilon_{\mathrm{L}_{\alpha\ell}2}^{\ell} & \ldots & \Upsilon_{\mathrm{L}_{\alpha\ell}\mathrm{L}_{\alpha\ell}}^{\ell} \end{pmatrix}.$$

Enfin, la log-vraisemblance complétée est déterminée comme avec la relation (B.5.20). Elle donnée par :

$$\mathcal{L}^{c}_{pen}(\Psi; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) = \mathcal{L}_c(\Psi; \{y_i(t_j), \mathrm{x}_i(t_j)\}_{i,j}) - \sum_{k=1}^{\mathrm{K}} b_k^{\top}(\lambda_k \mathrm{P})\, b_k - \sum_{k=1}^{\mathrm{K}-1} a_k^{\top}(\gamma_k \mathrm{Q})\, a_k. \quad \text{(B.5.22)}$$

De même, l'estimation des paramètres est effectué en maximisant cette fonction grâce à l'algorithme EM.

L'utilité du modèle proposé est illustrée sur des données réelles et simulées. L'application au cas d'utilisation de la VoIP est présentée au chapitre 5 et montre la pertinence de la méthode par rapport aux modèles sans mélange. En somme, cette thèse a servi à étendre les limites des modèles de prédiction de QoE existants en introduisant des méthodologies statistiques innovantes dans les cadres de l'ADF et des modèles de mélange d'experts. Ces travaux ouvrent la voie à de futures recherches et applications dans le domaine de l'évaluation de la QoE, ce qui permettra de répondre avec plus de précision et d'efficacité à la demande croissante de services de télécommunications.