ISSN 1746-7659, England, UK
Journal of Information and Computing Science
Vol. 15, No. 2, 2020, *pp*.134-140

Classification with application to Functional Data based on Gaussian process

Xin Liu, Chunzheng Cao
School of Mathematics and Statistics, Nanjing University of Information Science & Technology,
Nanjing 210044, China
(Received June 01, 2020, accepted July 05, 2020)

Abstract. In this paper, we briefly introduce four methods for functional classification. To compare the effects of the four models, we generate the data from Gaussian process based on a functional mixed-effects model, square exponential kernel is used in random-effect term to describe the nonlinear structure of the data. The outcomes show that the two functional classification models have a better prediction correct rate than the two machine learning classification models.

Keywords: functional classification, functional mixed-effects model, kernel function.

1. Introduction

Functional data analysis (FDA) is used in many research fields which is of great theoretical and practical value, for example: spatio-temporal data analysis in meteorology [1], pharmacokinetic analysis in medicine [2], genetic profiling analysis in biology [3, 4], image data analysis which is ultra-high dimension [5] and the clustering and prediction of traffic flow data [6]. For functional data, there is a strong correlation between variables, there are also complex correlations between the observations of the variables for each subject, these do not meet the assumptions of the common statistical analysis methods. And FDA can solve these questions.

An important concern of functional data analysis is classification, which means that we want to assign an individual to a pre-designed discrete category based on the observed functional or image data. The existing functional data classification methods can be divided into the following three categories: (1) Based on probability density [7]. (2) Based on the algorithm [8]. (3) Based on regression [9]. Although there are many literatures discussing the classification of functional data, there are still many problems that have not been considered: the influence of covariates on classification cannot be considered, the prediction after classification and the correlation between each subject is not taken into account. In this paper, we will propose a model which can solve these problems. There are also some classical machine learning classification method such as BP neural network [10] and SVM [11], which are very mature approaches.

The rest of this paper is organized as follows. In section 2, we briefly introduce four classification models: two functional classification models and two machine learning classification models and then give their Parameter estimation process. In section 3, we design a simulation study for the models proposed in this paper and compare the performance between the four models. Finally, a brief summary was given in section 4.

2. Model

In this section, we will briefly introduce four classification model: two functional classification models and two machine learning classification models.

2.1. Functional Generalized Linear Model

For Generalized linear model (GLM), we have a general structure

$$g(\mu) = \beta_0 + \beta_1^T \boldsymbol{X}, \tag{1}$$

where $\boldsymbol{X}$ is predictor, $\mu = E(Y; \theta, \phi)$, Y is response variable with density

$$p(y; \theta, \phi) = \exp\left\{\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi)\right\}, \tag{2}$$

and $g(\cdot)$ is a link function. But when the $\boldsymbol{X}$ is functional ,i.e.$\boldsymbol{X}(t)$, the GLM model may not be used, what we will do is to replace the summation over the dimensional space with an integral over the infinite dimensional one,

$$g(\mu) = \beta_0 + \int \omega_1(t)X(t)dt$$

although t is infinite dimensional in theory, in practice t is a finite set of time points. Now we use some basis function to approximate $X(t)$,

$$X(t) = s(t)^T\boldsymbol{c}, \boldsymbol{c} \sim N(\boldsymbol{\mu_c}, \Gamma)$$

then we can give the Functional generalized linear model (FGLM),

$$\begin{aligned} p(y_i;\theta_i,\phi) &= \exp\left\{\frac{y_i\theta_i - b(\theta_i)}{a(\phi)} + c(y_i,\phi)\right\}, \\ g(\mu_i) &= \beta_o + \int \omega_1(t)X(t)dt \\ &= \beta_o + \int \omega_1(t)s(t)^T\boldsymbol{c_i}dt \\ &= \beta_0 + \beta_1^T\boldsymbol{c}_i. \end{aligned}$$

For functional classification, we usually choose logistic function as link function and Y becomes a Bernoulli variable, then FGLM becomes

$$Y_i = \begin{cases} 1 & with\,probability\,\{1+\exp(-\beta_0 - \beta_1^T\boldsymbol{c}_i)\}^{-1}, \\ 0 & with\,probability\,\{1+\exp(\beta_0 + \beta_1^T\boldsymbol{c}_i)\}^{-1}, \end{cases}$$

In general, if $E(Y|\boldsymbol{X}) = P(Y=1|\boldsymbol{X}) > 0.5$, then a new subject is predicted as 1 and 0 otherwise. For parameter estimation, we first can get the observed data likelihood,

$$\begin{aligned} l(\boldsymbol{\mu_c},\Gamma,\beta_0,\beta_1,\phi,\sigma_\epsilon^2) = \sum_{i=1}^{N}&\left\{\frac{y_i\theta_i - b(\theta_i)}{a(\phi)} + c(y_i,\phi)\right\} \\ &-\textstyle\sum_{i=1}^{N}\{\frac{n_i}{2}\log(\sigma_\epsilon^2) + \frac{1}{2\sigma_\epsilon^2}(\boldsymbol{x}_i - S_i\boldsymbol{c}_i)^T(\boldsymbol{x}_i - S_i\boldsymbol{c}_i)\} \\ &-\textstyle\sum_{i=1}^{N}\left\{\frac{1}{2}\log|\Gamma| + \frac{1}{2}(\boldsymbol{c}_i - \boldsymbol{\mu_c})^T\Gamma^{-1}(\boldsymbol{c}_i - \boldsymbol{\mu_c})\right\}. \end{aligned} \tag{3}$$

but for this model we can not optimize directly because $\boldsymbol{c}_i$s are unobserved, so we use the EM algorithm which iterates between a maximization (M-step) and an expectation (E-step) to optimize the observed likelihood.

2.2. Functional Generalized Additive Model

Before introduce Functional Generalized Additive Model (FGAM), we first introduce Generalized Additive Model (GAM). For linear regression with a Bernoulli variable Y and a set of predictor variables $X_1,\dots,X_p$, we have model,

$$Y = g\left(\beta_0 + \textstyle\sum_{j=1}^{p}\beta_jX_j + \epsilon\right), \tag{4}$$

where $g(\cdot)$ is a link function, and GAM replaces the linear function β_jX_j by a non-linear function to get

$$Y = g\left(\beta_0 + \textstyle\sum_{j=1}^{p}f_j(X_j) + \epsilon\right).$$

And FGAM is the extension of FAM to functional predictor, which can be expressed as:

$$Y = g\left(\beta_0 + \textstyle\sum_{j=1}^{p}f_j(X_j(t)) + \epsilon\right).$$

For the estimation of the FGAM, we can use kernel estimation method to get $f_j(\cdot)$:

$$\widehat{f_j^l}(X_j) = \frac{\sum_{i=1}^{N}(Y_i - \hat{Y}_i^{-j,l})K_j(\frac{d_j(X_j,X_{ij})}{h_j})}{\sum_{i=1}^{N}K_j(\frac{d_j(X_j,X_{ij})}{h_j})}, \tag{5}$$

where $\hat{Y}_i^{-j,l} = \sum_{i=1}^{j-1}\widehat{f_j^l}(X_{i\cdot}) + \sum_{i=j+1}^{p}\hat{f}_i^{(l-1)}(X_{i\cdot})$ is the prediction without variable j, d_j is the distance (induced by the norm) , and K_j and h_j are an asymmetric kernel function and the bandwidth, respectively.

2.3. BP Neural Network

BP neural network is a kind of multi-layer feedforward neural network, whose characteristic are forward signal transmission and error back propagation. In the process of forward propagation, the input