Bootstrap aggregated classification for sparse functional data

Hyunsung Kim

July 24, 2020

Department of Statistics
Chung-Ang University

Outline

Introduction

Introduction

- Functional data is collected in the form of curves or functions in various fields such as meteorology and health science, among others and analyzing this is called functional data analysis (FDA).
- Since functional data are defined on infinite dimensions, dimensionality reduction becomes a key issue and one of the powerful method is functional principal component analysis (FPCA) (Silverman, 1996).
- In real-data analysis, functional data are commonly observed at sparse or irregular time points, therefore James et al. (2000) and Yao et al. (2005) proposed the FPCA method for sparse functional data with different ideas.

Introduction

- Similar to conventional multivariate data analysis, classifying into several groups is also important problem in FDA and there are some related works.
- Lee (2004) presented a support vector machine (SVM) based on FPC scores, and Rossi and Villa (2006) proposed a functional SVM (FSVM), extending the SVM for functional data.
- Song et al. (2008) compared classifiers based on FPC scores using linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), a k-nearest neighbor (KNN) classifier, and SVM.
- Gama (2004) proposed a functional tree model and bootstrap aggregating (bagging) functional trees.

Introduction

- In this study, we propose a new classification method for sparse functional data based on FPCA and bootstrap aggregating.
- Bagging is an ensemble method that enhances predictions by combining the classifiers from the bootstrap samples.
- By extending the bagging to functional data, we construct a bagged classification model, which combines classifiers based on the FPC scores from the bootstrap samples.

FPCA for sparse functional data

Functional principal component analysis

- $\{X(t) : t \in \mathcal{T}\}$: a square integrable random process in $L^2(\mathcal{T})$
  - ▷ $\mu(t) = E\big[X(t)\big], \quad t \in \mathcal{T}$
  - ▷ $G(s,t) = \mathsf{cov}\big[X(s), X(t)\big], \quad s,t \in \mathcal{T}$
- By Mercer's theorem, the covariance function can be represented as

$$G(s,t) = \sum_{k=1}^{\infty} \lambda_k \phi_k(s) \phi_k(t),$$

where $\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$ are nonnegative eigenvalues satisfying $\sum_{k=1}^{\infty} \lambda_k < \infty$, and ϕ_k is the corresponding orthonormal eigenfunction.

Functional principal component analysis

- Given n random curves, $\mathbf{X} = [X_1(t), \ldots, X_n(t)]$, the Karhunen–Loève expansion of $X_i(t)$ can be represented as

$$X_i(t) = \mu(t) + \sum_{k=1}^{\infty} \xi_{ik}\phi_k(t), \qquad t \in \mathcal{T},$$

where $\xi_{ik} = \int_{\mathcal{T}} \big(X_i(t) - \mu(t)\big)\phi_k(t)dt$ are uncorrelated variables with mean 0 and variance λ_k.

Functional principal component analysis

- The truncated approximation is written as

$$X_i(t) \approx \mu(t) + \sum_{k=1}^{K} \xi_{ik}\phi_k(t), \quad t \in \mathcal{T},$$

  where K is the number of basis functions.
- K is often selected using the proportion of variance explained (PVE), but the Akaike information criterion (AIC) or Bayesian information criterion (BIC) can also be used for consistency (Yao et al., 2005; Li et al., 2013).

FPCA for sparse functional data

- When each curve is observed at sparse or irregular time points, we cannot apply a conventional FPCA directly to the data.
- Especially, the covariance function cannot be computed easily and the estimated FPC scores are biased.
- James et al. (2000) proposed a reduced-rank model based on the mixed-effects model and estimated the FPC function and scores using an EM algorithm.
- Yao et al. (2005) proposed the principal component analysis through the conditional expectation (PACE) method to obtain unbiased FPC scores.
- Here, we consider PACE method.

FPCA for sparse functional data by PACE

- Consider an ith curve $\mathbf{X}_i = \big(X_i(t_{i1}), \ldots, X_i(t_{in_i})\big)^T$ with the mean function $\boldsymbol{\mu}_i = \big(\mu(t_{i1}), \ldots, \mu(t_{in_i})\big)^T$.
- Here, $t_{ij} \in \mathcal{T}$ is the jth time point observed in the ith curve X_i, for $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, n_i$.
- Let $\mathbf{U}_i = \big(U_i(t_{i1}), \ldots, U_i(t_{in_i})\big)^T$ be an observed ith curve with additional measurement errors, $\boldsymbol{\epsilon}_i = \big(\epsilon_i(t_{i1}), \ldots, \epsilon_i(t_{in_i})\big)^T$.

FPCA for sparse functional data by PACE

- Then, we have

$$\begin{aligned} U_i(t_{ij}) &= X_i(t_{ij}) + \epsilon_i(t_{ij}) \\ &= \mu(t_{ij}) + \sum_{k=1}^{\infty} \xi_{ik}\phi_k(t_{ij}) + \epsilon_i(t_{ij}), \qquad t_{ij} \in \mathcal{T}, \end{aligned}$$

where $\epsilon_i(t_{ij})$ is an $i.i.d.$ error with mean zero and variance σ^2 and is assumed to be independent of ξ_{ik}, for $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, n_i$, $k = 1, 2, \ldots$.

FPCA for sparse functional data by PACE

- Under the assumption that ξ_{ik} and ϵ_i are jointly Gaussian, the best linear unbiased prediction (BLUP) of ξ_{ik} is computed as

$$\tilde{\xi}_{ik} = E[\xi_{ik}|\mathbf{U}_i] = \lambda_k \boldsymbol{\phi}_{ik}^T \boldsymbol{\Sigma}_{\mathbf{U}_i}^{-1}(\mathbf{U}_i - \boldsymbol{\mu}_i),$$

where $\boldsymbol{\phi}_{ik} = \big(\phi_k(t_{i1}), \ldots, \phi_k(t_{in_i})\big)^T$ is the ith FPC function, and $\boldsymbol{\Sigma}_{\mathbf{U}_i} = \mathrm{cov}(\mathbf{U}_i, \mathbf{U}_i) = \mathrm{cov}(\mathbf{X}_i, \mathbf{X}_i) + \sigma^2 \mathbf{I}_{n_i}$.

FPCA for sparse functional data by PACE

- Yao et al. (2005) considered the following prediction for the trajectory $X_i(t)$ using the first K eigenfunctions:

$$\hat{X}_i(t) = \hat{\mu}(t) + \sum_{k=1}^{K} \hat{\xi}_{ik}\hat{\phi}_k(t),$$

where $\hat{\mu}(t)$, $\hat{\xi}_{ik}$, and $\hat{\phi}_k(t)$ are estimations obtained from the entire data sample.

Ensemble classification via FPCA

Classification with functional predictors

- To perform the classification for functional data, we consider the functional generalized linear model (FGLM) (James, 2002; Müller and Stadtmüller, 2005).
- Given the ith functional curve $X_i(t)$ and the corresponding response y_i, the FGLM can be represented as

$$g(\mu) \quad = \alpha + \int_{\mathcal{T}} \beta(t) X_i(t) dt, \tag{1}$$

  where $\mu = E(y_i|X_i)$, and $g(\cdot)$ is a link function.
- Because we can only observe $X_i(t)$ at a finite time points n_i, the integral can be substituted by a summation.

Classification with functional predictors

- The estimate of the coefficient $\beta(\cdot)$ may be unstable because it is an extremely high-dimensional vector.
- We can solve this problem by expanding $\beta(\cdot)$ using a set of basis functions.
- By employing an FPCA, we can use data-driven orthonormal basis functions to represent $\beta(\cdot)$.

Classification based on FPC scores

- Let $\phi_k(t)$, for $k = 1, 2, \ldots$, be an orthonormal basis function, and expand $X_i(t)$ and $\beta(t)$ as

$$X_i(t) = \sum_{k=1}^{K} \xi_{ik}\phi_k(t), \quad \beta(t) = \sum_{k=1}^{K} \beta_k\phi_k(t)$$

  using the K-truncated FPCA model described in Section 2.
- Then, the FGLM in (1) is represented as

$$g(\mu) = \alpha + \sum_{k=1}^{K} \beta_k\xi_{ik}.$$

- The estimation is stable because most of the variation can be expressed using a small number, K.

Bagging

- Bootstrap aggregating (Bagging) is a popular ensemble method that uses the bootstrap idea proposed by Breiman (1996).
- It extracts several samples with replacement and obtains the prediction by aggregating models from each bootstrap sample.

Bootstrap aggregated functional classifier via sparse FPCA

Notations

- $\boldsymbol{\mathcal{D}} = \{(\mathbf{U}_i, y_i) : i = 1, \ldots, n\}$: the set of sparse n curves with $\mathbf{U}_i = \big(U_i(t_{i1}), \ldots, U_i(t_{in_i})\big)^T$.
- $y_i \in \{1, \ldots, g\}$: the response class label.
  (Here, we assume $g = 2$)
- $\boldsymbol{\mathcal{D}}^{(b)} = \Big\{\Big(\mathbf{U}_i^{(b)}, y_i^{(b)}\Big) : i = 1, \ldots, n\Big\}$, for $b = 1, \ldots, B$: a bootstrap resample from $\boldsymbol{\mathcal{D}}$.

Bootstrap aggregated functional classifier via sparse FPCA

Algorithm

1. For $b = 1, \dots, B$, repeat
   (a) Generate a bootstrap resample $\boldsymbol{\mathcal{D}}^{(b)}$ from the data $\boldsymbol{\mathcal{D}}$.
   (b) Perform FPCA for $\boldsymbol{\mathcal{D}}^{(b)}$.
   (c) Estimate the FPC scores by PACE and chose K FPCs.
   (d) Construct a classifier using K FPC scores.
2. Given a new curve $\mathbf{U}^*(t)$,
   (a) Estimate the FPC scores by PACE from $\boldsymbol{\mathcal{D}}^{(b)}$, for $b = 1, \dots, B$.
   (b) Obtain the prediction $\hat{f}^{(b)}$ from the bth classifier constructed in step 1-(d), for $b = 1, \dots, B$.
3. Obtain the final prediction $\hat{y}_{bag}$ aggregated from $\hat{f}^{(1)}, \dots, \hat{f}^{(B)}$.

Aggregating methods

Majority vote

- The majority vote simply chooses the class that receives the highest total vote from all classifiers.
- The bagged classifier by the majority vote is

$$\hat{y}_{\textsf{bag}} = \arg\max_{j \in \{1,2\}} \frac{\sum_{b=1}^{B} \mathbf{I}\{\hat{f}^{(b)}(x) = j\}}{B},$$

where $\mathbf{I}$ is the indicator function.

Aggregating methods

Out-of-bag (OOB) error weighted vote (Pham, 2018)

- Let e_b, for $b = 1, \ldots, B$, be the OOB errors from B bootstrapped models.
- By defining the weight as $w_b = 1/e_b$, the models that exhibit good performance receive higher weights.
- If $e_b = 0$, which means there are no errors from the OOB samples, we set $e_b = \min\{e_1, \ldots, e_{b-1}, e_{b+1}, \ldots, e_B\}$.
- The bagged classifier using the OOB error weighted vote is

$$\hat{y}_{\mathsf{bag}} = \frac{\sum_{b=1}^{B} w_b \hat{f}^{(b)}(x)}{\sum_{b=1}^{B} w_b}.$$

Simulation studies

Simulation 1

- We generate $N = 200$ curves with two classes as

$$U_{gi}(t_{ij}) = \mu_g(t_{ij}) + \sum_{k=1}^{3} \xi_{gk}\phi_k(t_{ij}) + \epsilon_i(t_{ij}), \;\; i = 1, \ldots, 100, \; j = 1, \ldots$$

  where $g \in \{0, 1\}$ indicates a group label.
- The FPC functions are defined as

$$\phi_k(t) = \begin{cases} \cos(\pi k t/5)/\sqrt{5}, & k \text{ is odds} \\ \sin(\pi k t/5)/\sqrt{5}, & k \text{ is even,} \end{cases} \quad (2)$$

- The FPC scores ξ_{gk} are sampled from $i.i.d.$ $N(0, \lambda_{gk})$, for $k = 1, 2, 3$.
- The measurement error $\epsilon_i(t)$ is sampled from $i.i.d.$ $N(0, 0.5^2)$.

Simulation 1

Table 1: [Simulation 1] The parameters for three models.

Model	g	$\mu_g(t)$	$(\lambda_{g1}, \lambda_{g2}, \lambda_{g3})$
(A) Different means and variances	0	$t + \sin(t)$	$(4, 2, 1)$
	1	$t + \cos(t)$	$(16, 8, 4)$
(B) Different means	0	$t + \sin(t)$	$(4, 2, 1)$
	1	$t + \cos(t)$	$(4, 2, 1)$
(C) Different variances	0	$t + \sin(t)$	$(4, 2, 1)$
	1	$t + \sin(t)$	$(16, 8, 4)$

Simulation 1

- To make each curve sparse, the number of observations for the ith curve, n_i, is randomly selected from $\{5, 6, \ldots, 10\}$, and the corresponding time points t_{ij}, for $j = 1, \ldots, n_i$, are selected from $i.i.d.\ Uniform(0, 10)$.
- For validation, we randomly split the generated data into a training and a test set, with 100 elements in each.
- We compare the results of the proposed method with those of six single functional classification models: logistic regression, SVM with linear kernel, SVM with gaussian kernel, LDA, QDA, and naive Bayes.

Simulation 1

Table 2: The average classification errors (%) and standard errors (in parentheses) from 100 Monte Carlo repetitions for three designs of simulation 1.

Model	Method	Logistic Regression	SVM (Linear)	SVM (Gaussian)	LDA	QDA	Naive Bayes
(A)	Single	17.13 (4.86)	17.03 (5.22)	14.86 (5.26)	16.38 (4.93)	14.94 (5.20)	16.12 (4.64)
	Majority vote	14.98 (4.15)	15.24 (4.29)	12.81 (4.50)	15.03 (4.11)	13.55 (4.47)	14.98 (4.21)
	OOB weight	14.80 (3.95)	14.83 (4.10)	**12.56 (4.33)**	14.83 (4.00)	13.02 (4.16)	14.39 (4.02)
(B)	Single	11.42 (3.44)	10.89 (3.47)	11.60 (3.99)	10.71 (3.48)	12.19 (3.50)	13.56 (4.23)
	Majority vote	10.24 (3.24)	10.14 (3.21)	10.68 (3.57)	10.04 (3.12)	11.16 (3.33)	12.06 (3.35)
	OOB weight	10.30 (3.23)	**9.97 (3.18)**	10.52 (3.41)	10.04 (3.10)	11.03 (3.30)	11.82 (3.31)
(C)	Single	50.43 (5.72)	49.38 (5.60)	32.48 (4.89)	50.46 (5.73)	30.90 (4.45)	30.31 (4.51)
	Majority vote	49.62 (5.63)	48.41 (6.16)	31.11 (5.26)	49.49 (5.70)	30.70 (4.26)	29.70 (4.42)
	OOB weight	49.19 (5.78)	48.20 (6.21)	30.99 (5.32)	49.01 (5.75)	30.57 (4.18)	**29.59 (4.34)**

* The minimum error rate is marked in bold.

Simulation 2

- Simulation 2 is motivated by Yao et al. (2016).
- We generate $n = 700$ curves as

$$U_i(t) = \sum_{k=1}^{50} \xi_{ik}\phi_k(t) + \epsilon_i(t), \quad i = 1, \ldots, n.$$

- The FPC functions are generated as in (2), and the FPC scores, ξ_{ik}, are sampled from $i.i.d.\ N(0, k^{-3/2})$, for $k = 1, \ldots, 50$.
- For the sparsity of the data, the number of observations for the ith curve, n_i, is randomly selected from $\{10, 11, \ldots, 20\}$, and the corresponding time points t_{ij}, for $j = 1, \ldots, n_i$, are selected from $i.i.d.\ Uniform(0, 10)$.
- The measurement error $\epsilon_i(t)$ is sampled from $i.i.d.\ N(0, 0.1)$.

Simulation 2

- Now, we consider the following three models:

$$
\begin{aligned}
&\text{(A)} \quad f(U_i) = \exp(\langle \beta_1, U_i \rangle / 2) - 1, \\
&\text{(B)} \quad f(U_i) = \arctan(\pi \langle \beta_1, U_i \rangle) + \exp(\langle \beta_2, U_i \rangle / 3) - 1, \\
&\text{(C)} \quad f(U_i) = \arctan(\pi \langle \beta_1, U_i \rangle / 4),
\end{aligned}
$$

  where $\langle f, g \rangle = \displaystyle\int_{\mathcal{T}} f(t)g(t)dt$ for $f, g \in L^2(\mathcal{T})$.
- Here, $\beta_1(t) = \sum_{k=1}^{50} b_k \phi_k(t)$, where $b_k = 1$ for $k = 1, 2$, $b_k = (k-2)^{-3}$ for $k = 3, \ldots, 50$, and $\beta_2(t) = \sqrt{3/10}(t/5 - 1)$.

Simulation 2

- The class label for each curve, U_i, is defined as $y_i = \text{sign}\{f(U_i) + \epsilon_i\}$, where $\epsilon_i \sim \ i.i.d.\ N(0, 0.1)$.
- We randomly split the generated data into 200 training curves and 500 test curves and compare the results of the proposed methods with those of the single classification models.

Simulation 2

Table 3: The average classification errors (%) and standard errors (in parentheses) from 100 Monte Carlo repetitions for three models in simulation 2.

Model	Method	Logistic Regression	SVM (Linear)	SVM (Gaussian)	LDA	QDA	Naive Bayes
A	Single	16.71 (2.33)	16.82 (2.20)	17.50 (2.76)	16.62 (2.30)	17.77 (2.56)	18.41 (2.66)
	Majority vote	15.62 (1.95)	15.86 (1.87)	16.19 (2.28)	15.79 (1.96)	16.51 (2.14)	17.32 (2.42)
	OOB weight	**15.54 (1.93)**	15.78 (1.83)	16.17 (2.27)	15.73 (1.89)	16.43 (2.10)	17.21 (2.40)
B	Single	12.85 (2.41)	12.79 (2.40)	13.27 (2.65)	12.77 (2.40)	13.83 (2.56)	14.77 (2.74)
	Majority vote	11.20 (1.84)	11.14 (1.89)	11.54 (1.98)	11.19 (1.85)	11.93 (2.03)	13.29 (2.36)
	OOB weight	11.11 (1.87)	**11.00 (1.81)**	11.46 (1.94)	11.11 (1.85)	11.83 (2.00)	13.08 (2.35)
C	Single	14.46 (2.17)	14.34 (2.18)	15.27 (2.69)	14.29 (2.17)	15.32 (2.36)	16.05 (2.22)
	Majority vote	13.15 (1.73)	13.14 (1.78)	13.62 (2.08)	13.14 (1.82)	13.78 (1.90)	14.88 (2.09)
	OOB weight	13.11 (1.77)	**13.06 (1.77)**	13.54 (2.02)	13.09 (1.80)	13.74 (1.89)	14.72 (2.08)

* The minimum error rate is marked in bold.

Real-data analysis

Real-data analysis - Berkely growth data

- The Berkely growth data set (Tuddenham and Snyder, 1954) includes heights for 93 individuals (54 girls and 39 boys).
- There are 31 observations from ages 1 to 18 for each curve.

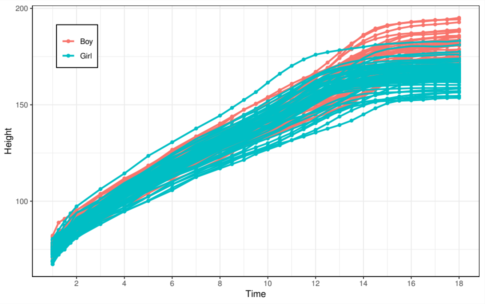

Figure 1: The Berkely growth data of 93 individuals.

Real-data analysis - Berkely growth data

- For sparseness, we artificially sparsify the data.
- The number of observations for each individual is randomly selected from $\{12, 13, \ldots, 15\}$, and the corresponding time points are randomly selected from the original time points.
- For validation, we randomly divide the 93 curves into 62 training and 31 test curves.
- We repeat this process 100 times with different splits.

Real-data analysis - Berkely growth data

Table 4: The average classification errors (%) and standard errors (in parentheses) from 100 random splits of the Berkerly growth data.

Method	Logistic Regression	SVM (Linear)	SVM (Gaussian)	LDA	QDA	Naive Bayes
Single	7.26 (4.80)	5.26 (3.20)	5.68 (4.03)	5.81 (3.34)	5.65 (3.35)	5.65 (3.90)
Majority vote	5.94 (4.12)	4.90 (3.19)	5.26 (3.51)	5.39 (3.24)	4.87 (3.57)	5.45 (3.96)
OOB weight	6.06 (4.36)	5.13 (3.11)	5.32 (3.68)	5.42 (3.24)	**4.87 (3.42)**	5.39 (3.97)

Real-data analysis - Spinal bone mineral density data

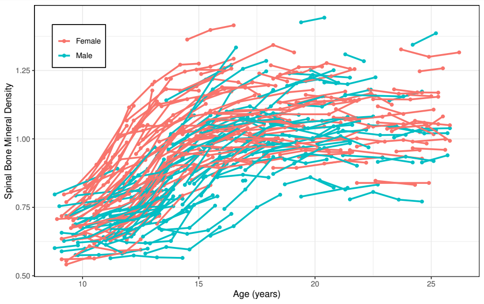

Figure 2: The spinal bone mineral density of 280 individuals.

Real-data analysis - Spinal bone mineral density data

- The spinal bone mineral density data (Bachrach et al., 1999) contains spinal bone mineral densities for 280 individuals (153 females and 127 males).
- Each curve was measured at sparse and irregular time points with 2 to 4 observations.
- The data are randomly divided into 187 training and 93 test sets.
- we apply the methods to 100 different splits of the data.

Real-data analysis - Spinal bone mineral density data

Table 5: The average classification errors (%) and standard errors (in parentheses) from 100 random splits of the spinal bone mineral density data

Method	Logistic Regression	SVM (Linear)	SVM (Gaussian)	LDA	QDA	Naive Bayes
Single	32.20 (4.15)	32.22 (4.21)	32.64 (4.05)	31.90 (4.22)	34.21 (4.25)	32.58 (4.02)
Majority vote	**31.12 (4.25)**	31.39 (4.27)	31.50 (4.56)	31.16 (4.15)	32.34 (3.90)	31.29 (3.88)
OOB weight	31.19 (4.15)	31.45 (4.23)	31.57 (4.44)	31.28 (4.12)	32.32 (3.95)	31.27 (3.87)

Conclusion and discussion

Conclusion and discussion

- In this study, we propose a new ensemble classification method for sparse functional data, which is a bagged model that combines classifiers based on FPC scores.
- To obtain FPC scores for sparse functional data, we perform FPCA by PACE method.
- We consider two aggregating methods, a majority vote and an OOB weighted vote, and find that both methods outperform the single classifiers.
- The results of several simulations confirm that the proposed classification model outperforms single classifiers in various situations.
- In two real-data analyses, the proposed model shows better performance than the single model.

Conclusion and discussion

- The proposed method can be easily extended to multi-class classification problems, in which we expect the aggregating method to outperform single classifiers.
- In addition, other ensemble methods such as boosting and stacking can be used (Opitz and Maclin, 1999).

Reference

Reference i

Bachrach, L. K., Hastie, T., Wang, M. C., Narasimhan, B., & Marcus, R. (1999). Bone mineral acquisition in healthy Asian, Hispanic, black, and Caucasian youth: a longitudinal study. *The journal of clinical endocrinology & metabolism*, **84**, 4702-4712.

Breiman, L. (1996). Bagging predictors. *Machine learning*, **24**, 123-140.

Breiman, L. (1996). Out-of-bag estimation. Technical report, Statistics Dept., University of California at Berkeley, CA.

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

Friedman, J., Hastie, T., & Tibshirani, R. (2001). *The elements of statistical learning.*, Springer series in statistics, New York.

Gama, J. (2004). Functional trees. *Machine Learning*, **55**, 219-250.

Reference ii

Hall, P., & Hosseini-Nasab, M. (2006). On properties of functional principal components analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **68**, 109-126.

James, G. M., Hastie, T. J., & Sugar, C. A. (2000). Principal component models for sparse functional data. *Biometrika*, **87**, 587-602.

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

James, G. M. (2002). Generalized linear models with functional predictors. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **64**, 411-432.

Lee, H. J. (2004). Functional data analysis: classification and regression. Ph. D. Dissertation, Texas A&M University.

Leng, X., & Müller, H. G. (2006). Classification using functional data analysis for temporal gene expression data. *Bioinformatics*, **22**, 68-76.

Reference iii

Li, Y., Wang, N., & Carroll, R. J. (2013). Selecting the number of principal components in functional data. *Journal of the American Statistical Association*, **108**, 1284-1294.

Müller, H. G. (2005). Functional modelling and classification of longitudinal data. *Scandinavian Journal of Statistics*, **32**, 223-240.

Müller, H. G., & Stadtmüller, U. (2005). Generalized functional linear models. *the Annals of Statistics*, **33**, 774-805.

Opitz, D., & Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of artificial intelligence research*, **11**, 169-198.

Pham, H. T. (2018). Generalized weighting for bagged ensembles. Ph. D. Dissertation, Iowa State University.

Ramsay, J. O., & Silverman, B. W. (2005) *Functional Data Analysis*, 2nd ed., Springer Series in Statistics, New York.

Reference iv

Rossi, F., & Villa, N. (2006). Support vector machine for functional data classification. *Neurocomputing*, **69**, 730-742.

Silverman, B. W. (1996). Smoothed functional principal components analysis by choice of norm. *The Annals of Statistics*, **24**, 1-24.

Song, J. J., Deng, W., Lee, H. J., & Kwon, D. (2008). Optimal classification for time-course gene expression data using functional data analysis. *Computational biology and chemistry*, **32**, 426-432.

Tuddenham, R. D., & Snyder, M. M. (1954). Physical growth of California boys and girls from birth to eighteen years. *University of California publications in child development*, **1**, 183-364.

Yao, F., Müller, H. G., & Wang, J. L. (2005). Functional data analysis for sparse longitudinal data. *Journal of the American Statistical Association*, **100**, 577-590.

Yao, F., Wu, Y., & Zou, J. (2016). Probability-enhanced effective dimension reduction for classifying sparse functional data. *Test*, **25**, 1-22.

Thank You!