Achieving near perfect classification for functional data

November 24, 2016
Introduction

Centroid classifier

Numerical studies
Problem

Independent and identically distributed data pairs \((X_i, l_i)_{i=1}^n\),
- \(X_i\) is a random function defined on a compact interval \(I\);
- \(l_i\) is class label, taking the values of 0 or 1;
- population is a mixture of subpopulations \(\Pi_0\) and \(\Pi_1\).
Our goal is to classify functional data.
Assumptions

- These two subpopulations have a common covariance, which admits the following eigen-decomposition:

\[ \text{Cov}(X(u), X(v)) = K(u, v) = \sum_{j=1}^{\infty} \theta_j \phi_j(u) \phi_j(v); \]

- WLLG, assume that \( E_0(X(t)) = 0 \) and \( E_1(X(t)) = \mu(t) \), where \( E_k \) denotes the expectation for \( X \) from \( \Pi_k, k = 0, 1 \);
- \( K \) strictly positive definite and uniformly bounded;
- The mean difference can be expanded in terms of \( \phi_j(t) \)'s:
  \[ \mu(t) = \sum_{j=1}^{\infty} \mu_j \phi_j(t). \]
The centroid classifier is based on the statistic:

\[ T(X) = D^2(X, \bar{X}_1) - D^2(X, \bar{X}_0), \]

where

- \( \bar{X}_k \) denotes the sample mean curve of \( \Pi_k \), \( k = 0, 1 \);
- \( D \) denotes a distance measure between two functions;
- \( X \) will be assigned to \( \Pi_0 \) if \( T > 0 \) and \( \Pi_1 \) otherwise.

In this paper, the distance is defined as

\[ D(X, \bar{X}_k) = |\langle X, \psi \rangle - \langle \bar{X}_k, \psi \rangle|, \]

where \( |\langle X, \psi \rangle| = \int_{\mathcal{I}} X \psi \) and \( \psi \) is a function defined on \( \mathcal{I} \). Then \( T(X) \) converges to

\[ |\langle X, \psi \rangle - \langle \mu, \psi \rangle|^2 - \langle X, \psi \rangle^2, \]

denoted as \( T^0(X) \).
Properties

Defined $Q = \int_{I}(X - E(X))\psi$, $\nu = \langle \mu, \psi \rangle$, $\sigma^2 = \text{Var}(Q)$, $\psi^{(r)} = \sum_{j=1}^{r} \theta_j^{-1}\mu_j\phi_j$. Then under Gaussian assumptions,

- if $\sum_{j=1}^{\infty} \theta_j^{-2}\mu_j^2 < \infty$, then $T^0$ is optimal but imperfect and the classification error is $1 - \Phi(\nu/2\sigma)$;
- the optimal classifier can be computed via the sequence of $(\psi^{(r)})_{r=1}^{\infty}$.
- if $\sum_{j=1}^{\infty} \theta_j^{-1}\mu_j^2 = \infty$, the classification error rate of $T^0$ is 0. But $\psi^{(r)}$ does not converge.

In this paper, the distance is defined as $D(X, \bar{X}_k) = |\langle X, \psi \rangle - \bar{X}_k, \psi \rangle|$, where $|\langle X, \psi \rangle = \int_{I} X\psi$ and $\psi$ is a function defined on $I$. 

\[
\begin{align*}
\int_{I}(X - E(X))\psi, & \quad \nu = \langle \mu, \psi \rangle, \quad \sigma^2 = \text{Var}(Q), \quad \psi^{(r)} = \\
\sum_{j=1}^{r} \theta_j^{-1}\mu_j\phi_j. & \\
\end{align*}
\]
Choice of $\psi$ in practice

Two approaches are proposed:

- FPCA combined with truncation: $\hat{\psi}(r) = \sum_{j=1}^{r} \hat{\theta}_j^{-1} \hat{\mu}_j \hat{\phi}_j$;
- the truncation level $r$ can be chosen via cross validation;
- partial functional linear regression, (review)
- choosing $r$ based on cross validation.
FPCA explains the largest part of the empirical covariance of $X$;

PLS tries to capture as much of the linear relation between $X$ and $Y$ as possible;

if the main differences between the means of $\Pi_0$ and $\Pi_1$ come from $\mu_j$ with $j$ relatively small, then the two approaches will give very similar results.

otherwise PLS usually outperforms FPCA-based approach.
Setup of three simulations

Two approaches are proposed:

▶ $K = 5$-fold (reverse) CV, 200 splits for a training sample;

▶ training data sample size $n$, $n/2$ for each subpopulation, $B = 200$ test data, $M = 200$ simulations,

▶ simulation 1: $\theta_j = j^{-2}, 1 \leq j \leq 40, \mu_{jk} = 0, j > 6,$

$(\mu_{10}, \ldots, \mu_{60}) = (0, -0.5, 1, -0.5, 1, 0.5)$ and

$(\mu_{11}, \ldots, \mu_{61}) = (0, -0.75, 0.75, -0.15, 1.4, 0.1)$.

▶ simulation 2: $\theta_{j0} = 1.5\theta_j$;

▶ $\theta_j = \exp\left[-2.1 - (j - 1)/20\right]^2$, $1 \leq j \leq 40$, $\mu_{j0} = 0$ and $\mu_{j1} = 0.75(-1)^{j+1}I(j \leq 3)$;

▶ largest $\theta_j$s are for $j$ close to 40, whereas the $\mu_j$s are all 0, except for the first three.
Table 1. Percentage of misclassified observations in the simulated examples: mean of $P_m$ (with the standard deviation of $P_m$ in parentheses) calculated from $M = 200$ Monte Carlo simulations.

<table>
<thead>
<tr>
<th>Data</th>
<th>n</th>
<th>Results (%) for the following classifiers:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$CENT_{PC1}$</td>
</tr>
<tr>
<td>Example 1</td>
<td>30</td>
<td>4.45 (4.31)</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>3.13 (2.26)</td>
</tr>
<tr>
<td>Example 2</td>
<td>30</td>
<td>6.51 (5.45)</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>4.84 (4.32)</td>
</tr>
<tr>
<td>Example 3</td>
<td>30</td>
<td>50.0 (3.64)</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>49.6 (3.70)</td>
</tr>
</tbody>
</table>
Remark

- In the first two examples, $\sum_j \mu_j^2 \theta_j^{-1}$ are large, $\theta_j$ is the $j$th largest eigenvalue of the pooled centred data. FPCA and PLS perform similarly.

- In the third example, FPCA will encounter difficulties in practice, as to work well this method needs to estimate reasonably well the principal components corresponding to the smallest 38th-40th eigenvalues. PLS outperforms FPCA method.
Wheat sample classification

Description:

- near infrared spectra of 100 wheat samples with known protein content, and measured from 1100 nm to 2500 nm in 2 nm intervals.
- two populations $\Pi_0$ (protein content less than 15) and $\Pi_1$ (protein content greater than 15) of sizes $N_0 = 41$ and $N_1 = 59$, respectively,

Centroid classifier:

- sample size of training data: $n = 30$ or $50$, test size $100 - n$,
- randomly split $M = 200$ times for the whole data,
- FPCA, PLS, multivariate FPC scores, nonparametric classifiers,
Fig. 2. Wheat derivative curves when the protein level is (a) less or (b) greater than 15, and (c) a typical test sample projected on $\psi^r$ when $n = 30$ (O, level less than 15; +, level greater than or equal to 15)
Table 2. Percentage of misclassified observations: mean of $P_m$ (and standard deviation of $P_m$ in parentheses) calculated from $M = 200$ randomly chosen test samples

<table>
<thead>
<tr>
<th>Data</th>
<th>$n$</th>
<th>Results (%) for the following classifiers:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$CENT_{PC1}$</td>
</tr>
<tr>
<td>Wheat</td>
<td>30</td>
<td>0.89 (2.49)</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>0.22 (1.09)</td>
</tr>
<tr>
<td>Rain</td>
<td>30</td>
<td>10.6 (3.09)</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>10.3 (2.90)</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>9.45 (3.02)</td>
</tr>
<tr>
<td>Phoneme</td>
<td>30</td>
<td>22.5 (3.59)</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>20.8 (2.08)</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>20.0 (1.09)</td>
</tr>
</tbody>
</table>