IE 582 Statistical Learning for Data Mining
Homework 2, due December 4th, 2020

Instructions: Please solve the following exercises using R (http://www.r-project.org/) or Python (https://www.python.org/). You are expected to use GitHub Classroom and present your work as an html file (i.e. web page) on your progress journals. There are alternative ways to generate an html page for you work:

- A Jupyter Notebook including your codes and comments. This works for R and Python, to enable using R scripts in notebooks, please check:
  - https://docs.anaconda.com/anaconda/navigator/tutorials/r-lang/
  - https://medium.com/@kyleake/how-to-install-r-in-jupyter-with-irkernel-in-3-steps-917519326e41

  Things are little easier if you install Anaconda (https://www.anaconda.com/). Please export your work to an html file. Please provide your *. ipynb file in your repository and a link to this file in your html report will help us a lot.

- A Markdown html document. This can be created using RMarkdown for R and Python-Markdown for Python

Note that html pages are just to describe how you approach to the exercises in the homework. They should include your codes. You are also required to provide your R/Python codes separately in the repository so that anybody can run it with minimal change in the code. This can be presented as the script file itself or your notebook file (the one with *.ipynb file extension).

The last and the most important thing to mention is that academic integrity is expected! Do not share your code (except the one in your progress journals). You are always free to discuss about tasks but your work must be implemented by yourself. As a fundamental principle for any educational institution, academic integrity is highly valued and seriously regarded at Boğaziçi University.

Task: Dimensionality reduction for time series data

The conference paper by Liu et al. (2009) starts with the following statement: "Gestures have recently become attractive for spontaneous interaction with consumer electronics and mobile devices in the context of pervasive computing". The aim is to provide efficient personalized gesture recognition on wide range of devices.

To achieve this, Liu et al. (2009) uses a single three-axis accelerometer to collect data from eight users to characterize eight gesture patterns. The library, uWaveGestureLibrary, consists over 4000 instances each of which has the accelerometer readings in three dimensions (i.e. *x*, *y* and *z*). Eight gestures are illustrated in Figure 1.

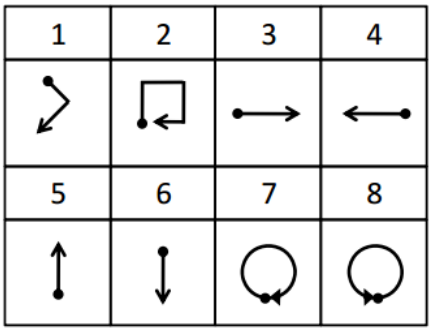

Figure 1: Gesture vocabulary considered by Liu et al. (2009). The dot denotes the start and the arrow the end

a) The dataset is provided in the following link:
https://drive.google.com/drive/u/1/folders/13553neknux7U8why55KM1WrjgkA9IJKm

Note that there are separate files for each axis and each row corresponds to one gesture in the files. First column has the class information. The information between second and last column is the time ordered observations in the corresponding axis (provided in the file name as X, Y or Z).

Read the data and visualize one instance (all axes) from each class and try to relate the shape (time series) you see with the gestures shown in Figure 1 (this is just for fun, sometimes it is good to start with data visualization to understand what is going on). A 3D scatter plot would be interesting. Note that this is an acceleration information. You can transform this information to a velocity vector by computing the cumulative sum of acceleration over time.

b) As you may have noticed, the data is provided as a regular data matrix (i.e. each row represents an instance and columns represent the time index of the observations). On the other hand, this is an example of multivariate time series where we have X, Y and Z variables. Suppose our aim is to reduce this multivariate time series to a univariate one with a dimensionality reduction approach. One way to achieve this task is to transform your data into the following format (so called long format):

time series id	time index	X	Y	Z	class
1	1	.	.	.	a
1	2	.	.	.	a
1	3	.	.	.	a
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
2	1	.	.	.	b
2	2	.	.	.	b
2	3	.	.	.	b
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
N	T-2	.	.	.	c
N	T-1	.	.	.	c
N	T	.	.	.	c

In other words, we create a new data matrix for which each row represent an observation from a time series at a particular time index. This is basically concatenation of observations from each axis based on time index and the time series id. Having the last column as the class information may be helpful for this part of the task.

Suppose we decide to reduce the data from 3D (i.e. X, Y and Z features) to 1D using PCA. Relevant columns for PCA is X, Y and Z. Apply PCA to the whole data (regardless of the time series id or class) and report PCA results.

Originally this problem is a classification problem in which the aim is to predict the gesture for a given time series. Reduction of 3D to 1D results in univariate time series (now you can attach the time series ids to the reduced features). Select 2 random time series from each class and

visualize the reduced dimensions as time series in a single plot to see if classes can be separated in the reduced dimensions. Visual inspection is enough. You can check the answer from the following link: https://stackoverflow.com/a/3777592 as an example of such visualization (i.e. multiple lines on a single plot).

c) It is also interesting to compare the first principal component when PCA is applied on the data from each gesture. In other words, you are expected to filter the data for each class and apply PCA (you will perform PCA eight times). Report PCA results (components, variance covered by each component). Is there any interesting observation on the first component of the PCA results applied to the data from each class? Are the first components similar? Comment on your findings.

d) Suppose, our aim now is to visualize the time series in reduced dimensions for classification purposes. Assume that we compute the distance between the time series for each axis using the original representation (i.e. a row represents individual time series, column is the time index and entries are the observations) over each axis and sum them up to obtain a final distance measure. You are expected to obtain a symmetric distance matrix. Let's apply multi-dimensional scaling to this distance matrix to represent each time series on a 2-dimensional feature space. Visualize the observations using the reduced features and color-code the points with the class information. Comment on your findings.

Reference

J. Liu, Z. Wang, L. Zhong, J. Wickramasuriya, and V. Vasudevan. uWave: Accelerometer-based personalized gesture recognition and its applications. Pervasive Computing and Communications, IEEE International Conference on, 0:1-9, 2009.
(link: http://www.ruf.rice.edu/~mobile/publications/liu09percom.pdf)