Project 4: Fashion-MNIST Data

Due Date: *Sunday, December 14 (11.59PM)*

Submission: on Gradescope

Instructions:

The projects may be completed in groups (up to 4 students). If you do so, please make sure that you include everyone's full name, and that you *also select everyone's name when submitting the assignment on Gradescope*. This ensures that each group member will get a grade assigned and have access to the comments from the graders.

For this project, the following items need to be submitted:

1. the **code** in `R` or *Python* (Markdown or Jupyter notebook are ok)
2. a **report** (as a pdf file) that summarizes the process you followed for data pre-processing and data analysis addresssing the specific questions below. The report **should not contain any code**, but you should incorporate *plots*, *relevant output*, and *other metrics* used in your analysis and decision making process. The report should include: (i) an introduction, (ii) the pre-processing steps, (iii) the methods you used with discussion of the results (as needed for each part) and (iv) all appropriate results. You can structure the report in the way that you see fit, probably by separating by task (see specific tasks below). Finally, the report should end with a conclusion where you briefly summarize your findings.

Project Description:

In this project we are going to use the Kaggle Fashion MNIST data set. You can directly download the data from Kaggle: you should be able to get `fashion-mnist_train.csv` training data set and testing the `fashion-mnist_test.csv` testing data set. All model training processes should be done on the training data, and the results should be based on the testing data.

Fashion-MNIST is a data set by the online store Zalando and contains images from $70,000$ items in total ($60,000$ in the traning set and $10,000$ in the testing data set). Each item is represented as a 28×28 grayscale image associated with a label from 10 classes: '0' for `T-shirt/top`, '1' for `Trouser`, '2' `Pullover`, '3' `Dress`, '4' `Coat`, '5' `Sandal`, '6' `Shirt`, '7' `Sneaker`, '8' `Bag`, '9' `Ankle boot`.

Our **goal** is to do multi-class classification to predict the **class label** in the testing data.

1. **Data Processing and Unsupervised Learning.**

   (a) Compute and report a frequency table of the outcome variable in *both training and testing* data.

   (b) Perform **two clustering algorithms** to the *training data*.
   - Use a systematic approach (e.g., gap statistic, silhouette statistic) to find the optimal number of clusters (if applicable).
   - What is the dominating label in each of these clusters?
   - Do your clusters help to separate the labels? Comment on the results.

2. **Multi-class Classification Model**

   Choose **four** of the methods we discussed in class to do **multi-class** classification. Similar to coding assignment 4, some of the methods were presented for binary classification, but you are free to extend and use them for multi-class classification (e.g. you can use *One-vs-Rest*). Make sure you describe your approach.

   Report the *overall classification error*. Also, please provide enough supporting information, e.g. tables/figures, to demonstrate the fit of the models.

Remarks:

This is a large data set that needs some dditional **pre**-processing before feeding the data to the various algorithms – before parts (1) and (2). These steps are *optional* and depend on the specific methods that you will choose to implement. You are also welcome to apply additional techniques provided that you explain your process and provide necessary references, as needed.

Some examples for additional pre-processing include:

- Each image is represented as a 2D matrix of 28×28 pixels, where each pixel has a value from 0-255. Most of the functions/algorithms that we work with in R/Python expect an 1D feature vector as an input for each sample. So, you need to convert this matrix into a single vector that will consist of $28 times 28 = 784$ features.

- Because of the high-dimensional nature of the data, you could consider some dimensionality reductions methods, such as PCA before certain methods, e.g., before clustering or SVM.

– Make sure that you scale the features for the methods that are sensitive to different scales.

Grading Rubric:

Data Pre-processing: **10 points**
Data Processing and Unsupervised Learning: **30 points**
Multi-class Classification: **30 points**
Report Write-up, structure and presentation of results: **20 points**
Code Submitted: **10 points**