CSCN8000 – Artificial Intelligence Algorithms and Mathematics

Final Exam Part B

Figure 1: Small sample of greyscale elements of the FashionMNIST dataset.

Data Set - Fashion MNIST . . . with a Twist!

This is an image dataset based on publicly available dataset Fashion MNIST:

https://github.com/zalandoresearch/fashion-mnist

As shown in Fig. 1 The dataset is composed of small (28x28 pixels), grey-scale images of clothing items such as shoes, coats, pants, etc. It was created to serve as a direct drop-in replacement for the well known MNIST dataset (http://yann.lecun.com/exdb/mnist/) dataset for benchmarking machine learning algorithms. So the dataset shares the same image size and structure as the original MNIST.

The Twist...

The label was originally an assigned clothing type represented by an integer from 0-9. The training set contains 60,000 examples and the test set 10,000 examples.

For this final exam, the same input features are provided, but the **labels** you will be using will be a **new mystery label**. A new label based on the data was created and calculated to be associated with each entry. The y_train and y_test files provide this modified label which is a category numbered 0-4. What does the number mean? That's for you to figure out through your analysis!

File Descriptions:

- x train.csv - the training set for your model. The training file contains vectors of size 784 representing pixel values of a 28x28 image.
- y train.csv - is the "mystery label", a numeric target obtained using our formulation.
- x test.csv - are the features for the test set to use for final evaluation.
- y test.csv - the "mystery label" for the test data.

Part A: Model Training and Evaluation (25 Points)

In this section, we want to investigate the Bagging and Boosting ensemble methods and determine which method would work best for our dataset. Specifically, the deliverables for the section are as follows (can be done in any order):

- **Choose** a shallow (simple) classification baseline model from the ones mentioned in class after the midterm. A shallow classifier is one that doesn't generate a complex decision boundary. We will use this classifier in the following points. **Include** a comment on why you choose this particular classifier and why is it considered shallow.
- **Train** the chosen baseline model on the training set. Make sure to use 5-fold cross-validation to properly judge the performance of the model on a validation set.
- Utilize the **bagging** method, with the base estimator being the model chosen in the first step, to do a training run on the training set. Make sure to use 5-fold cross-validation to properly judge the performance of the model on a validation set.
- Utilize the **boosting** method, with the base estimator being the model chosen in the first step, to do a training run on the training set. Make sure to use 5-fold cross-validation to properly judge the performance of the model on a validation set.
- *Note: to avoid long computational time, reduce the number of estimators in both methods to 10 or 20. You can check the sklearn.ensemble package for more information.*
- **Validate** the trained baseline model, trained bagging and boosting models on the testing set and compare their performance. **Include** a comment on which method worked better and a possible reasoning behind the observations.

Part B: Guessing the Mystery Label (25 Points)

In this part, we will try to use Dimensionality Reduction and Clustering techniques to guess what the mystery labels in the dataset are. Specifically, the deliverables for the section are as follows (can be done in any order):

- **Train** a simple neural network (MLP) with 3 hidden layers having the following number of neurons $[256,128,64]$ on the training set. Make sure to normalize the data beforehand with z-score normalization.
- **Utilize** the uploaded helper code in "helper.py" to generate an embeddings matrix representing the output of the last hidden layer in the MLP for **the test dataset**.

- **Carry out** PCA and LDA to reduce the dimensionality of the *generated embeddings* to 2. **Produce** a plot of the data in two dimensions for PCA and LDA, using easily distinguishable colours and markers to indicate the labels of each datapoint. **Include** a comment on any interesting patterns you see in the two plots.
- **Carry out** K-Means on the *generated embeddings* with 5 clusters and **visualize** the results by using the resulting clusters as alternate colour mappings for the *PCA* plot above.
- Based on the results seen in the plots, **can you guess** what are the labels for the given dataset (what each label number represents in terms of the category of clothes)? It might also help to list out a random selection of data entries (the original images) for each cluster and their label value to help understand the patterns each cluster might represent.
    - *Hint: to do this, you can reshape the images to (28,28) and use the .imshow() function in matplotlib to plot the images.*

Organization Criteria (5 Points)

1. Provide an organized notebook at the end with clear section, markdown comments and printed outputs. Make sure no errors are printed in the final submission.

Deliverables

1. Include all your findings and task solutions in one Jupyter notebook (.ipynb) that shows all the printed cell outputs. Prepare a html version (.html) of the notebook file. Both files should be named as follows: [Full Name]_[Student ID]_[Section Number]_Final_Part_B.[html/ipynb].
2. Submit both .html and .ipynb files on eConestoga in the Final Exam Part B section under the Assignments section.