<u>Assignment 2: Supervised Learning</u>

Szymon Pawlica

R00187226

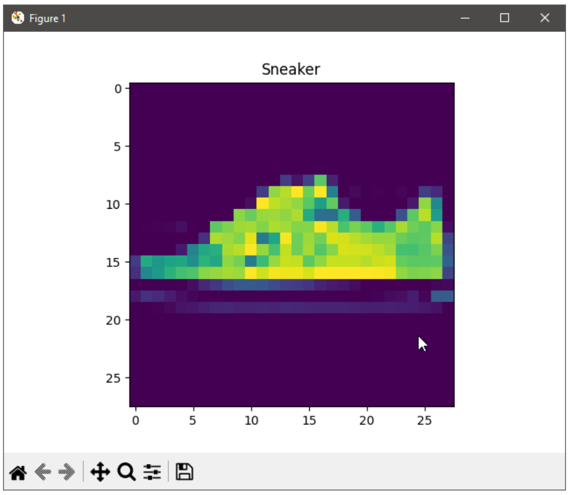
I initially misunderstood the question and haven't implemented gamma for SVM rbf.

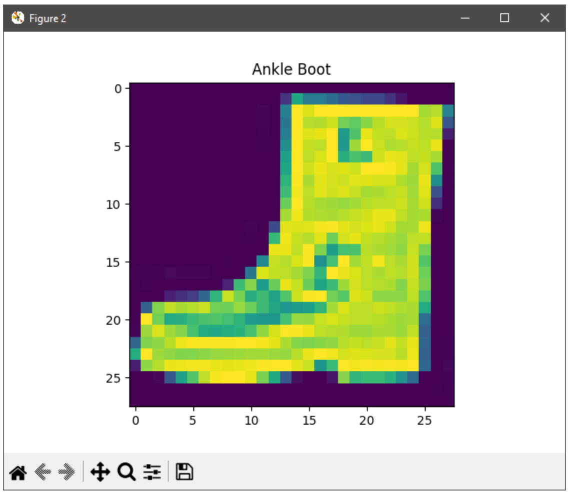
Sample outputs are under each line requiring one

Task 1:

Lines [18-41]

Separate Labels from Feature Vectors, count the amount of sneaker samples and ankle boot samples. Display a random image containing a sneaker and containing an ankle boot.





Task 2 – Task 6:

Lines [44-168]

[45] Different samples sizes, I vary the % of samples used in *model_selection.train_test_split.*

[46] Classifiers dictionary to hold runtimes for different samples sizes.

[64] Split data into training and test subsets, using *model_selection.train_test_split.*

[91-95] Processing time required for training, by taking the time before and after *clf.fit.*

[98-102] Processing time required for prediction, by taking the time before and after *clf.predict.*

[107] Accuracy score. *Example for Decision_Trees:*

Split 1: Processing Time required for Training 0.3263s

Split 1: Processing Time required for Prediction 0.0020s

Split 1: Accuracy Score - 0.9143

Split 2: Processing Time required for Training 0.4554s

Split 2: Processing Time required for Prediction 0.0010s

Split 2: Accuracy Score - 0.9021

Split 3: Processing Time required for Training 0.3603s

Split 3: Processing Time required for Prediction 0.0020s

Split 3: Accuracy Score - 0.9129

Split 4: Processing Time required for Training 0.2853s

Split 4: Processing Time required for Prediction 0.0010s

Split 4: Accuracy Score - 0.9014

Split 5: Processing Time required for Training 0.3573s

Split 5: Processing Time required for Prediction 0.0010s

Split 5: Accuracy Score - 0.9014

[113] Confusion Matrix.

True Positive: 45.7857

False Positive: 5.7143

True Negative: 44.3571

False Negative: 4.1429

[119-132] Minimum, maximum and average of training time per sample, prediction time per sample and prediction accuracy, by finding the *min*, *max*, *mean* of the corresponding lists for each K.

Minimum Processing Time Required for Training 1.0090839295160203e-05s

Maximum Processing Time Required for Training 1.8187363942464192e-05s

Average Processing Time Required for Training 0.0556s

Minimum Processing Time Required for Prediction 4.90063712710426e-07s

Maximum Processing Time Required for Prediction 7.85873049781436e-07s

Average Processing Time Required for Prediction 0.0023s

[154] Find the best mean prediction accuracy for each classifier from all sample sizes, by finding the *max* scoring prediction accuracy for current classifier.

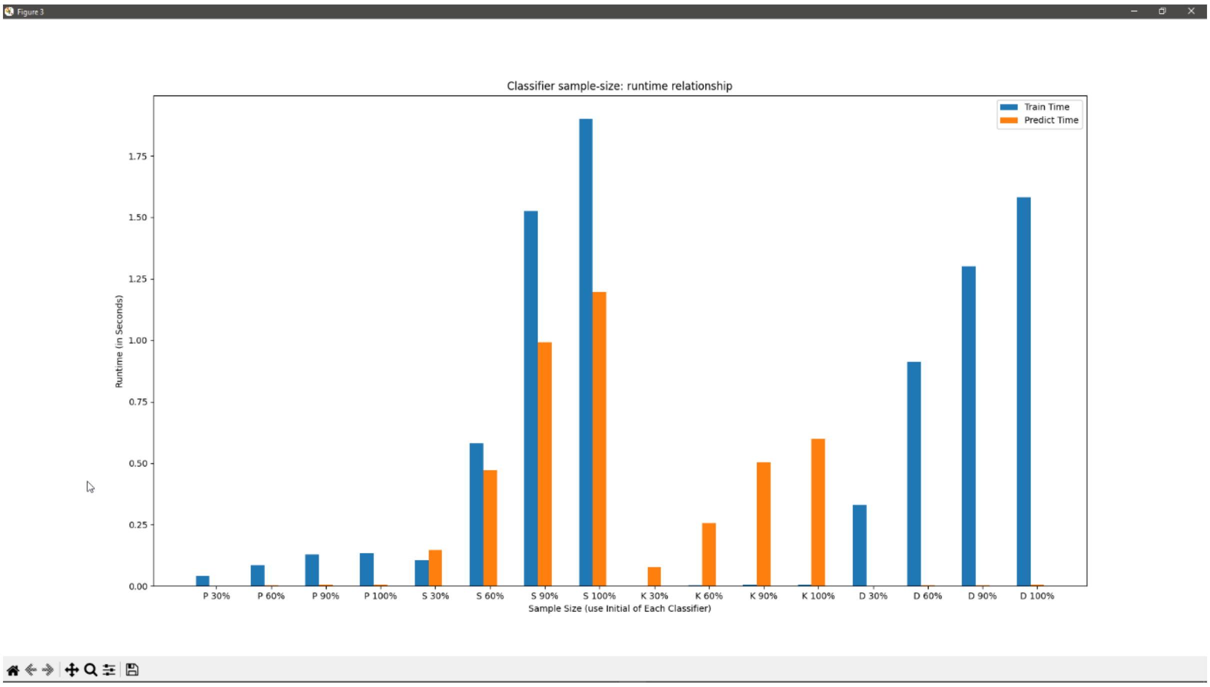
The best Achievable Mean Prediction for perceptron is 0.9466

The best Achievable Mean Prediction for SVM is 0.9675

The best Achievable Mean Prediction for k_nearest_neighbour is 0.9583

The best Achievable Mean Prediction for decision_trees is 0.9257

[158-168] Plot the relationship between sample size and runtime for each classifier using the dictionary created on [46].



Task 5:

I implemented going through K for all the classifiers instead of just for K neighbours which causes the execution time of the script to be very long.

[74, 86] Use different values for K neighbours

[136] Find the best K by finding the *max* scoring K prediction accuracy

Best K = 6, 4, 4, 6 for sample sizes 30%, 60%, 90%, 100%

Task 7:

As the sample size increases all the classifiers have longer runtimes which is to be expected. However, classifiers like perceptron and decision trees have almost instant prediction, whereas K Neighbours has almost instant training and SVM takes longer in both training and prediction runtimes than any of the other classifiers.

- SVM is the most accuracy classifier while also having the longest training and prediction runtimes.
- K Neighbours achieves the second highest mean accuracy score while also not taking too long to execute.
- Perceptron accuracy is less than both SVM and K Neighbours however it has a very short runtime for training and prediction.
- Decision trees is the worst classifier for this task as it takes a long time for training while producing the lowest mean accuracy.