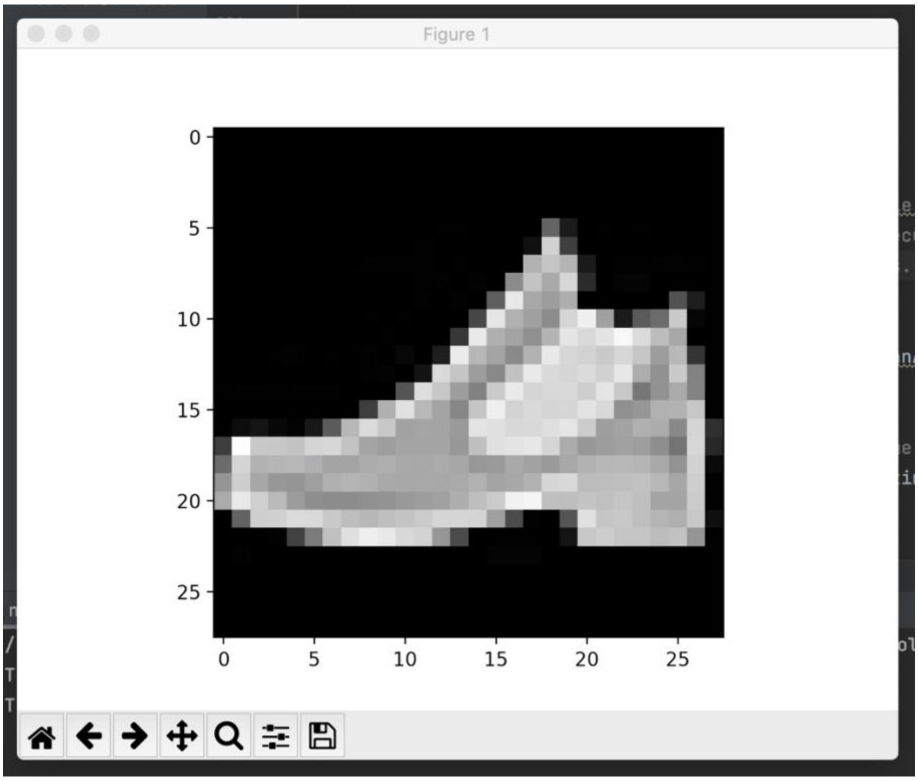
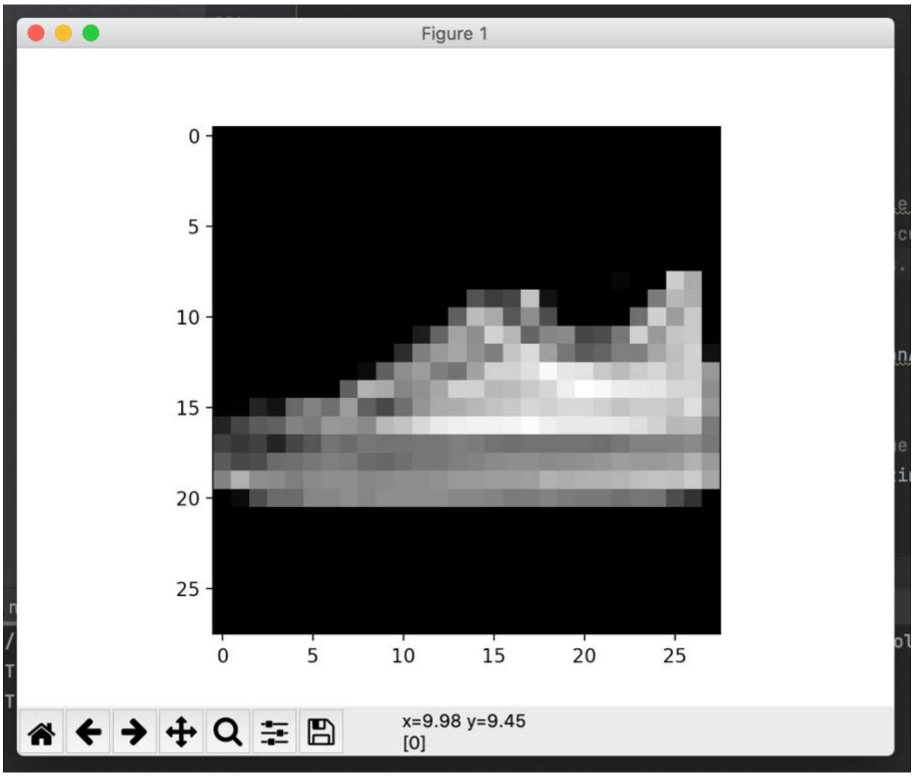
David Irwin
R00109532

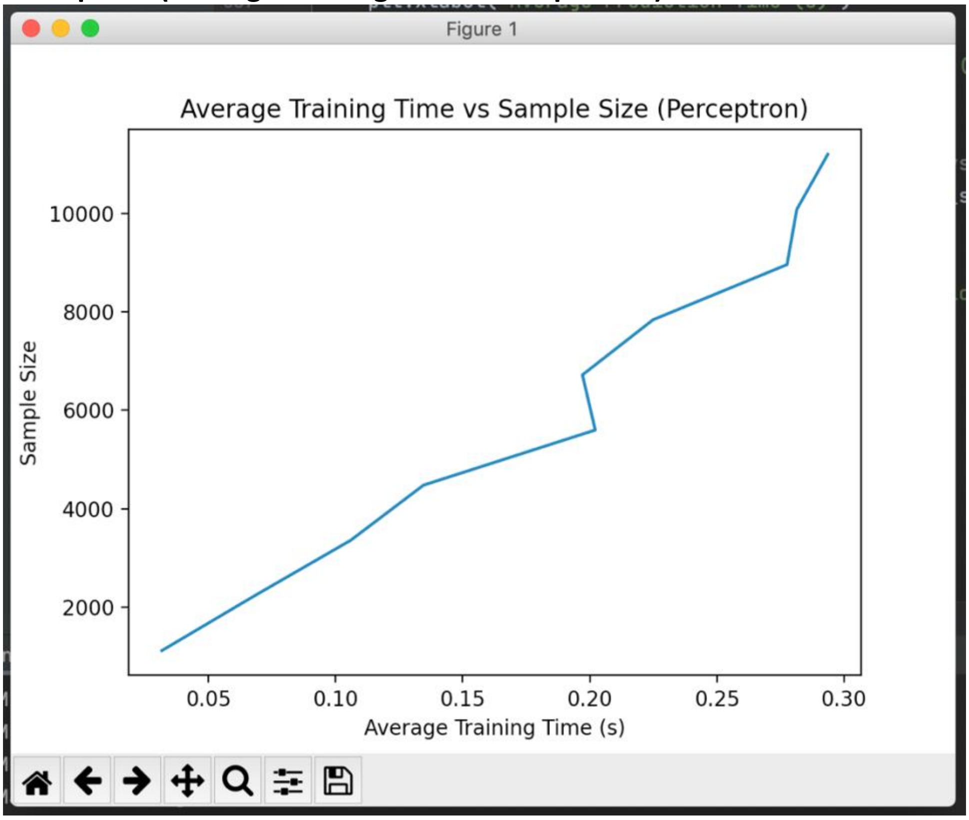
Task 1



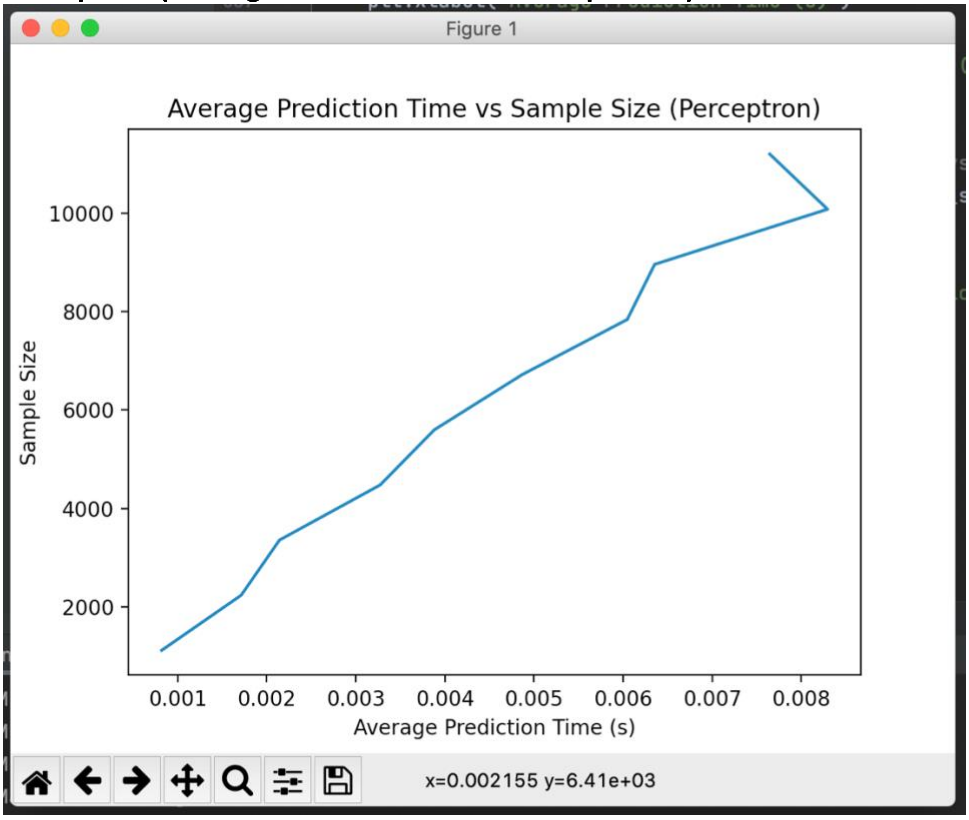


Task 3

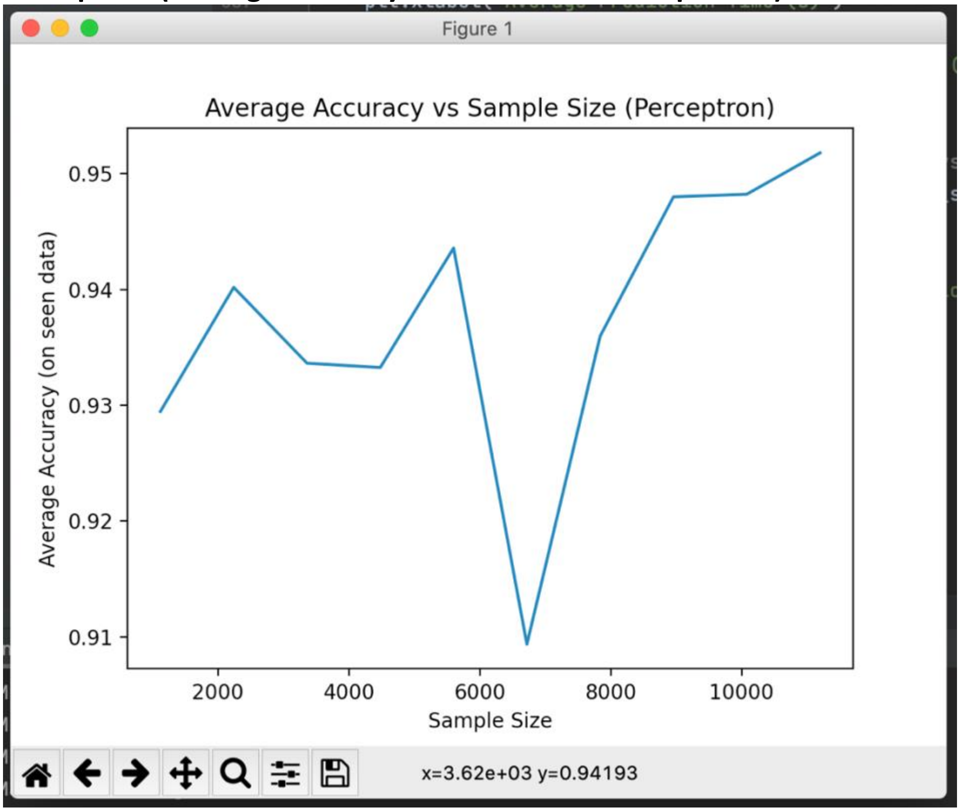
Perceptron (Average training time vs sample Size)



Perceptron (Average Prediction time vs sample Size)



Perceptron (Average accuracy on seen data vs sample size)



Accuracy of model (on unseen data, sample size = 11200)

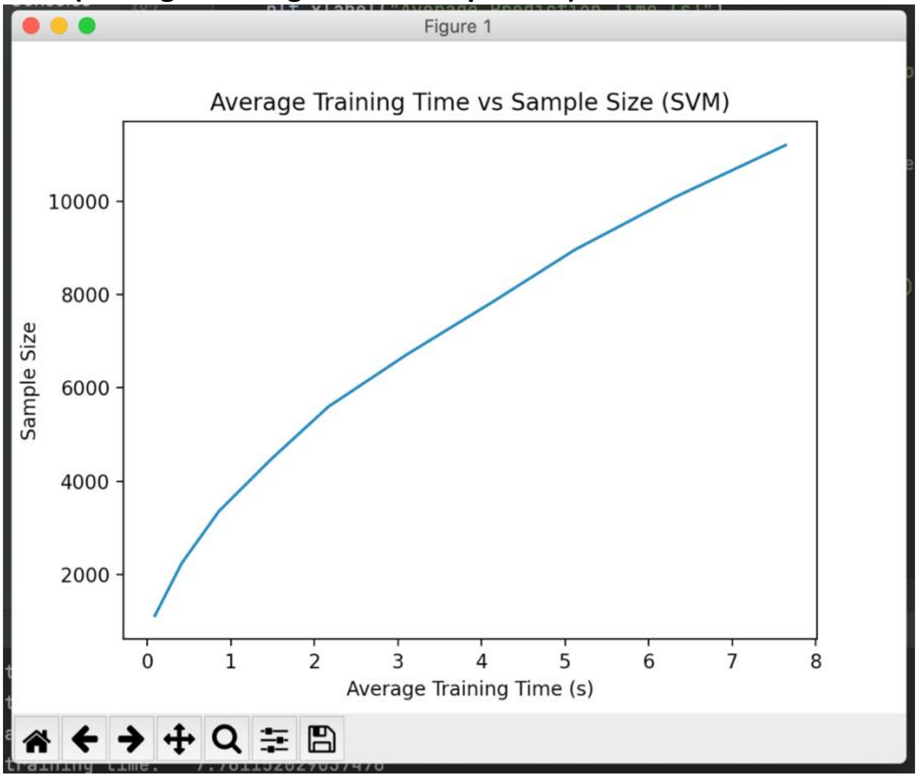
```
Accuracy (on unseen data):  0.9592857142857143
```

Training times when sample size = 11200

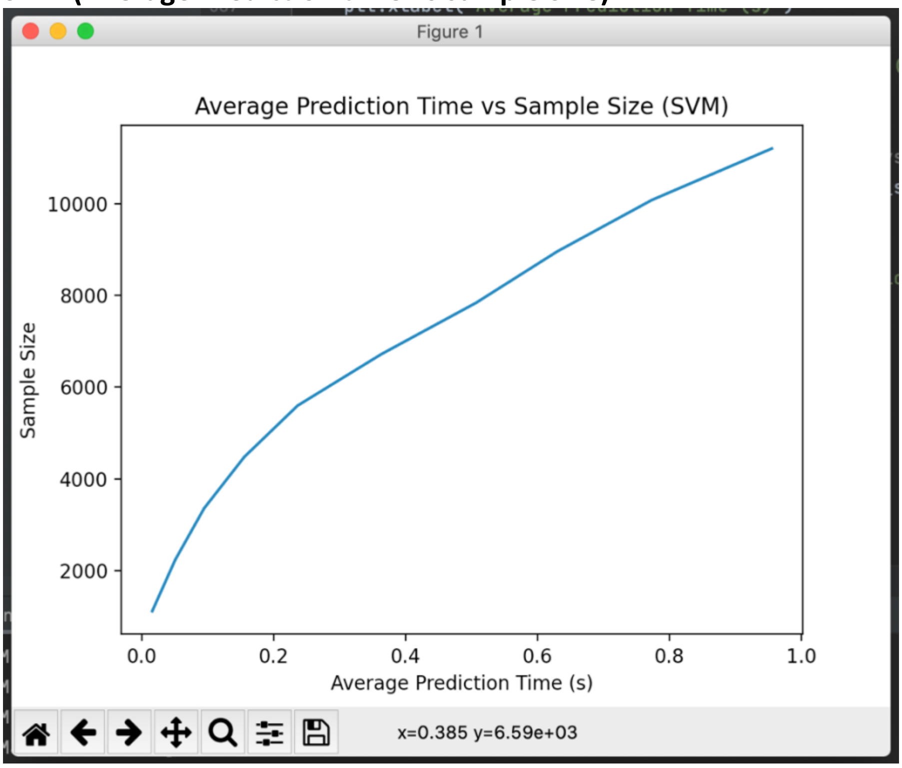
```
Minimum training time:   0.19684505462646484
Minimum testing time:    0.00703191757202148
Minimum accuracy:        0.9321428571428572
Maximum training time:   0.593667030334472
Maximum testing time:    0.01304388046264648
Maximum accuracy:        0.9696428571428571
Average training time:   0.31124613285064695
Average testing time:    0.00795044898986816
Average accuracy:        0.9517857142857142
```

Task 4

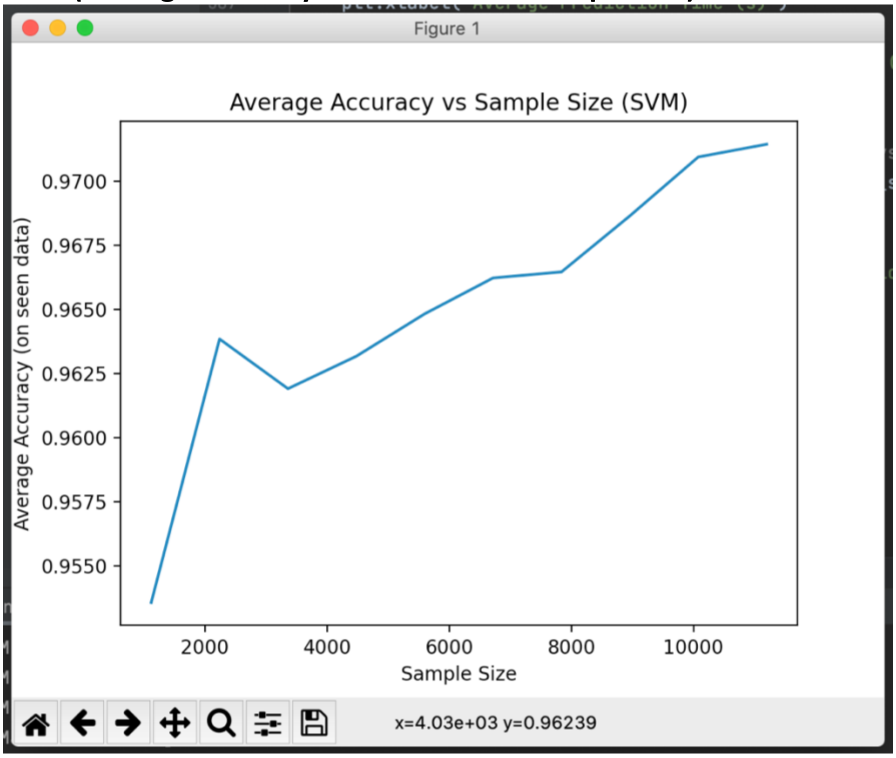
SVM (Average training time vs sample Size)



SVM (Average Prediction time vs sample Size)



SVM (Average accuracy on seen data vs sample size)



Best gamma value (determined by highest mean accuracy on seen data when sample size = 3000)

```
Best gamma value:  5e-07

********************************
```

Accuracy of model (on unseen data, sample size = 11200)

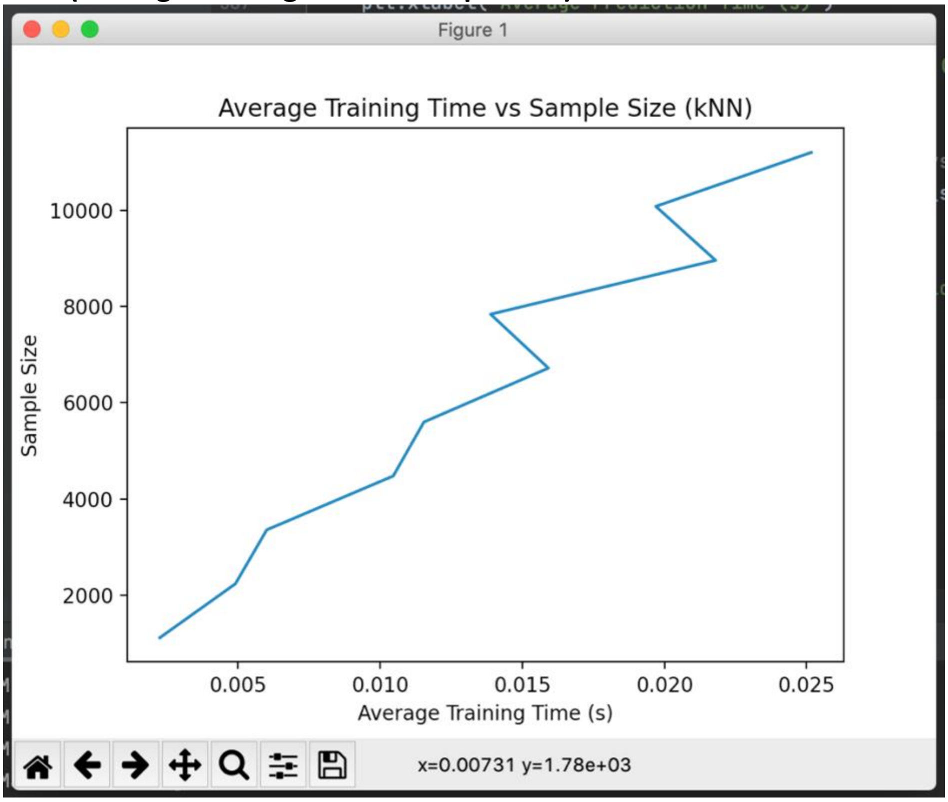
```
Accuracy (on unseen data):  0.9739285714285715
```

Training times when sample size = 11200

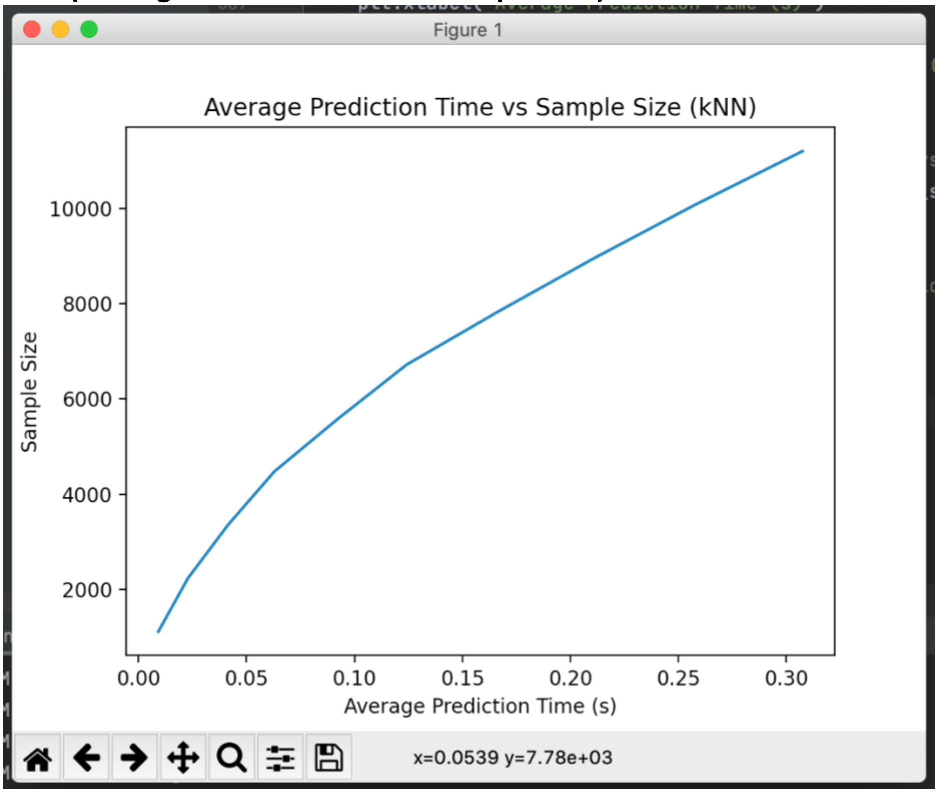
```
Minimum training time:   7.537694215774536
Minimum testing time:    0.9308781623840332
Minimum accuracy:        0.9669642857142857
Maximum training time:   7.873740911483765
Maximum testing time:    1.0243539810180664
Maximum accuracy:        0.9794642857142857
Average training time:   7.725786089897156
Average testing time:    0.9703830003738403
Average accuracy:        0.9714285714285713
```

Task 5

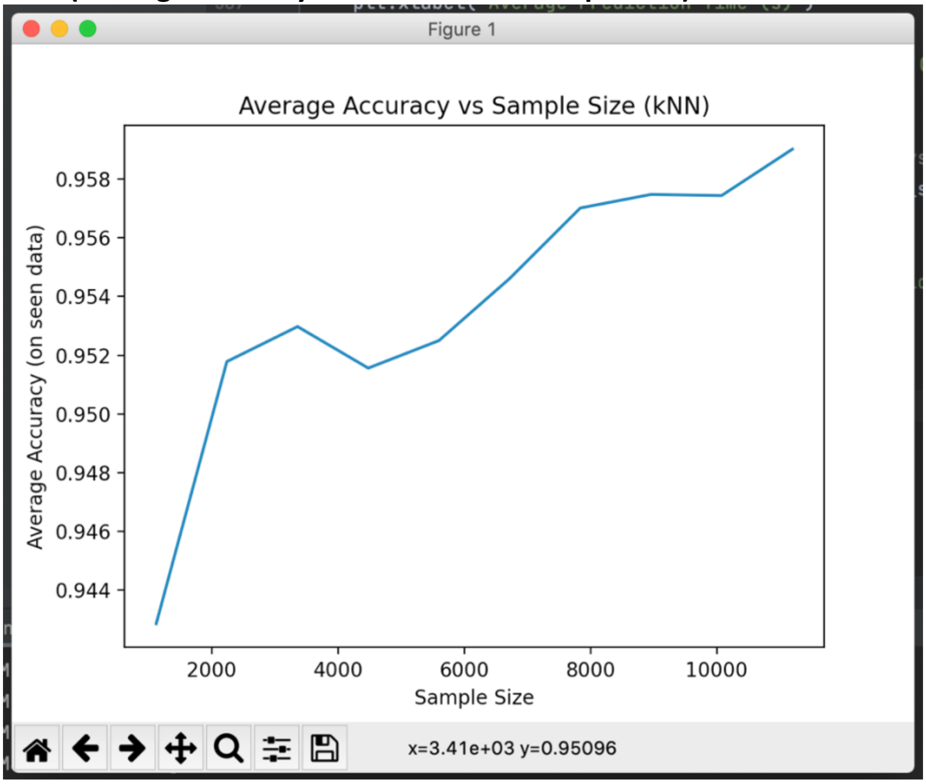
kNN (Average training time vs sample Size)



kNN (Average Prediction time vs sample Size)



kNN (Average accuracy on seen data vs sample size)



Best k value (from a choice of 1 to 10) (determined by highest mean accuracy on seen data when sample size = 3000)

```
Best k value:  7
```

Accuracy of model (on unseen data, sample size = 11200)

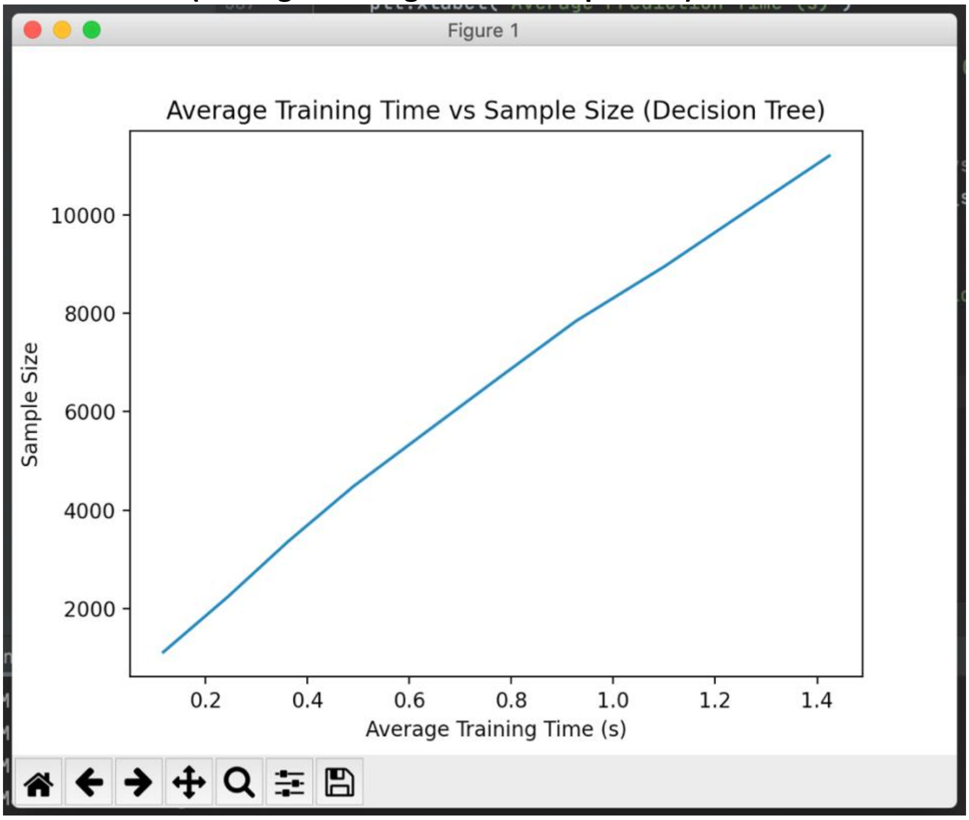
```
Accuracy (on unseen data):  0.9628571428571429
```

Training times when sample size = 11200

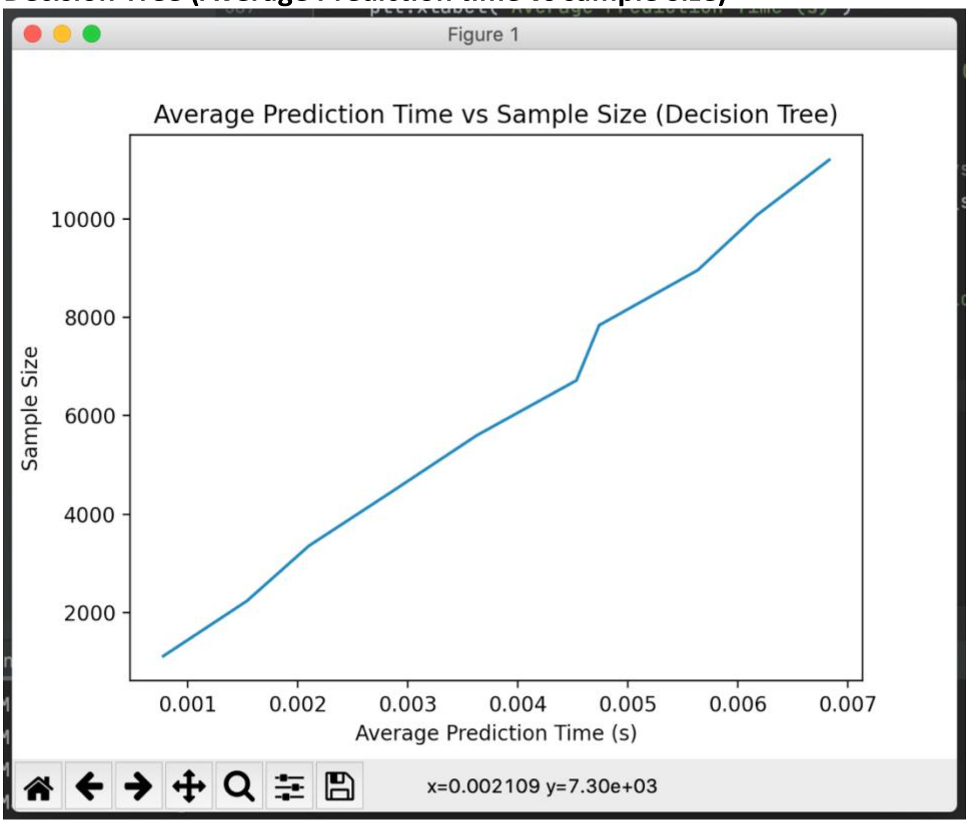
```
Minimum training time:   0.024376869201660156
Minimum testing time:    0.3010427951812744
Minimum accuracy:        0.9526785714285714
Maximum training time:   0.027286052703857422
Maximum testing time:    0.3645608425140381
Maximum accuracy:        0.96875
Average training time:   0.025330209732055665
Average testing time:    0.32100620269775393
Average accuracy:        0.9590178571428571
```

Task 6

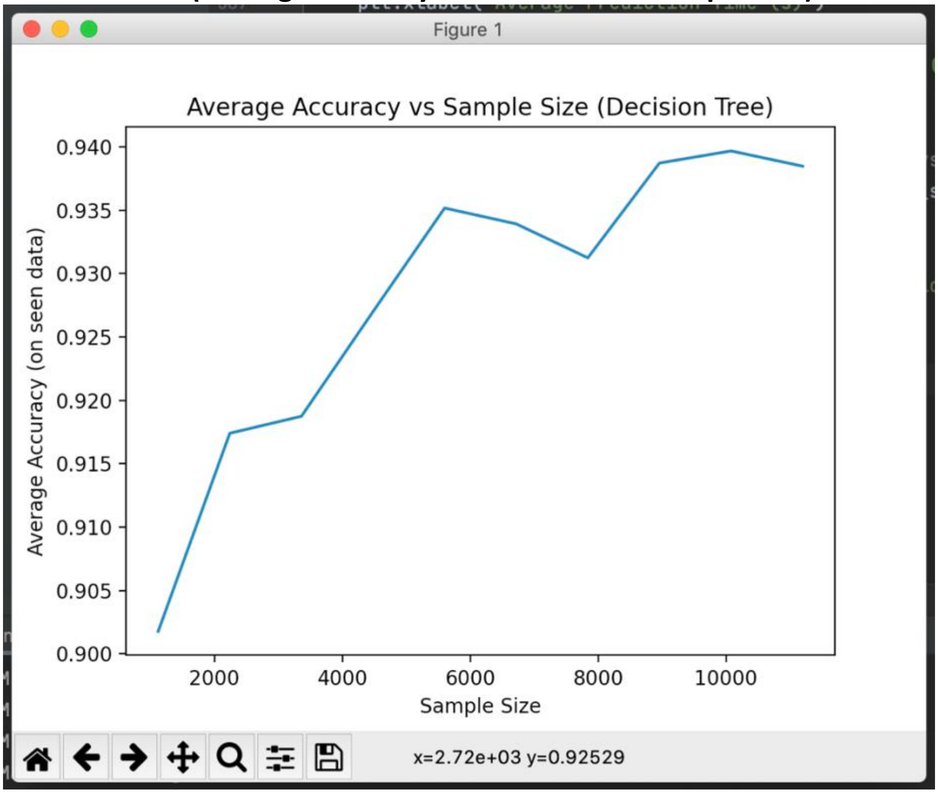
Decision Tree (Average training time vs sample Size)



Decision Tree (Average Prediction time vs sample Size)



Decision Tree (Average accuracy on seen data vs sample size)



Best depth value (from a choice of 1 – 10) (determined by highest mean accuracy on seen data when sample size = 3000)

```
Accuracy (on unseen data):  0.915714285714285
Best d value:  7
****************************************
```

Accuracy of model (on unseen data, sample size = 11200)

```
Accuracy (on unseen data):  0.9414285714285714
```

Training times when sample size = 11200

```
Minimum training time:   1.2494759559631348
Minimum testing time:    0.006673336029052734
Minimum accuracy:        0.925
Maximum training time:   1.294084072113037
Maximum testing time:    0.007397890090942383
Maximum accuracy:        0.9526785714285714
Average training time:   1.267964768409729
Average testing time:    0.006956148147583008
Average accuracy:        0.9401785714285713
```

Task 7
Trends

Perceptron
Average training time seems to increase largely linearly to the sample size. It can be inferred that the average training time is directly correlated to the sample size. Average prediction time seems to increase largely linearly to the sample size. It can be inferred that the average training time is directly correlated to the sample size.

I think this is the case because as the dataset grows, the number of calculations required also grows proportionately.

Average accuracy on seen data also seems to increase linearly. Inferring that average accuracy has a direct correlation with sample size.

I think this is the case because as the dataset grows, it has more x-y pairs to learn from.

Note the dip between 6,000 & 8,000. My assumption here is that because I take a sample randomly, I believe that maybe there is an unbalance between the number of sneaker rows in the sample and the number of ankle boot rows in the dataset. And that this imbalance means that one label gets predicted wrong a lot. This dip is also present in the other models (except from the decision tree).

SVM
After sample equals 4000 **average training time** seems to increase largely linearly to the sample size. It can be inferred that the average training time is directly correlated to the sample size.

After sample equals 4000 **Average prediction time** seems to increase largely linearly to the sample size. It can be inferred that the average training time is directly correlated to the sample size.

I think this is the case because as the dataset grows, the number of calculations required also grows proportionately.

Average accuracy on seen data also seems to increase linearly after sample size 3,000. Inferring that average accuracy has a direct correlation with sample size.

kNN
Seems to follow the same trends as the previous models. However, it’s accuracy sharply increases going to 2,000.

Decision Tree

Also seems to follow the same trends as model 2 & 3. This model however is more linear than the rest for training & testing time.

Classifier Ranking

The accuracy of the four models on unseen data is as follows:

Model	Accuracy
Perceptron	0.9593
SVM	0.9739
kNN	0.9629
Decision tree	0.9414

The average time to train & test the four models on seen data is as follows (sample size = 11400):

Model	Train	Test
Perceptron	0.3112	0.0080
SVM	7.7258	0.9703
kNN	0.0253	0.3210
Decision tree	1.2680	0.0070

As the accuracy seem to be very similar, I need to look at the training time / testing time to see the trade-off. I will rank the models in terms of lowest training / testing times.

1. Perceptron
2. kNN
3. Decision Tree
4. SVM

One could argue that because the training / testing times of the Perceptron & kNN are very similar, that kNN could be ranked as number 1. However, I decided to stick with the above rankings.

Note that SVM has the highest accuracy, yet I ranked it lowest. This is because for the small gain in accuracy, it takes way too much time compared to the other models. However, the SVM can have much better training / testing times in real life as it can very effectively utilise Nvidia GPUs for training which will make it much faster. All above models were completely CPU bound for this assignment.