# CS487 - HW3

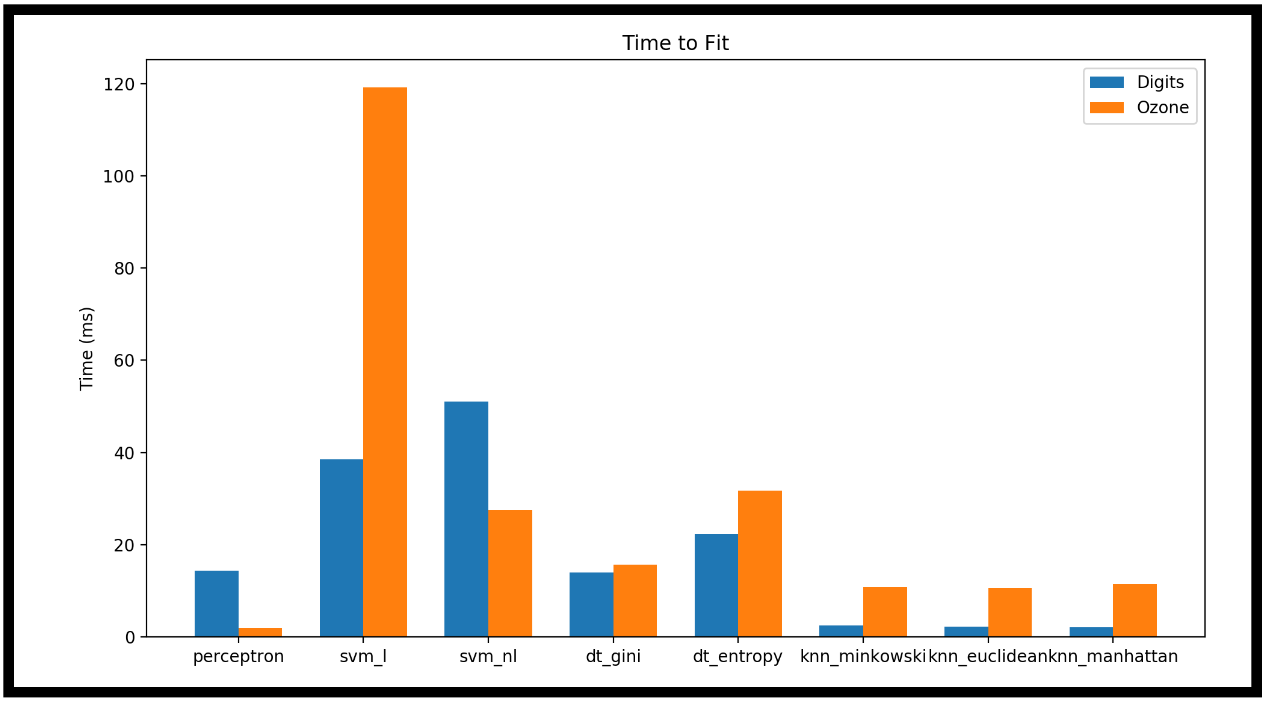
Cyrus Baker
March 1st, 2020

## Datasets

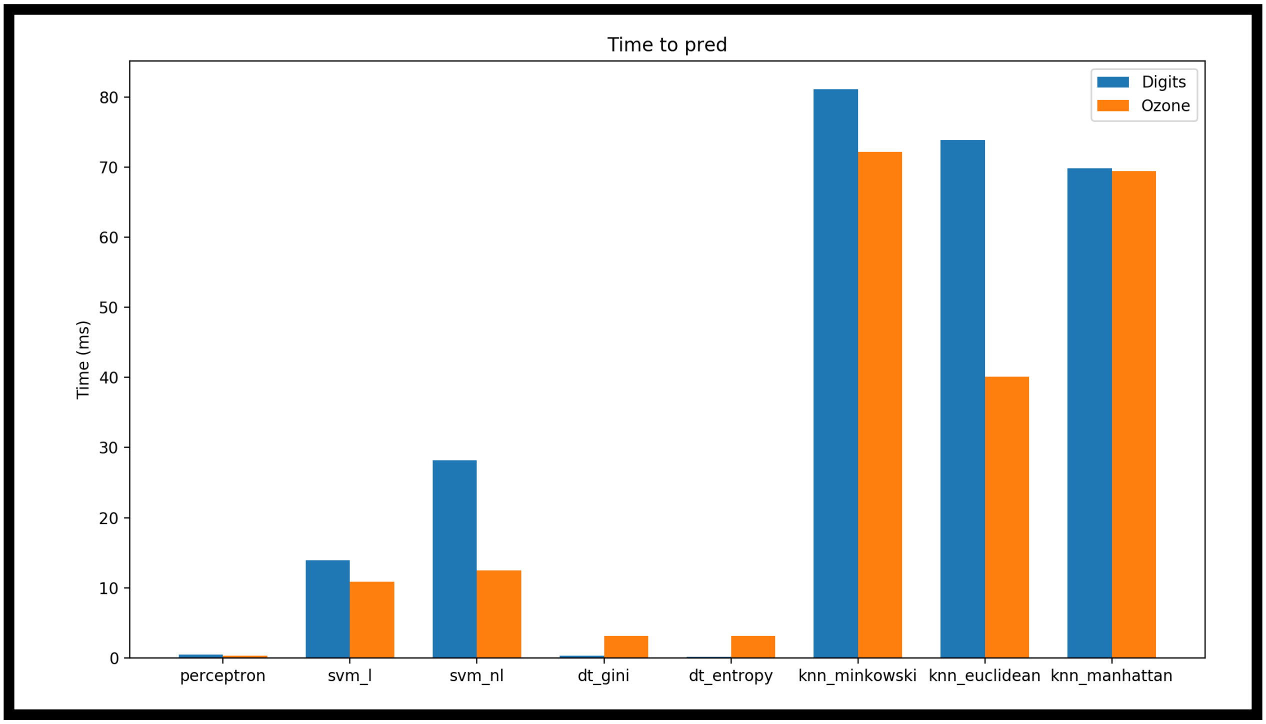
For this assignment I experimented with different classifiers using two datasets. The required dataset is the Digits dataset from the Sklearn library and the other dataset I chose is the Ozone Level Detection dataset from the UCI Repository. The Ozone dataset contains a column for dates which I removed because it doesn't add any meaningful information to my classifier. The Ozone dataset also contained missing or improper values which I had to cleanse from the dataset before running the classifiers.

## Analysis of Classifiers

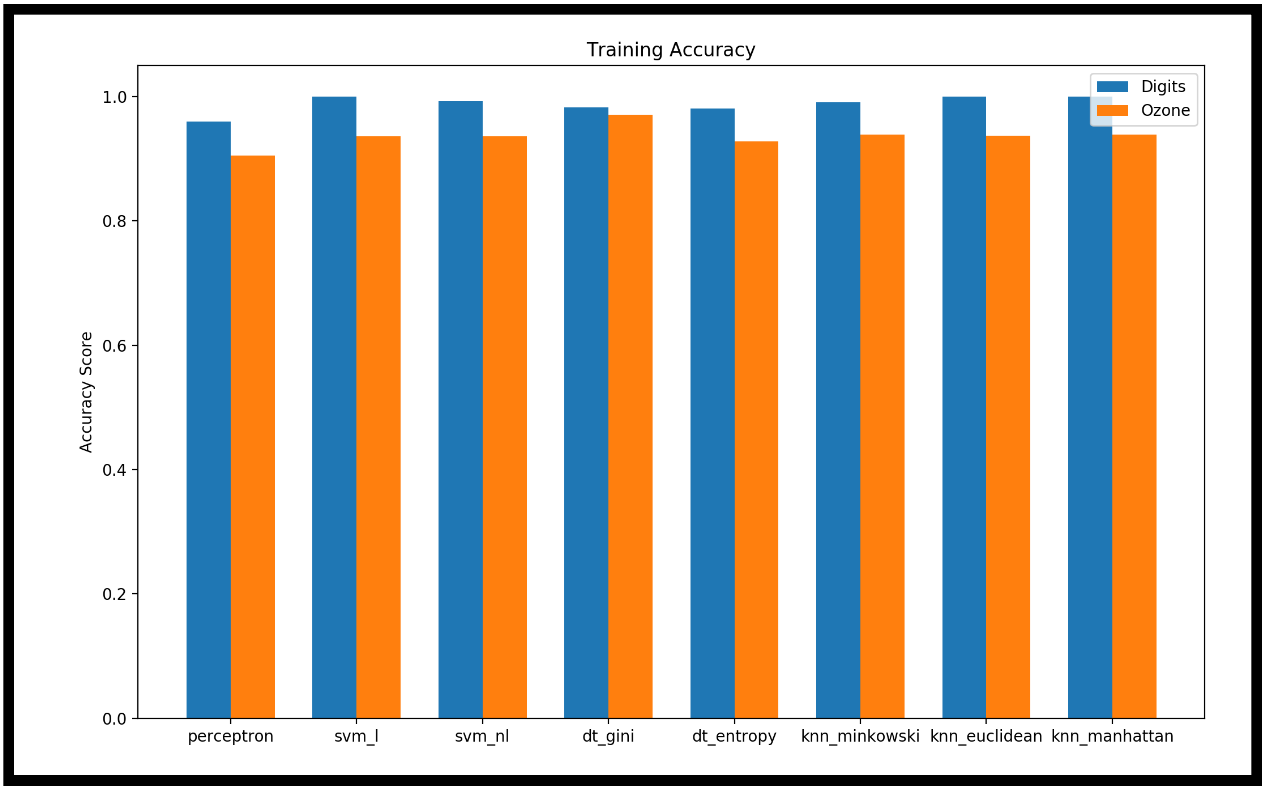
The following graphs illustrate the differences between the classifiers for the Digit dataset from the Sklearn library and the Ozone Level Detection Dataset from the UCI Repository.



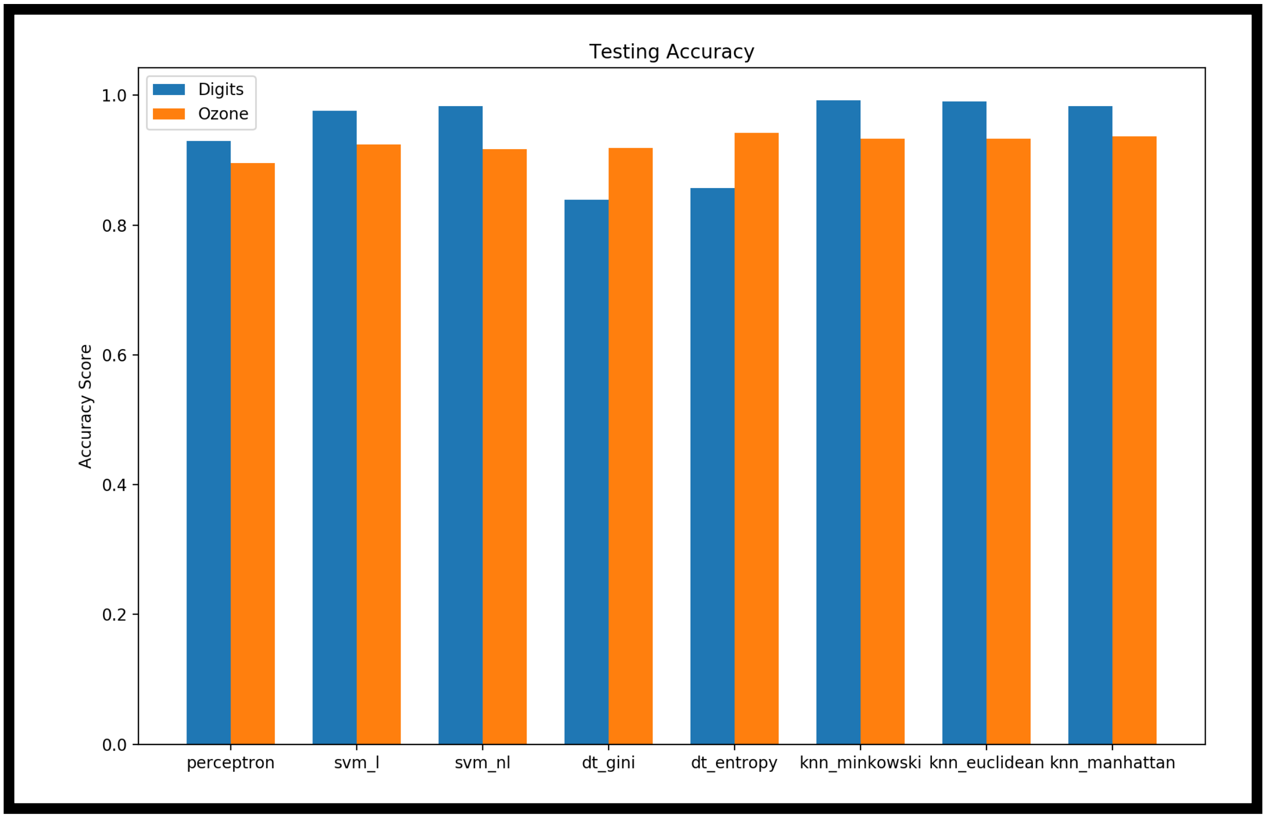
From the graph we can see that the perceptron and knn classifiers had the fastest runtime for training while the linear support vector machine took the longest time.



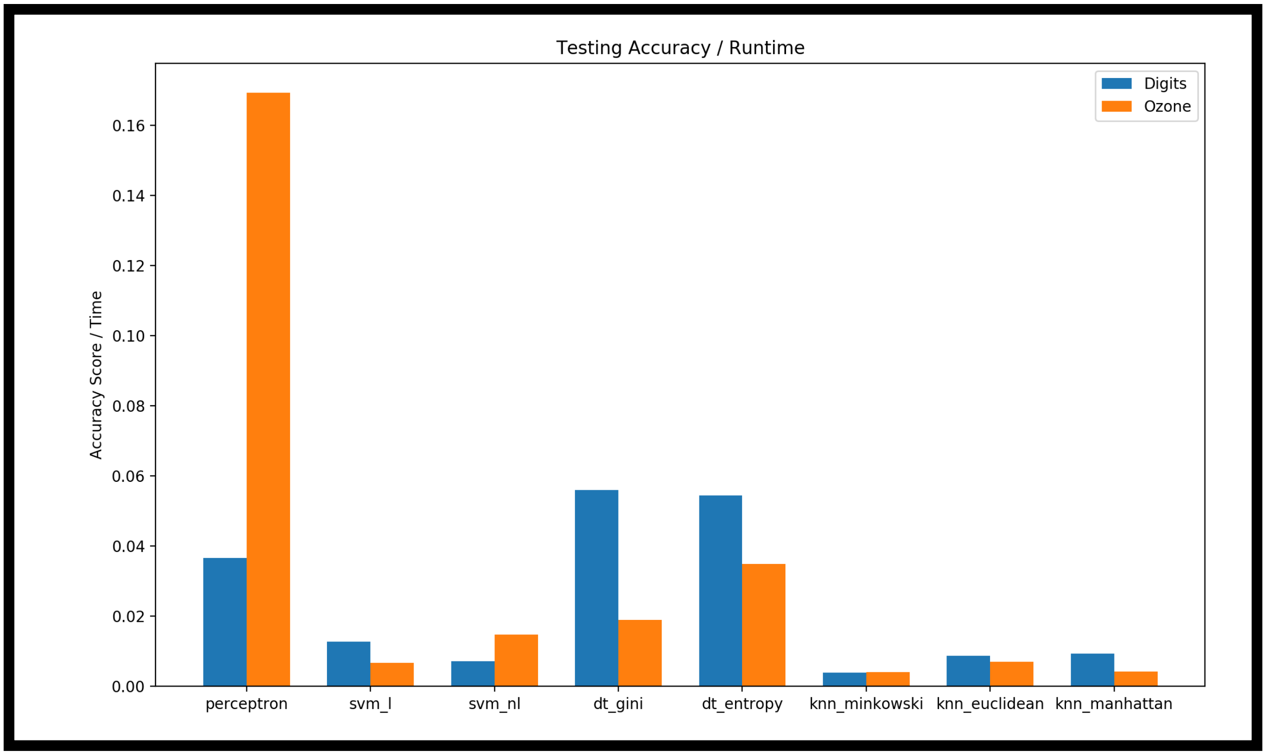
However, the runtime for testing knn grew to be much longer than the other classifiers but the runtime for perceptron remained very short.



The training accuracy for each classifier was very high for both datasets. But the important measurement is the test accuracy.



All of these would be good candidates for classifiers for these datasets but I found that the best classifier in terms of accuracy is knn_euclidean for both datasets. For the digits dataset I found that the optimum number of neighbors is 5 and for the ozone dataset the optimum number of neighbors is 9.



In order to get a sense of how efficient each classifier is I plotted the accuracy over the total runtime. Although knn had the highest accuracy score it seems that for the Ozone dataset the best classifier might be perceptron and for the Digits dataset it might be a decision tree.

# DecisionTreeClassifier

Two strategies that the DecisionTreeClassifier implements to pre-prune or post-prune the tree are max_depth and min_samples_leaf.

- max_depth is the longest path allowed from the root to any given leaf in the tree. (line 88 **scikit-learn/sklearn/tree/_classes.py)**
- min_samples_leaf is the minimum number of samples required for each leaf node. (line 90 **scikit-learn/sklearn/tree/_classes.py)**