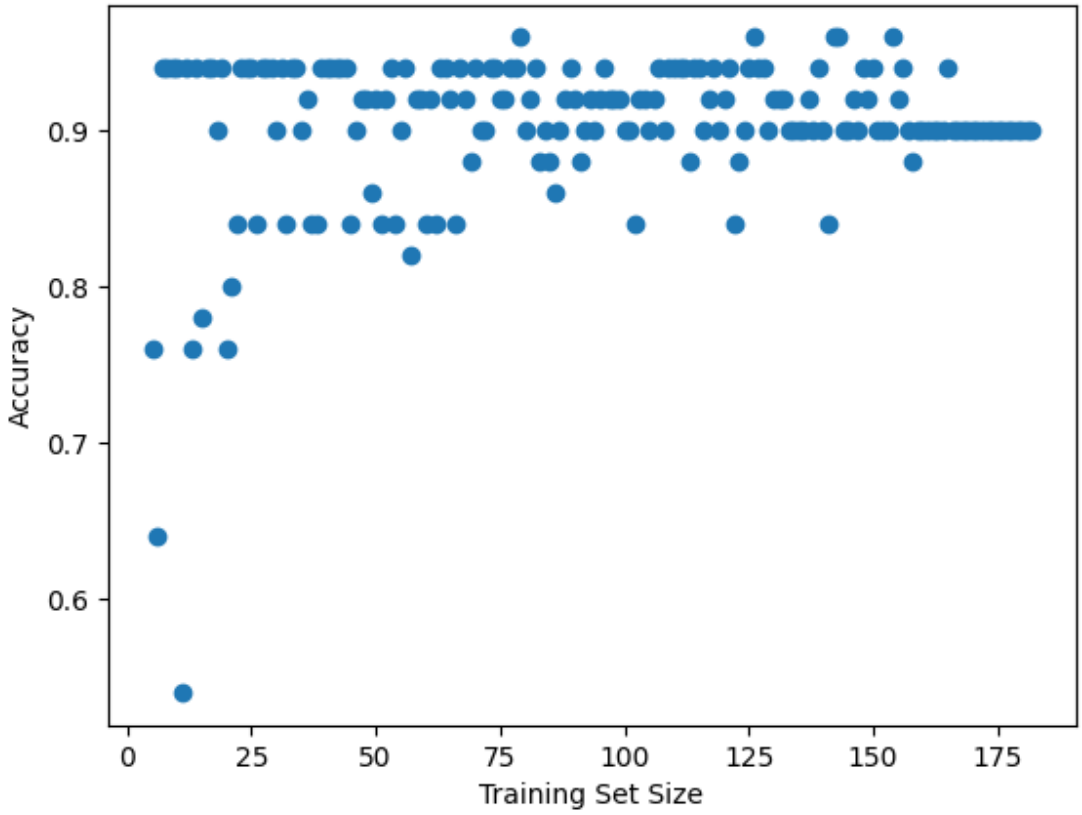
Vivian Feng
Random Forest Classification
Artificial Intelligence
Mr. Eckel

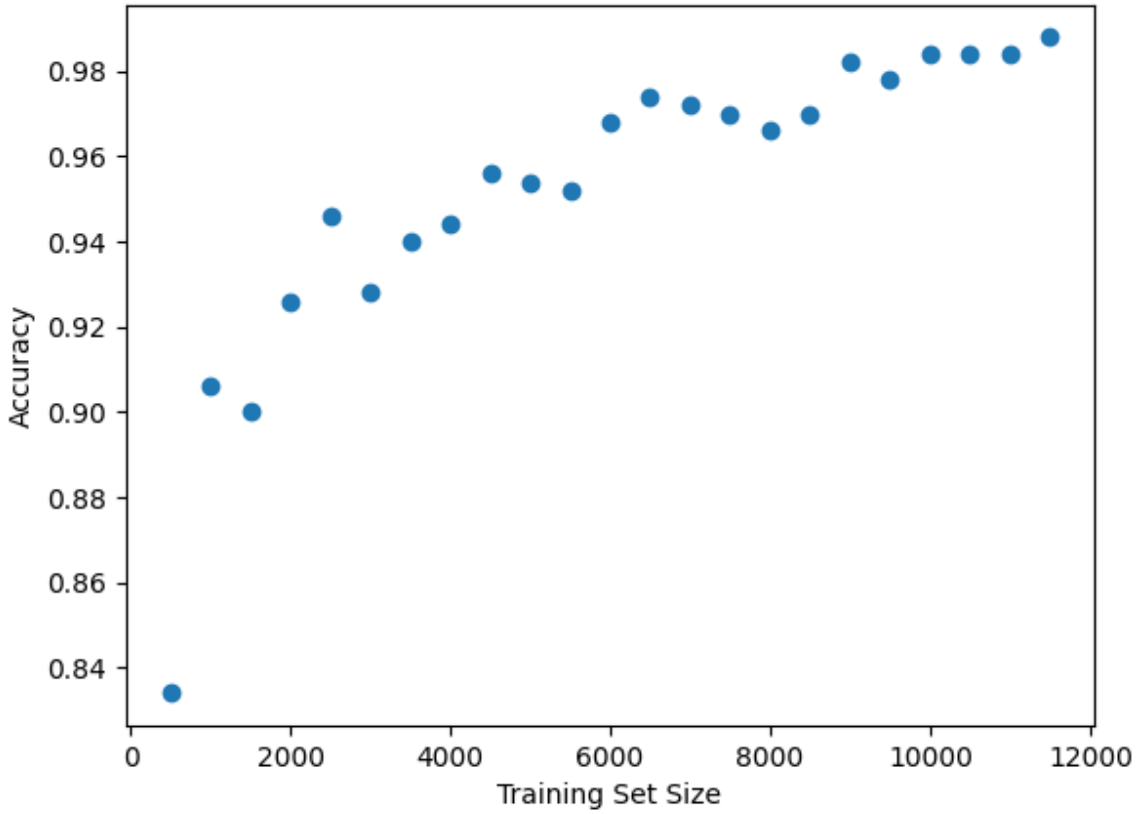
Random Forest algorithm sketch - I used the same code I used to generate decision trees, except I created 10 decision trees by randomly sampling a set of 1/10 the training set data 10 times.
Results for house-votes-84.csv leaving out voting records with missing data (denoted by ?)
Size of tree - 18
Number of trees: 10
Accuracy: 0.96
Chart with the house-votes-84.csv data from Decision trees part 2:



- The accuracy using a random forest of size 10, each tree built with a random sample size of 18 (which is 1/10 the size of the entire training pool), is comparable to that using a data set with a larger sample size and substantially better than a training set of size 20.

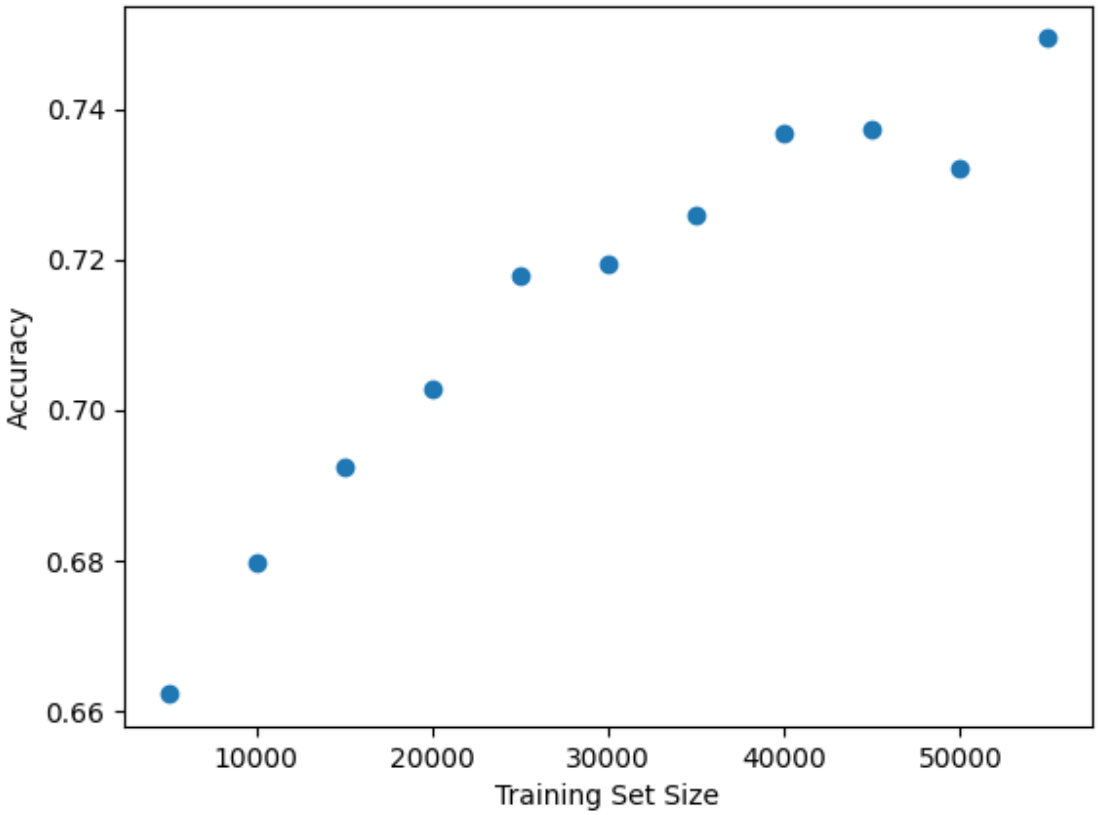
Results for nursery.csv:
Size of tree - 1246
Number of trees: 10

Accuracy: 0.93



- The accuracy of the random forest on a test set of 500 is substantially higher than constructing a single tree out of 1000 training set data points. Again, the trees are built from 10 different randomized samples with a size of 1/10 the total size of the training pool. The results are less accurate than the entire training set (at around 12000), because the nursery school likely followed a flowchart procedure which more data points is able to account for. However, the runtime is faster than using a training set of 12000, which is an advantage.

Results for connect-four.csv:
Size of tree - 6055
Number of trees: 10
Accuracy: 0.779



- Accuracy with 10 random trees is marginally higher than a training set of 55000. Results are mainly the same, but runtime is substantially less because a tree likely grows exponentially. Random forest classification also helps avoid overfitting, so it beats a single tree in terms of practicality.