Supervised Learning - Analysis

Two classification problems

To select two classification problems and analyze their behaviors in different learning algorithms, my criteria of picking the problems is that they should be distinguished from each other in terms of attributes and classes and training example sizes, and also interesting.

Problem 1: Speech Emotion Recognition

Speech Emotion Recognition is a task of recognizing human's emotion from speech regardless of the speech contents. We have being always want to achieve the natural communication between human and machine just like the communication between humans. After years of research, we are able to converting the human speech into a sequence of words. However, despite the great progress made in speech recognition, we are still far from having a natural interaction between man and machine because the machine does not understand the emotional state of the speaker[1]. That's why I want to pick the problem of speech emotion recognition(SER). It's important to let the machine have enough intelligence to understand human emotion by voice.

Problem 2: Forest Cover Type

Predicting forest cover type is learning the forest cover type classes and it's cartographic variables, and predict the cover type in the future. The forest in this study are wildness area with minimal human-caused disturbances. To do inventory of natural resources in a wild area can be very difficult but also really important. It helps to observe ecological changes and identify the type of biosphere within the area. Generally, cover type data is either directly recorded by field personnel or estimated from remotely sensed data. Both of these techniques may be prohibitively time consuming and/or costly in some situations[2]. Thus we can use the predictive forest cover type models to obtain such data efficiently.

Data Processing

Speech Emotion Recognition

The rawest datasets I started with is a collection of 3 seconds audio files in which different actors repeat similar sentences with different emotions. Then I processed the audios with python audio analyzing library Librosa, and extracted three features:

MFCC (Mel Frequency Cepstral Coefficient, represents the short-term power spectrum of a sound).
Chroma (Pertains to the 12 different pitch classes).
Mel (Mel Spectrogram Frequency).

Each feature provides some attributes for further classification. The challenge for this classification problem is, the datasets are not very large (around 1300), while there are 180 attributes and 8 classes. To simplify the task I reduced the classes to 4 emotions that might be easier to identify: calm, happy, fearful, and disgust. The training size shrinks to 768. It would be a challenge to implement the learning algorithms to achieve good learning results.

Audio Data: https://drive.google.com/file/d/1wWsrN2Ep7x6lWqOXfr4rpKGYrJhWc8z7/view

Forest Cover Type

This study area includes four wilderness areas located in the Roosevelt National Forest of northern Colorado. The actual forest cover type for a given observation (30 x 30 meter cell) was determined from US Forest Service (USFS) Region 2 Resource Information System (RIS) data. Independent variables were derived from data originally obtained from US Geological Survey (USGS) and USFS data. Data is in raw form and contains binary columns of data for qualitative independent variables (wilderness areas and soil types)[3].

The original datasets have more than 580,000 instances, and 54 attributes. To build the model efficiently, in the following learning algorithm studies, I randomly selected around 3000 instances to train the models.

Data Set: https://archive.ics.uci.edu/ml/datasets/Covertype

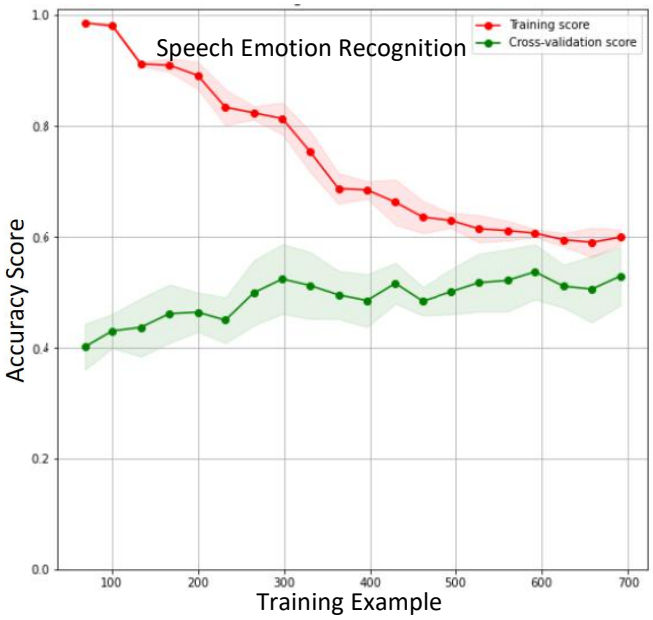
Learning Algorithms

Decision Trees

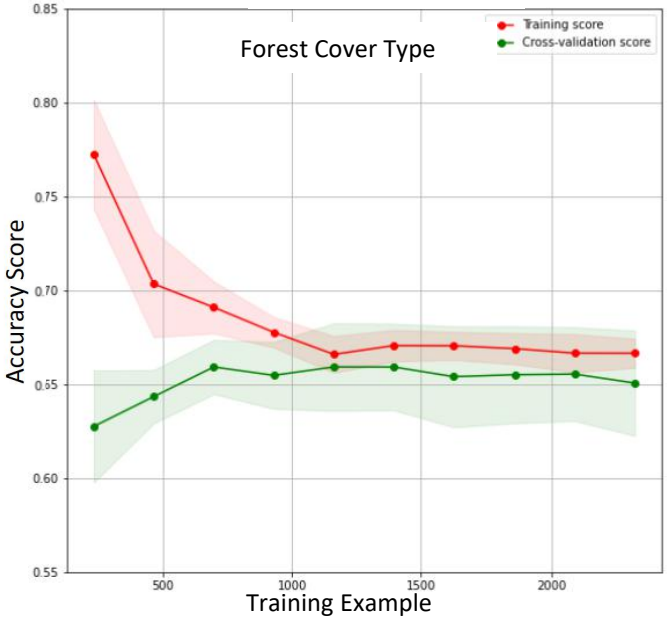
The Decision Tree algorithm I implement is CART (Classification and Regression Trees), which is very similar to C4.5 but it differs in that it supports regression and does not compute rule sets. CART constructs binary trees using the feature and threshold that yield the largest information gain at each node. To determine each tree node, I used Gini impurity because it's a faster way to compute the result and it performs well. And for pruning, I used Minimal Cost-Complexity Pruning to avoid overfitting.

My first study is decision tree's learning curve over training size. By increasing the training example size from 10% to 100%, and using 10 folds cross validation, I get the learning curve of training score and test score. As expected, in both classification problems the training data accuracy decreases as the training size goes bigger, while the test data accuracy increases.

Speech Emotion Recognition

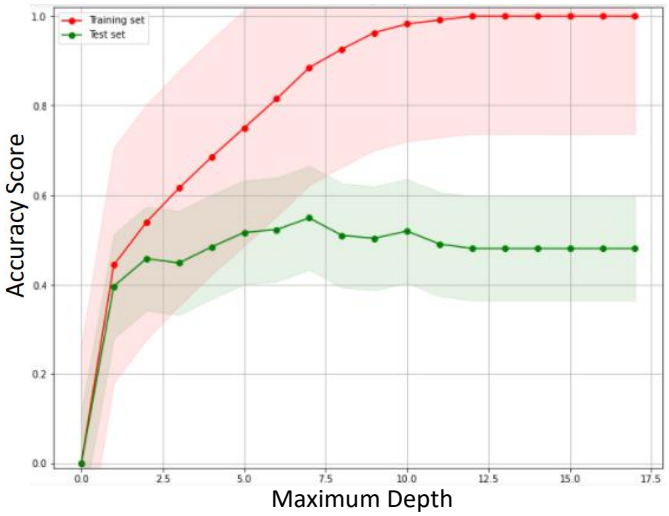


Forest Cover Type

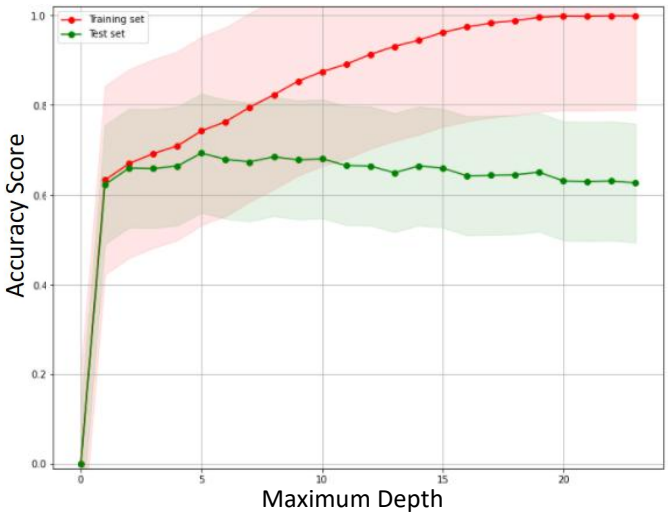


The second study is the learning curve over maximum depth of the tree. I set the maximum depth of the tree to increase from 2 to 50, and remove the pruning method. The goal is to see how overfitting influence the predict accuracy of the model. For both classification problems, the train set accuracy increases as the maximum depth goes up, and till the end the train set almost did perfectly. But the test set accuracy increases at the first and then decreases. Without pruning, the model overfit the training set and did poorly on the test set.

Speech Emotion Recognition



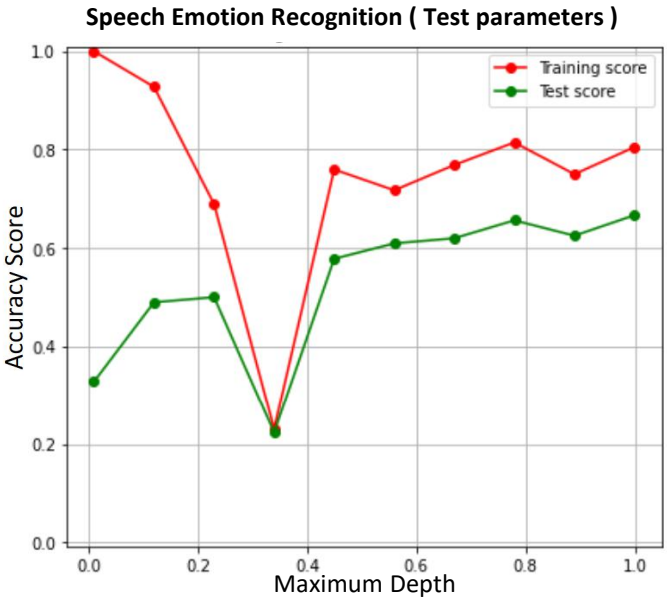
Forest Cover Type

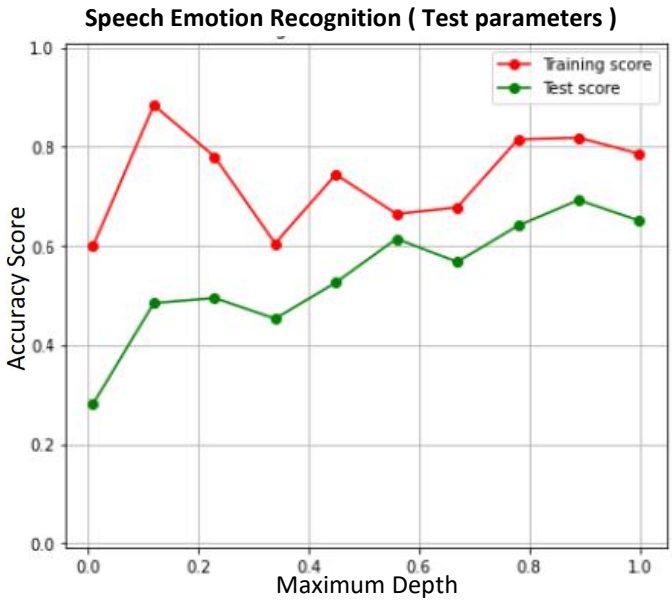


Neural Network

For Neural Network algorithm, I implemented Multi-layer Perceptron classifier algorithm from Scikit-Learn. The name explains itself, MLPClassifier relies on multi-layer perceptron to classify the data, and it trains the model using Backpropagation.

I analyzed the learning curve as the size of the neural network training size grows. Like what I did before, I increased the training example from 10% to 100%, but I got a output of Speech Emotion Recognition learning curve like the image below (left):



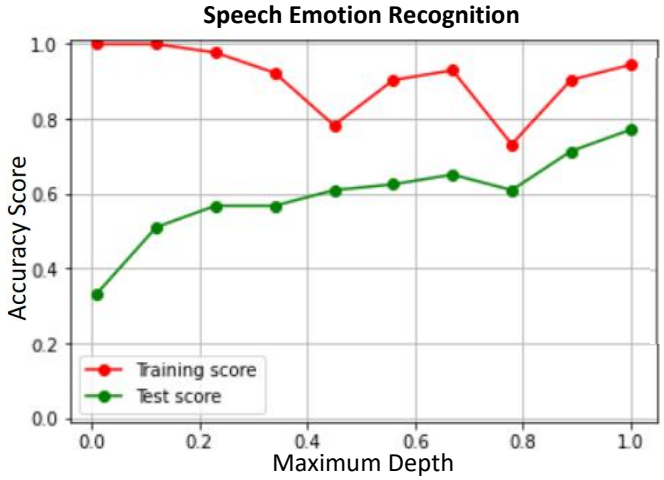


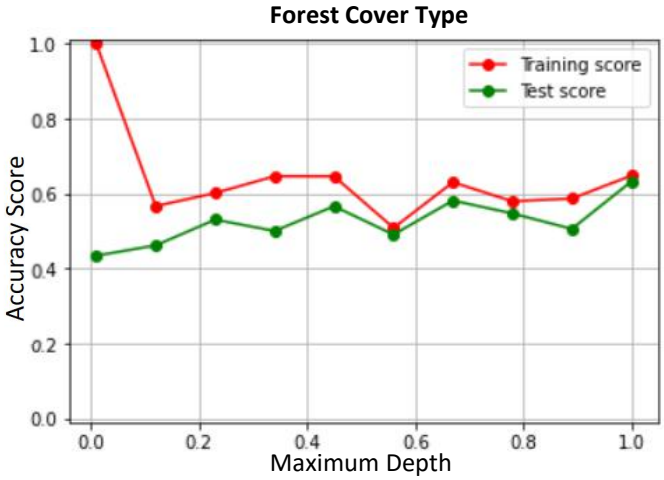
Default: hidden_layer_size = (100, 0)
Maximum iteration = 200

hidden_layer_size = (300, 0)
Maximum iteration = 500

I started to change the parameters from the default values, the parameters change and the result are shown above (right).

And I tried a lot of parameter combinations it still didn't make sense to me. I read the algorithm document again and realized that I set the hidden layer size to be 300 hidden units on 1 hidden layer, which is not what I want, I want it to be a sufficient multiple-layer network. After some experiment, I set the hidden layer size to be 4 hidden layers and each has (200, 100, 50, 25) hidden units, I got the result as below (left).





hidden_layer_size = (200, 100, 50, 25)

hidden_layer_size = (200, 100, 50, 25)

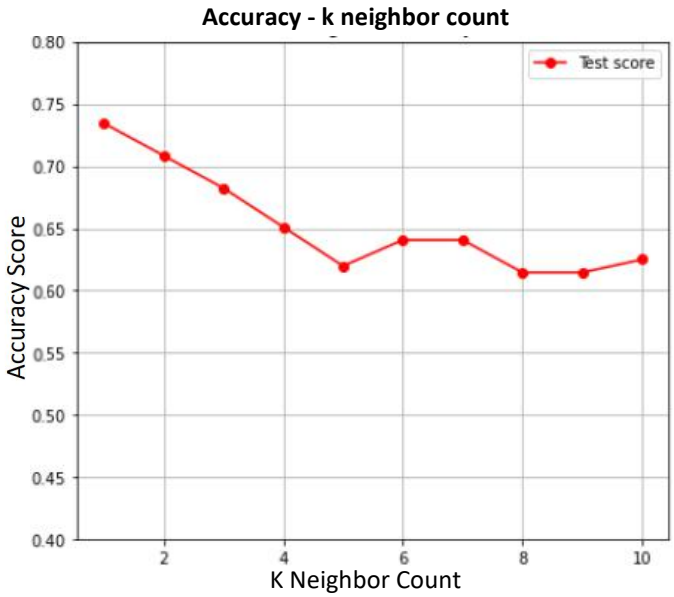
For Forest Cover Type, I set the hidden layer size to be 4 hidden layers and each has (200, 100, 50, 25) hidden units as well, and removed the maximum iteration limit. The result is shown as above(right).

K-Nearest Neighbors

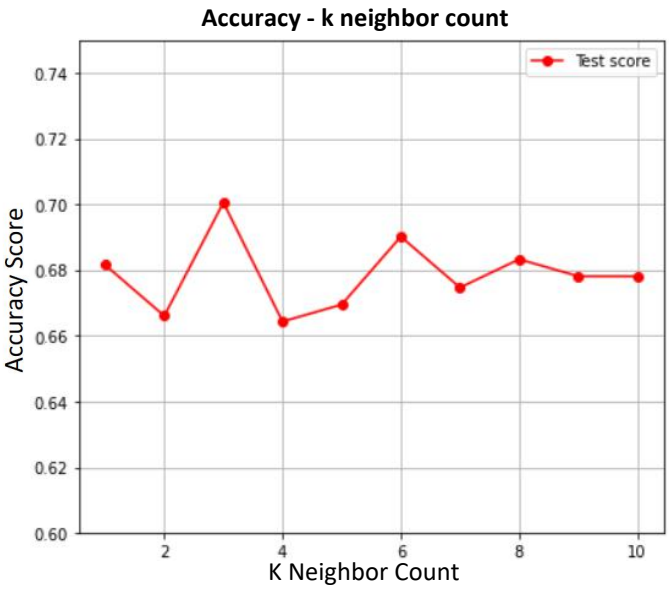
I implement the k-nearest neighbors vote algorithms for both classification problems. The analysis has two steps, first, use uniform weight for k-NN votes, analyze how k number influences the test accuracy, and how much time it cost for learning and classifying new data. Second, use weighted neighbor votes based on the distance, and compare the performance with uniform weight k-NN.

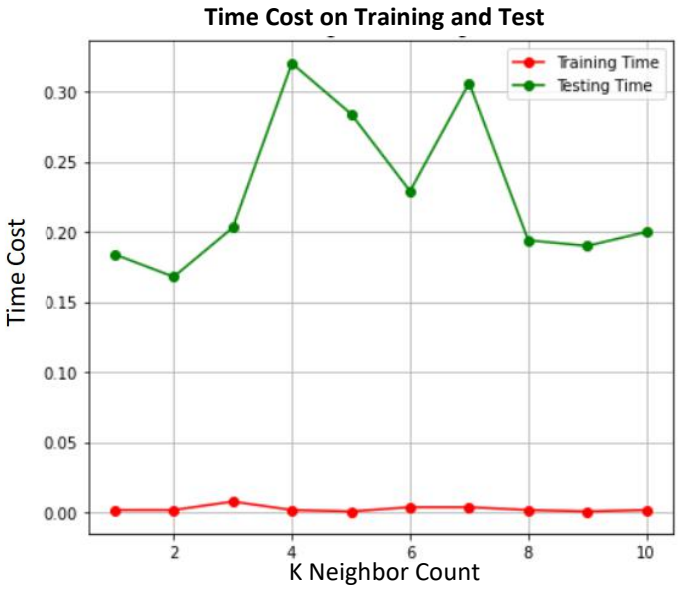
Uniform Weight:

Speech Emotion Recognition:



Forest Cover Type:





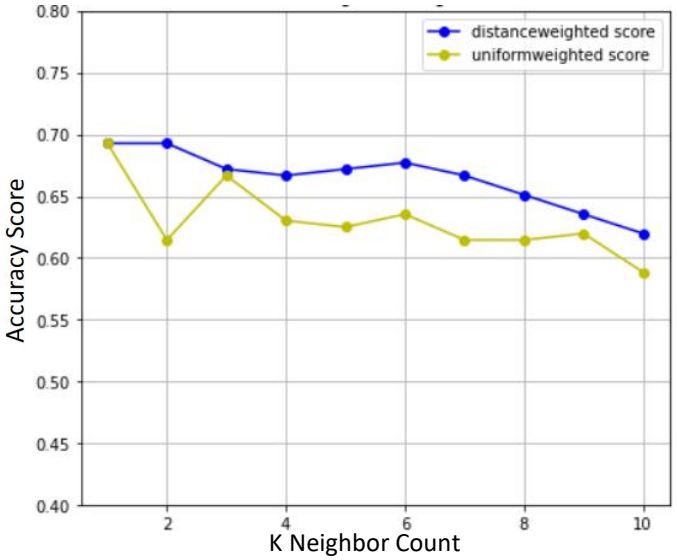
Analyzing the charts above, for Speech Emotion Recognition, the uniform weight accuracy decreases as k number increases. My explanation is when all the data points are weighted same, the more neighbors included in the vote, the more error happens. It may not be a good idea to vote equally (use uniform weight). But for Forest Cover Type, the overall accuracy isn't influenced by the k number change, though there are best perform k found(k = 3). My explanation is for Forest Cover type there are more training samples and it's easier to find neighbors close to each other. Even though more neighbors vote for the classification, they are close to each other so the result didn't change much.

For both classification problems, the learning time of the model is way lower than the query time of new data. That's an important feature of k-NN algorithm, the learning process is simply storing instances to database. And when new classification problems appear, it constructs different target functions for different instances, which will take longer time.

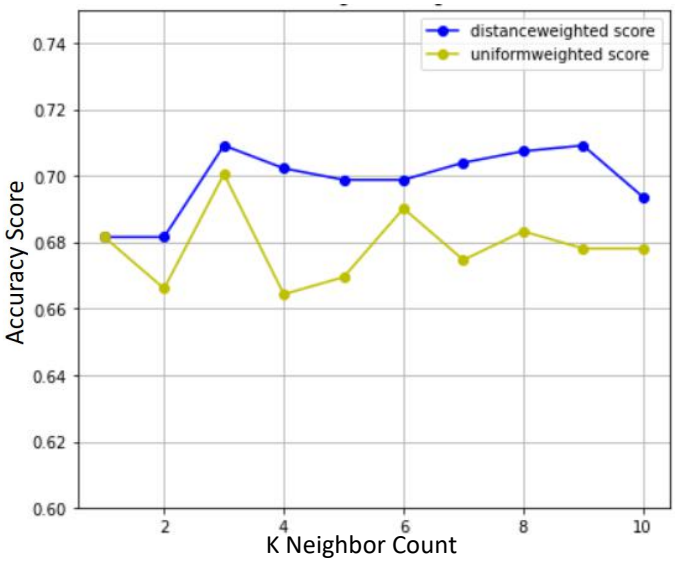
Distance Weighted vs Uniform Weight:

Below are the charts of test accuracy for both distance-weighted kNN and uniform weight kNN.

Speech Emotion Recognition



Forest Cover Type



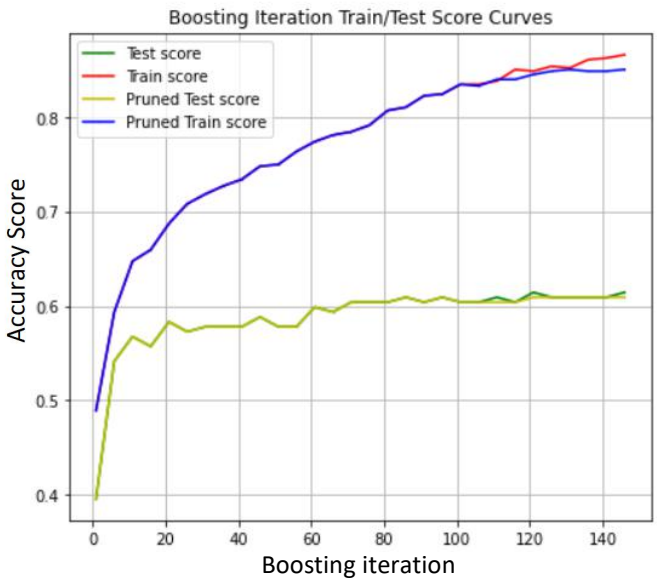
Analyze the charts above, for both classification problems, the distance weighted k-NN performs better than the uniform weight one. It proves that for some classification problems distance weighted neighbors vote can improve the accuracy of the k-NN algorithm.

Boosting

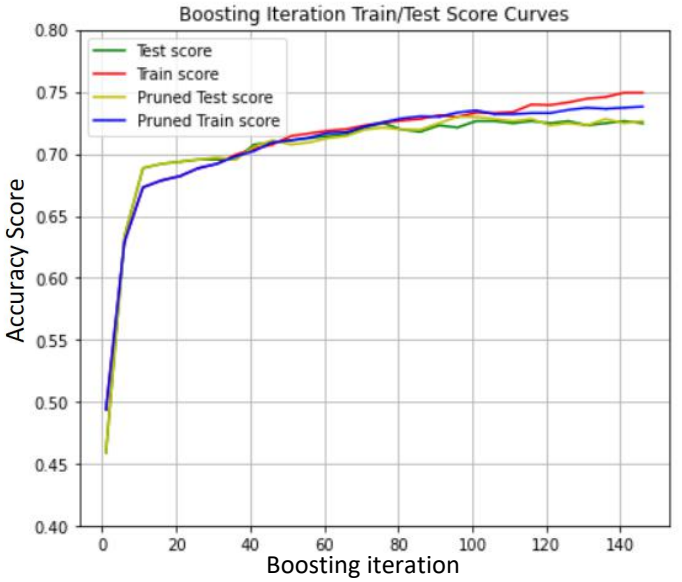
For Boosting learning method, I implemented the gradient boosting algorithm, which combines weak "learners" into a single strong learner in an iterative fashion, and can be used in both classification problems and regression problems. Gradient Boosting produces a random decision forest. I have applied Minimal Cost-Complexity Pruning method for pruning.

It is said boosting usually doesn't overfit, to experiment with this I compared two types of learning curves of the pruned and unpruned boosting methods. The first learning curve is the test accuracy changes according to the boosting iteration (how many weak learners it assembled) increases. There are 4 sets of data: unpruned test score, unpruned training score, pruned test score, pruned test score. The chart of Speech Emotion Recognition(left) and Forest Cover Type(right) are shown below.

Speech Emotion Recognition



Forest Cover Type

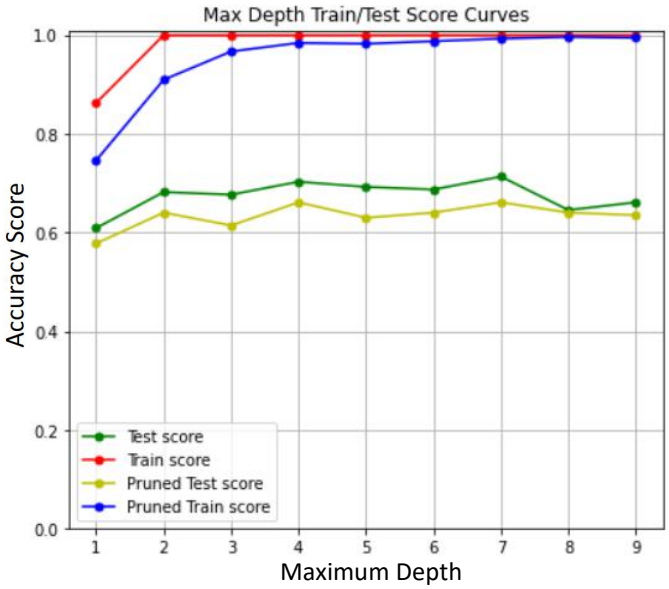


In Forest Cover Type chart, the pruned training curve and unpruned training curve overlaps when iteration is not that big, and when it becomes bigger, unpruned training curve starts to perform slightly better than the pruned training curve. It makes sense that unpruned boosting is trying to fit the training data better. However the pruned test data and unpruned test data are performing the same the whole time, unlike the decision trees, the unpruned test curve doesn't decrease no matter how many iterations are added.

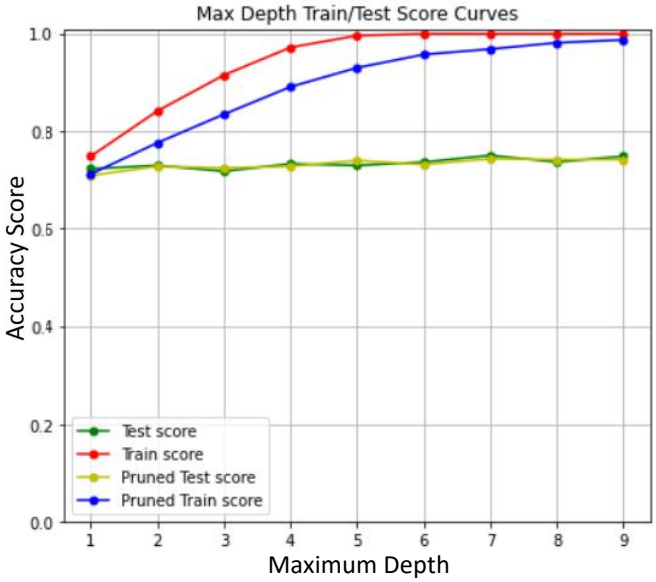
Same for Speech Emotion Recognition chart, the unpruned test data doesn't seem to overfit. But its learning result is not as good as the Forest Cover Type.

The second type of charts is to see how the four types of curves act when the maximum depth of each decision tree changes. I limited the max depth of each individual tree to (1, 2, 3....., 9). The results are shown below.

Speech Emotion Recognition



Forest Cover Type



Overall, both classification problems' pruned/unpruned training scores increase as I give them more depth in each "weak" leaner, and the unpruned training set performs slightly better than the pruned training set. But the pruned/unpruned test set accuracy scores haven't changed much. My explanation is that I only restrict the max depth of each weak learner, but they can have as many weak learners as they want, and weak learners only need to do a little better than the possibility, so given enough depth doesn't influence the performance. In Speech Emotion Recognition the unpruned test set even did better than pruned test set. The pruning method would be unnecessary in this case.

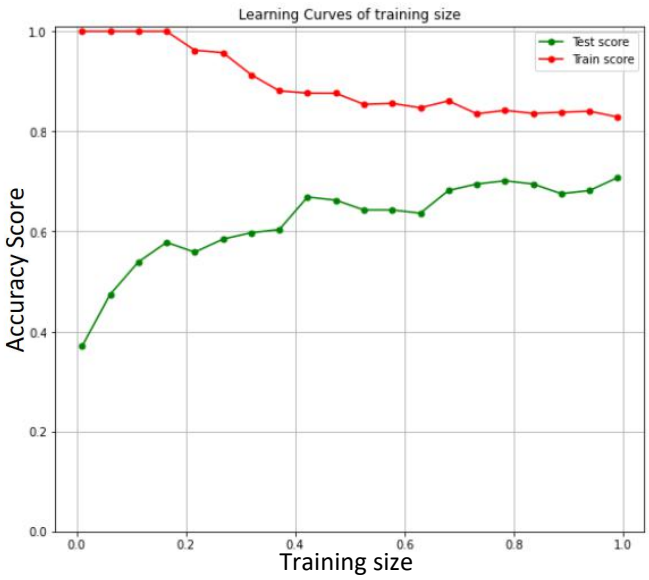
SVMs

The implementation of SVMs is based on libsvm. I applied 4 kernel functions to SVMs, Radial basis function kernel, Sigmoid kernel, Polynomial kernel, and Linear kernel.

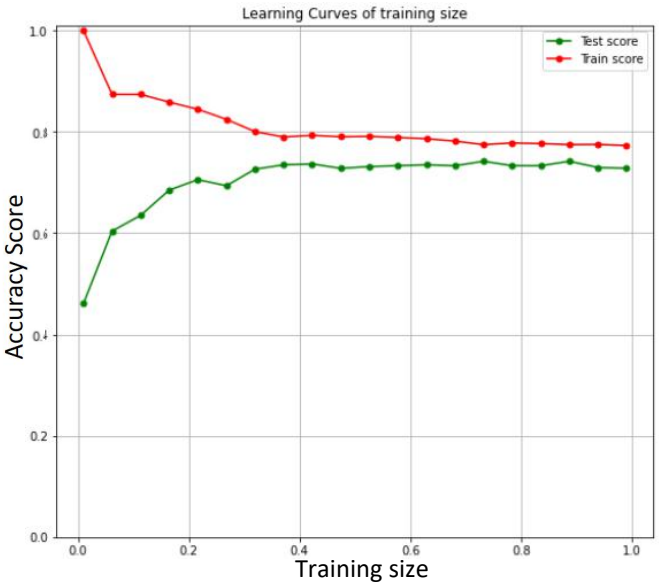
Radial basis function kernel:

RBF can be seen as a combination of distance-weighted regression and neural network. I applied RBF kernel SVM to both classification problems and generated the learning curve of the training size. Training size increases from 10% to 100%.

Speech Emotion Recognition



Forest Cover Type

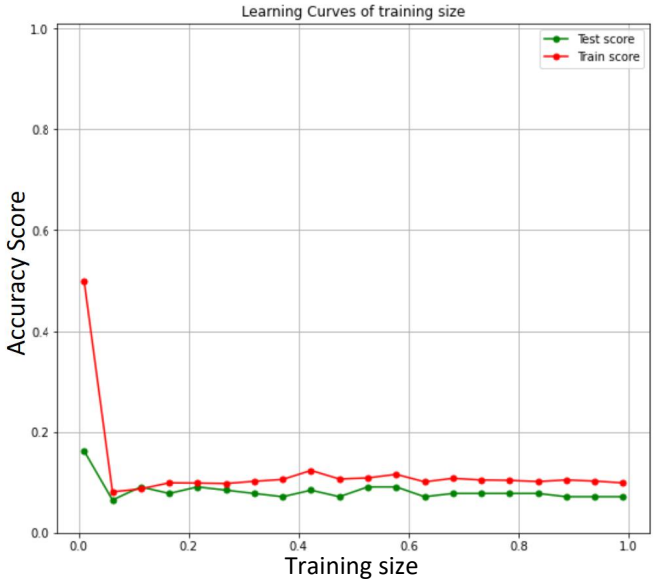


Both classification problems show the training scores decrease and test scores increase as the training sizes get bigger. The Speech Emotion Recognition has fewer total datasets, I believe if more examples can be given, the test score can keep increasing.

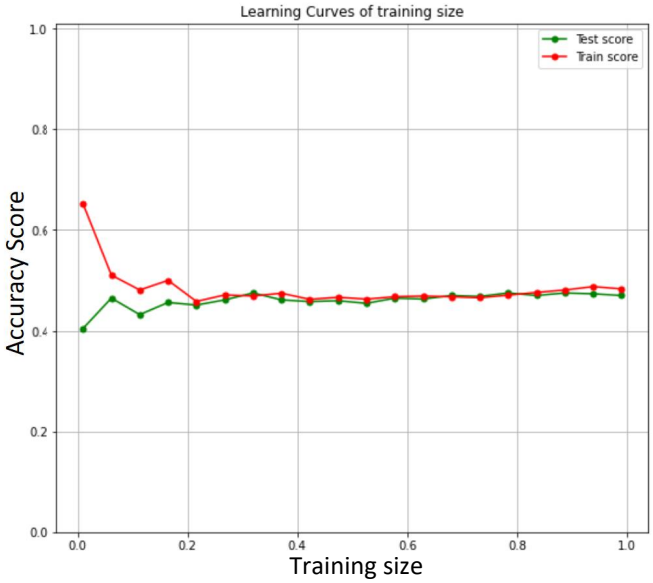
Sigmoid kernel:

Sigmoid kernel can be seen as a two-layer, perceptron model of the neural network, which is used as an activation function for artificial neurons.

Speech Emotion Recognition



Forest Cover Type

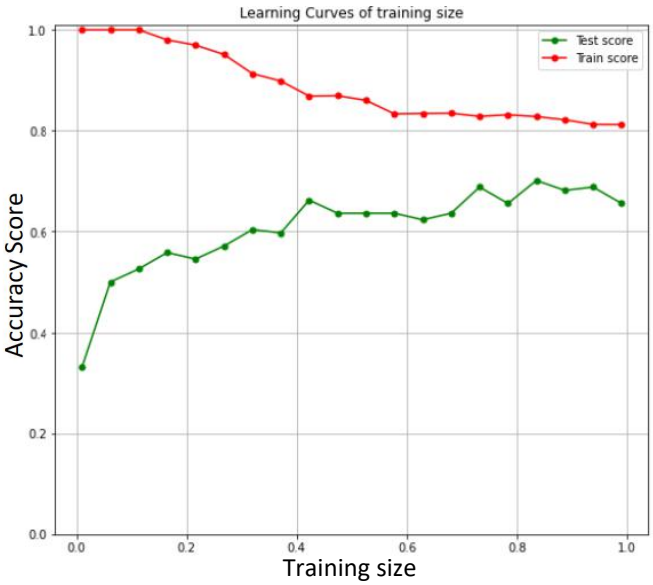


The results are shown as above. From the chart both classification problems did poorly, both didn't reach 50% predict accuracy. That's because the sigmoid kernel is more suitable for binary classification, but Speech Emotion Recognition and Forest Cover Type have 4 and 8 classes, sigmoid kernel is not the best kernel function for them.

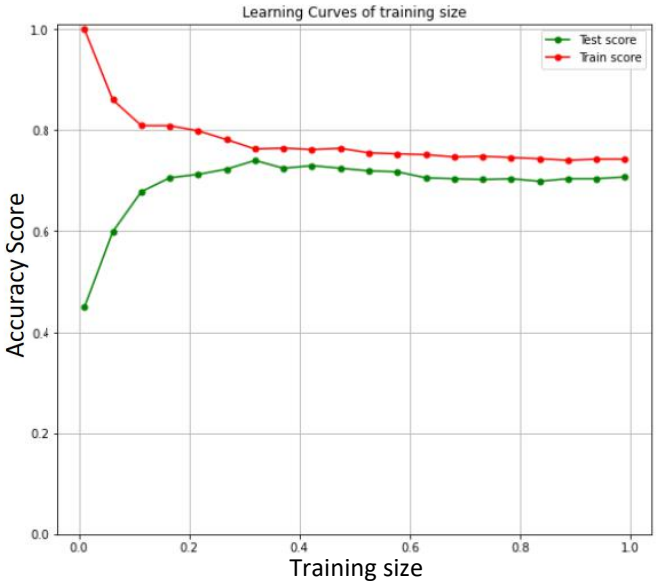
Polynomial kernel:

It represents the similarity of vectors in the training set of data in a feature space over polynomials of the original variables used in the kernel. I used the degree 3 polynomial kernel in the SVMs.

Speech Emotion Recognition



Forest Cover Type

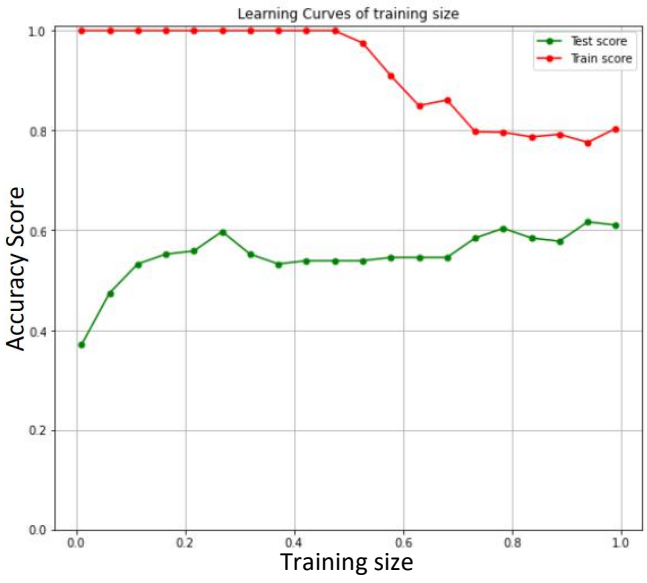


Linear Kernel:

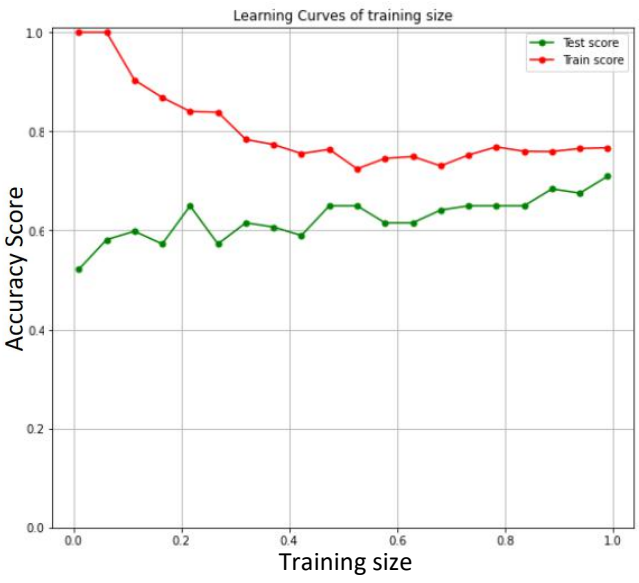
I also tried the linear kernel. This kernel function is the most time-consuming one within the 4 kernels I picked. It is also very expensive in computation. Based on libsvm, the implementation of linear kernel scales between O (nfeatures × nsamples^2) and O (nfeatures × nsamples^3) depending on how efficiently the libsvm cache is used in practice [4]. And my classification problems have a lot of features, which leads to a long processing time.

Because it took forever to run the analysis for Forest Cover Type data, I had to scale down the datasets, I reduced the training sample and test sample by 80% (It still took a whole day to run). The learning curves of Speech Emotion Recognition and Forest Cover Type are shown below.

Speech Emotion Recognition



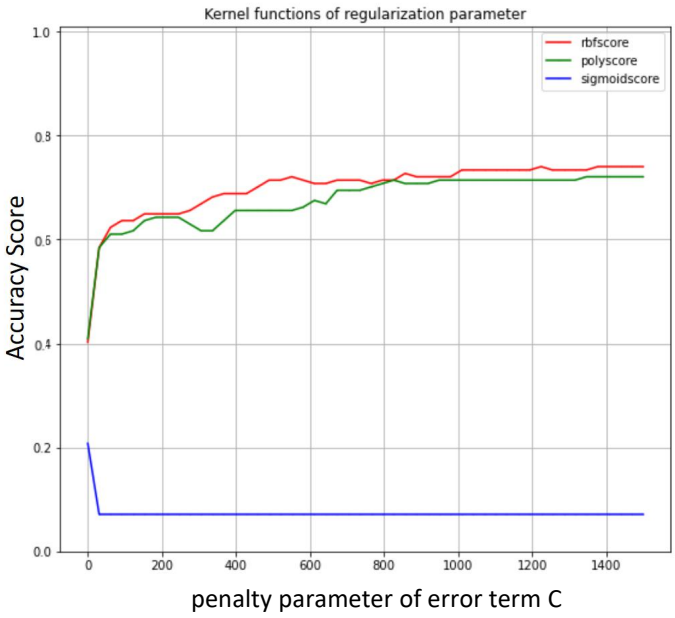
Forest Cover Type



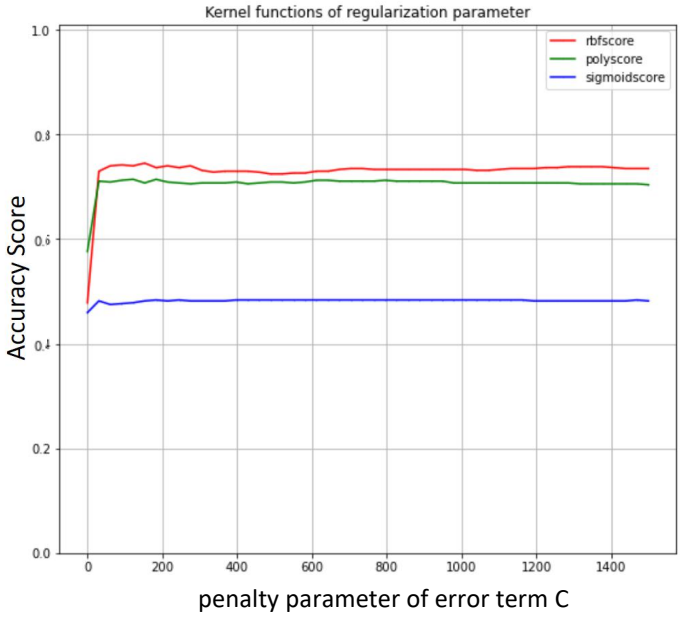
After implementing each single kernel function, I want to compare the performance of 3 kernel functions: Radial basis function, polynomial, and sigmoid. (Because linear kernel took too much time to run I didn't include it.)

The chart below shows 3 kernel function curves, and how the model accuracy changes as the penalty parameter of the error term C increases from 0.1 to 1500.

Speech Emotion Recognition



Forest Cover Type

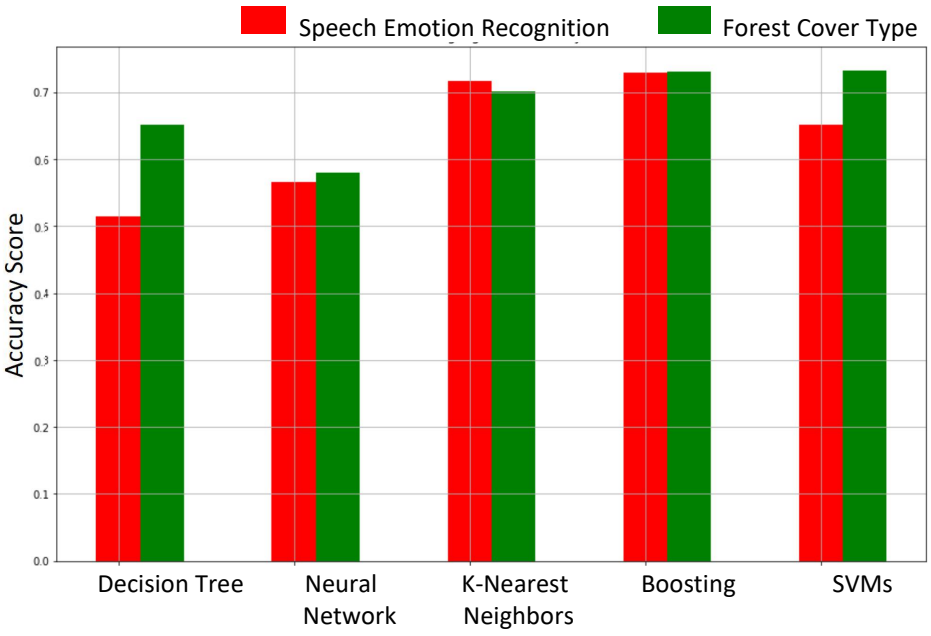


For Speech Emotion Recognition, Radial Basis Function kernel is the best-performed kernel function in these 3 kernels, the accuracy is the highest and it reaches 75%. The sigmoid kernel doesn't suit the classification problem, as the penalty parameter of the error term C increase, the performance get worse, lower than the probability.

For Forest Cover Type, Radial Basis Function kernel also performs best, and sigmoid the worst. But the sigmoid kernel did better than probability. I assume the Forest Cover Type problem is more binary separable than Speech Emotion Recognition.

Cross Validation Set Comparison and Conclusion

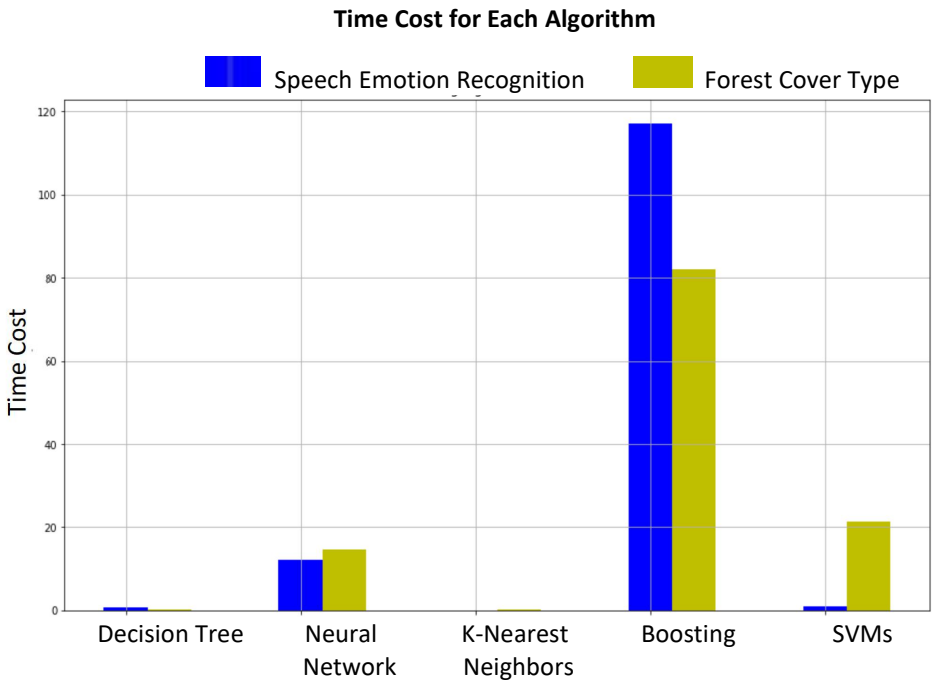
In the end, I compared the five learning algorithms and used 5 fold cross-validation to fit the model and compute the accuracy. Also, I compared each algorithm running time to observe the computation cost. The result for both classification problems is shown below.



As shown in the chart, for Speech Emotion Recognition problem, the best-performed learning algorithm is Boosting, and k-NN is very close to it. The worst-performed is decision tree.

For Forest Cover Type, Boosting and SVMs perform the best, and the worst one is Neural Network.

Overall k-NN, Boosting, and SVMs trained the models better than the rest. Especially for boosting, it has the least error in both classification problems.



The boosting costs the most time in both classification problems, and it also has the most accurate models. The neural network also costs some time for both datasets.

SVMs costs more time on Forest Cover Type than on Speech Emotion Recognition. This can be explained as Forest Cover Type's data take longer to map to high dimensional feature space, so it takes more time to find the classification hyperplane.

Reference

[1]
M. Ayadi, M. Kamel, F .Karray
Survey on speech emotion recognition: Features, classification schemes, and databases
Pattern Recognition, Volume 44, Issue 3, March 2011, Pages 572-587

[2]
J. Blackard, D. Dean
Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables
Computers and Electronics in Agriculture, Volume 24, Issue 3, December 1999, Pages 131-151

[3]
UC Irvine Machine Learning Repository
Covertype Data Set
Web page: https://archive.ics.uci.edu/ml/datasets/covertype

[4]
scikit-learn.org
Support Vector Machines
Web page: https://scikit-learn.org/stable/modules/svm.html#complexity