Assignment 1

1. Introduction

In the current work, 5 popular machine learning algorithms have been implemented on two different data sets. These algorithms are Decision Tree, K-Nearest Neighbors, Boosting applied to Decision Trees, Support Vector Machines and Neural Networks. These algorithms were implemented using standard sklearn packages. More information pertaining to the packages used for the project are included in the 'Readme.txt' file. A binary classification has been attempted on two different datasets using the aforementioned algorithms. Cross Validation on the hyper parameters have been performed to analyze the tradeoff between bias and variance. We intentionally selected one dataset of very large size (`142,000 instances) and another dataset with comparatively smaller size (~1300 instances) to contrast the performance of different algorithms on datasets of different sizes. Furthermore the Weather Dataset is big enough to experiment with varying sample training size and hence it makes it interesting to evaluate performance with training size and observe how having more data impact the performance of the algorithm. The Weather dataset is mix of categorical and continuous numerical values and it will be interesting to see how some algorithms perform on this type of data as data will be non-uniformly distributed.

2. Dataset Description

The two datasets explored in this problem are the Australian Weather Data from Kaggle and NBA data from DataWorld (refer to links in the Readme.txt file). For analysis of each algorithm, entire dataset is split into training and validation set (referred as test set in the rest of the paper). In this assignment accuracy-score is used to track the performance of the algorithms.

2.1 Australian Weather Dataset

The Australian weather dataset comprises of 142,000 instances and 18 attributes. These attributes include both continuous numerical data such as humidity, wind speed at different times of the day, pressure at different times of the day and categorical variables like Wind Direction and Wind gust direction at different times of the day. The target is a binary variable and is 'Yes' if it rains and 'No' if it does not. 'Yes' was mapped to 1 and 'No' was mapped to '0'. During data preprocessing, rows which had a single NA values were omitted. One Hot Encoding was used to handle string categorical variables.

2.2 NBA Dataset

This dataset has ~1300 instances and 19 attributes. The target outcome is 1 if career of the player lasted for more than 5 years and 0 if career lasted lesser than 5 years. All attributes are numerical. All rows containing one or more NA values were dropped.

The above mentioned datasets are interesting due to the difference in sizes of the datasets. Also the Weather Dataset is mix of categorical as well as continuous numerical variables. This might show

3 Algorithm Implementation and Analysis

In this section we will implement the five different machine algorithms: Decision Tree with pruning, K-Nearest Neighbors, Boosted version of Decision Tree, Neural Networks and Support Vector Machines. Various hyper parameters are varied for cross validation.

3.1 Decision Trees

A decision tree algorithm was implemented using sklearn's decision tree classifier. We will first show the performance of the tree with varying depth of the tree. More the depth of the tree better it can fit the training data well however it might have a poor performance on the test data as our tree is less generalized and we might over-fit the data. As the depth of the tree increases we can see that the training accuracy increases to almost 0.97 while the test accuracy starts to drop. Based on this analysis an optimum depth of 6 is suitable for the tree where we achieved a maximum test accuracy of ~0.85. We analyzed the same behavior for the NBA dataset and we observed similar trend and maximum accuracy was obtained around tree depth of 6-7. However the accuracy was quite low (~0.67) as compared to that obtained for the weather dataset. We will further look into this deeper in the subsequent sections.

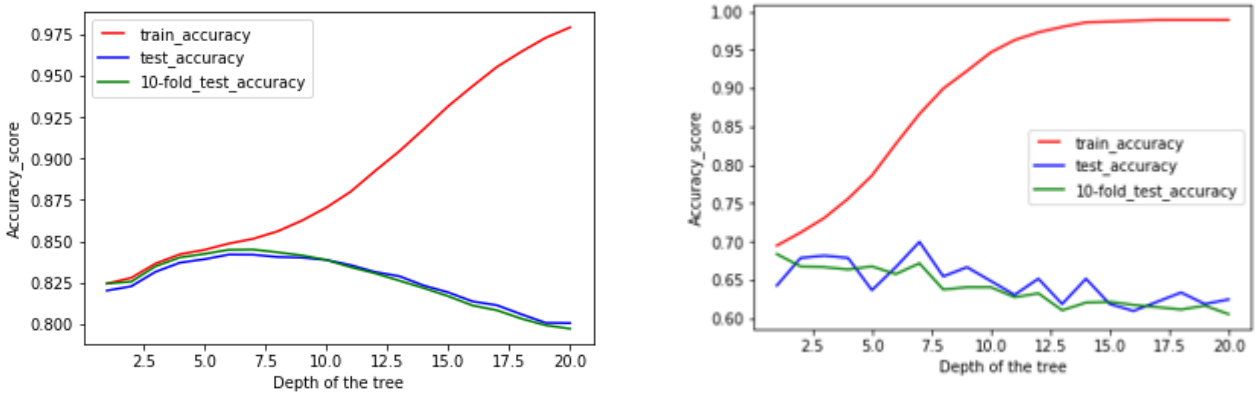

Figure 1. Variation of train, test accuracy and 10-fold test accuracy with tree depth for the Australian Weather Dataset (left) and for the NBA dataset (right).

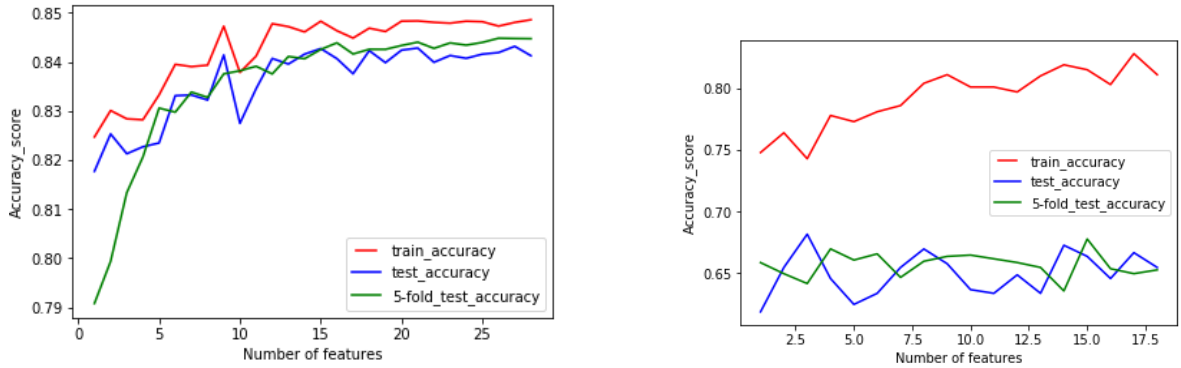

Figure 2. Variation of train, test accuracy and 5-fold test accuracy with number of features for the Australian Weather Dataset (left) and for the NBA dataset (right), tree depth=6

From the trends in Figure 2 we see that as we increased the number of features to be considered during the split, the accuracy first increased and then stayed constant after the Australian Weather Dataset while for the NBA dataset, accuracy remained more or less constant around ~0.65. When the number of features are specified to the sklearn decision classifier, it randomly selects the specified number of features to determine the best split. The Australian Weather Dataset might have some important features which behave similar in entropy reduction and hence after certain we start considering certain number of features, we start including these features which leads to similar entropy reduction and hence accuracy stays constant. The NBA dataset on the other hand might have majority or all of the features which produces similar entropy reduction and hence the trend stays more or less constant with increasing

number of features. Overall the features of NBA dataset are weakly related to the outputs and overall performance is poor

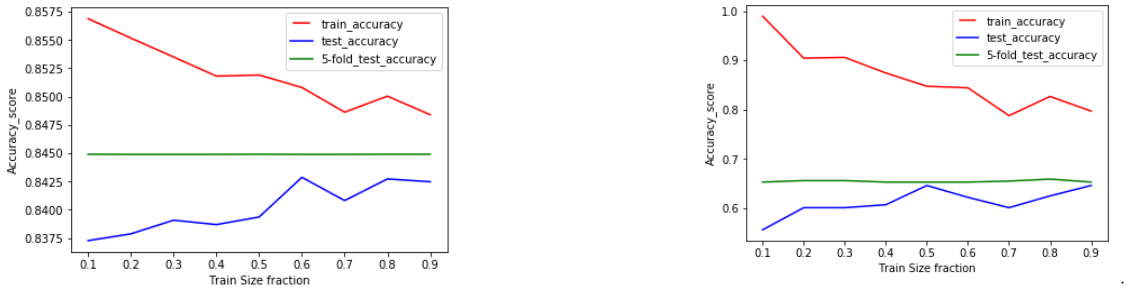

Figure 3. Variation of train, test accuracy and 5-fold test accuracy with training size for the Australian Weather Dataset (left) and for the NBA dataset (right), tree depth=6, number of features were not specified and was default selection.

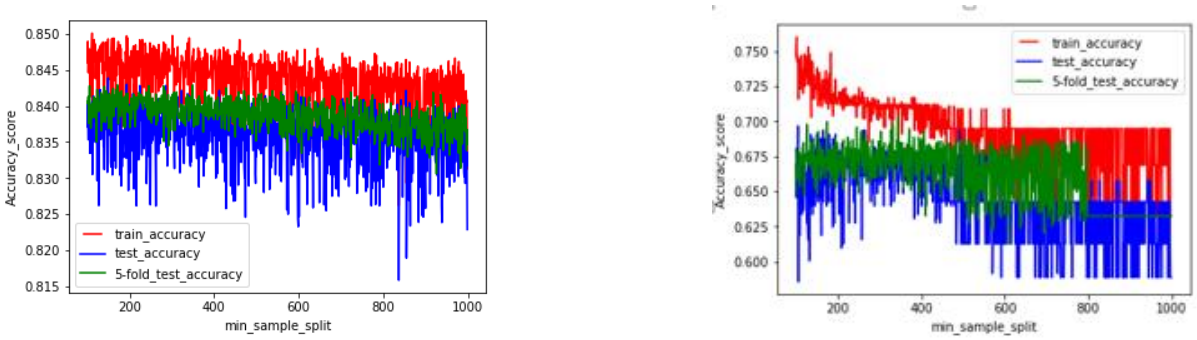

Figure 4. Variation of train, test accuracy and 5-fold test accuracy with training size for the Australian Weather Dataset (left) and for the NBA dataset (right), tree depth=7, number of features were not specified and was default selection.

Besides tree depth, other parameters such as minimum sample split and train size were varied keeping tree depth at 6. With varying training size, the performance stayed the same (Figure 3). Minimum sample split did not show a trend when varied for the NBA dataset, the entropy stayed around ~0.68.

3.2 K Nearest Neighbors

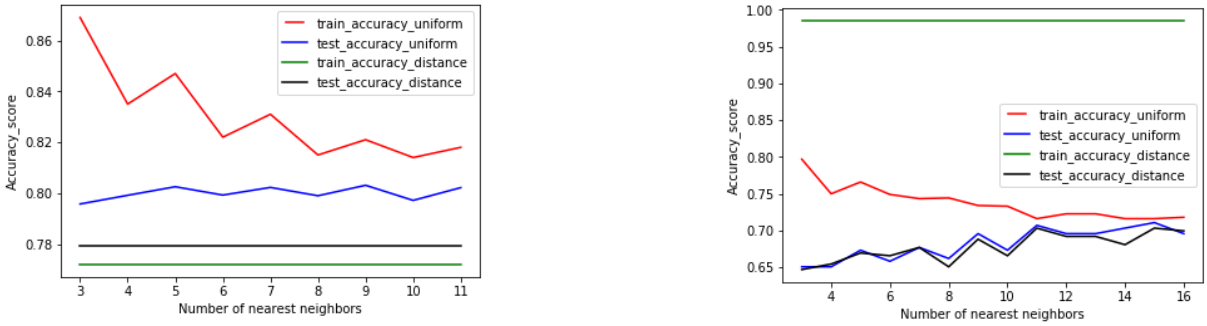

Figure 5. Trend of test and train accuracy for uniform and distance weights for p=1 for 1000 samples from Australian Weather Dataset and NBA dataset (right).

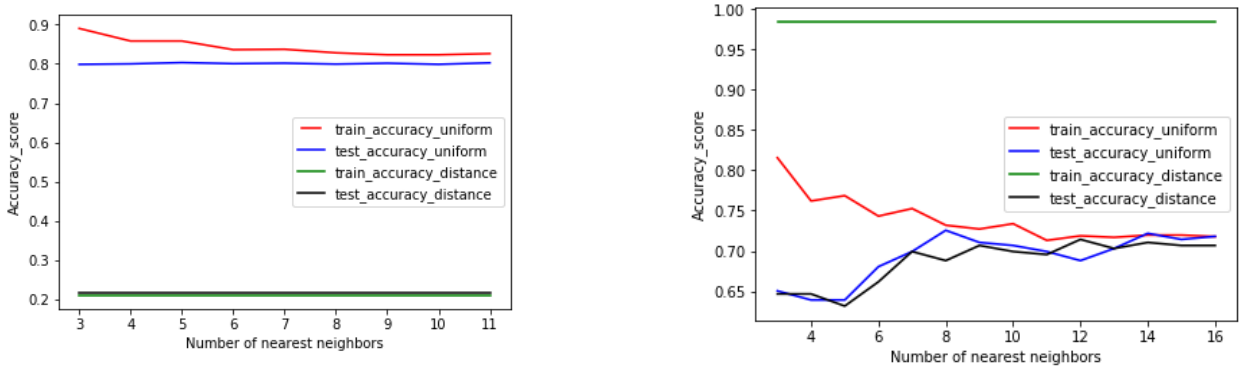

Figure 6. Trend of test and train accuracy for uniform and distance weights for p=2 for 1000 samples from Australian Weather Dataset (left) and NBA dataset.

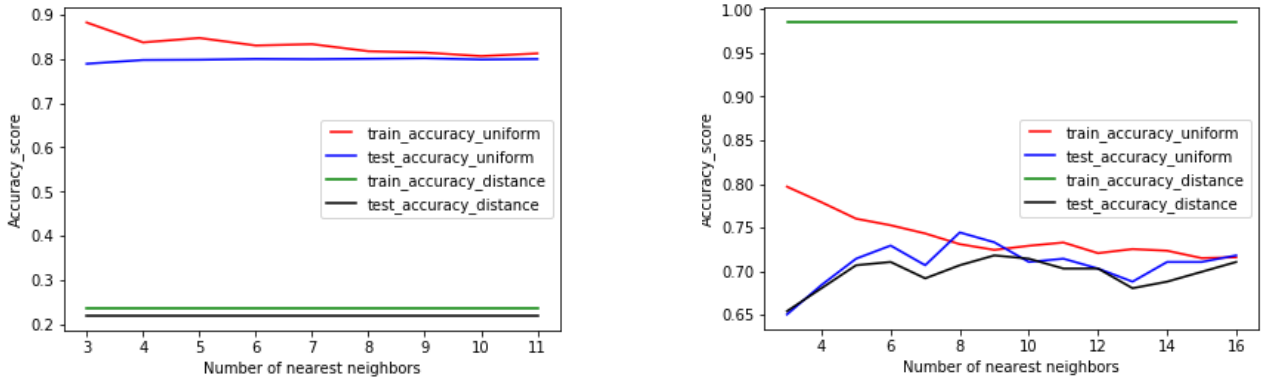

Figure 7. Trend of test and train accuracy for uniform and distance weights for p=3 for Australian Weather Dataset (1000 samples) and NBA dataset.

K-NN algorithm was implemented using sklearn KNeighbors classifier. We analyzed performance for type of weights: 'uniform' and 'distance' (inversely proportional to distance) and also type of distance (Euclidian, Manhattan, etc). We sliced the training data for Australian Dataset to 1000 samples to compare with the NBA dataset. For the Australian Weather Dataset, the accuracy stayed the same with varying number of neighbors irrespective of choice of type of distance (p=1, 2 or 3). For the NBA dataset the accuracy increased to ~0.75 for 8 nearest neighbors. Further we observed that for the weather dataset the algorithm performed well when the uniform weights were chosen for p=1, p=2 and p=3. Best accuracy obtained for the dataset of size 1000 for the K-NN algorithm was approximately 0.80. We further noticed that training accuracy declined as we increased number of neighbors and this is expected since increasing K will generalize and will underfit the data.

Next we explored the entire weather dataset and fine-tuned K-NN parameters to obtain best performance. Results are shown in the Table 1. Initially we attempted to find the K parameter which will give the optimum performance. We did that by taking a slice of 40,000 samples. Entire dataset was not considered due to the extensive time taken by the KNN algorithm to do computations over a range of values of K considered here for analysis.

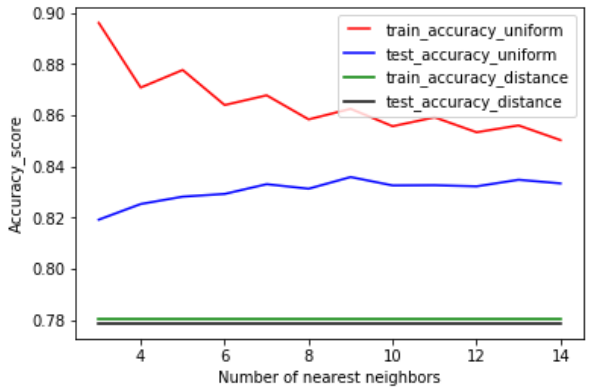

Figure 8. Accuracy score with varying values of K for a training size of 40,000

From figure 8, we observed that performance levelled off after K=6. We further investigated impact of training size on the performance. Results are shown in Figure 9.

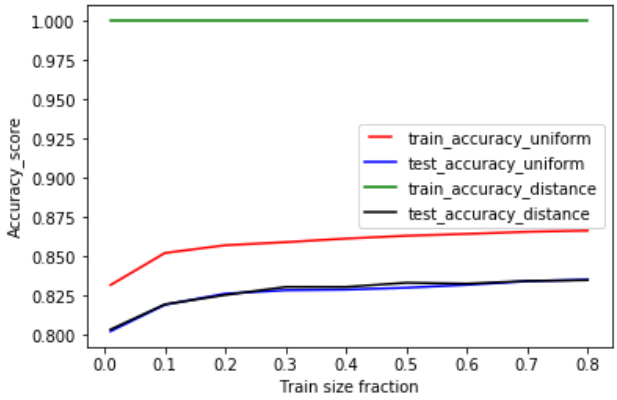

Figure 9. Trend of Accuracy score with increasing training size for the Weather Dataset for k=6 and p=1.

Impact of training size and performance of algorithm was analyzed for K=6 nearest neighbors, p=1. We observed that as training size was increased, the accuracy score for test set increased. A maximum accuracy of 0.83 was observed when training size increased to 80% of original set. Furthermore the performances were similar when the nearest neighbors were equally weighted and when the nearest neighbors were weighted by distance. This might imply that all the data is clustered at one place and is not uniformly distributed.

The optimized parameters are summarized in Table 1.

Table 1. Summary Table KNN

Parameter	Australian Weather Dataset	NBA Dataset
p (type of distance: Euclidian, Manhattan, etc)	1	3
K	6	9
weight	Uniform, distance	Uniform
Best Test Accuracy	0.83	0.75

3.3 Neural Network

The important hyper parameters for a Neural Network are: Number of hidden layers, Number of nodes at each hidden layer and number of iterations. These hyper parameters have been studied in this analysis. Furthermore for the weather dataset we analyzed impact of increasing size on the performance.

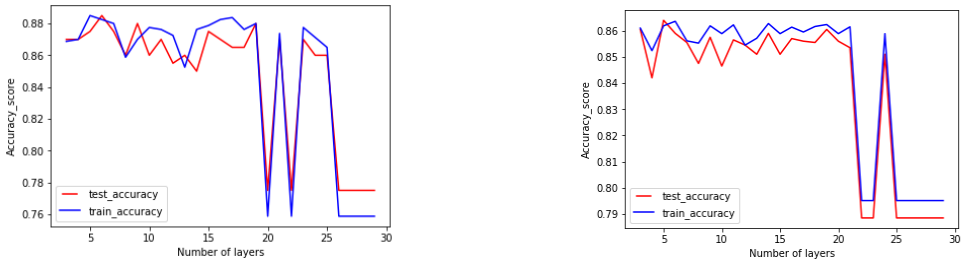

Figure 10. Trend of test and train accuracy with increasing neural network complexity (increase in number of layers) for Weather Dataset with sample size of 1000(left) and sample size of 10000 (right) each with number of nodes=10

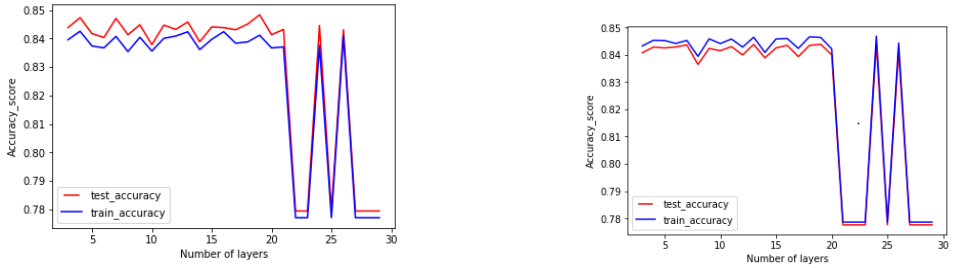

Figure 11. Trend of test and train accuracy with increasing neural network complexity (increase in number of layers) for Weather Dataset with sample size of 40000(left) and entire dataset (right) each with number of nodes=10

We conducted 4 experiments with 10 nodes and varied training data size for the Weather dataset (Figure 10 and 11). The accuracy of the test set stayed more or less constant at ~0.84. In the above curves we saw a sudden dip and spike in the accuracy scores for more than 20 layers. This might behavior might be due to the fact that we used the RELU activation function and after 20 layers, most of the weights might be turning into negative values making particular neuron dead. Similar behavior was observed in the NBA dataset as well.

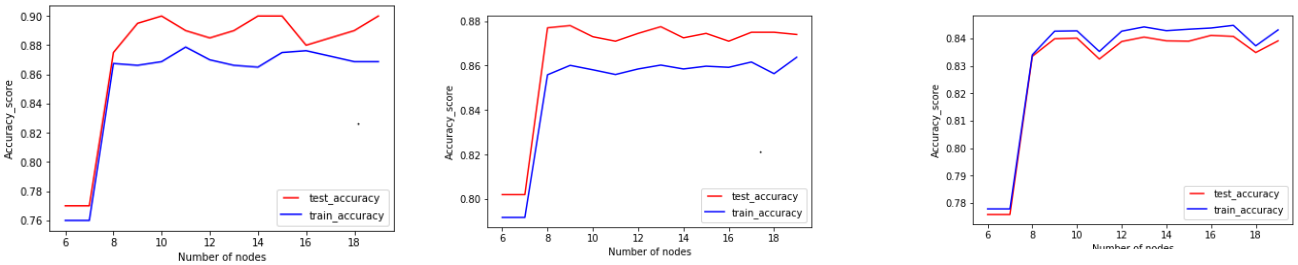

Figure 12. Trend of test and train accuracy with increasing layer complexity (increase in number of nodes) for Weather Dataset with sample size of 1000(left) and 10000 (middle) and 40000 (right)each with number of layers=5

Based on the previous experiments we maintained number of layers as 5 so as to not increase model complexity unnecessarily and we varied number of nodes. Varying number of nodes in the hidden layer will imply fitting a more complex function each time input is fed to a layer. We notice that post 8 nodes, accuracy stayed constant and there was no significant improvement in performance. Thus 8 nodes was considered sufficient for the weather dataset. We further noticed that as we increased the data size the test performance decreased to 0.84 for sample size of 40,000 data points from ~0.90 for a sample size of 1000. This might be attributed to the inherent noise in the dataset.

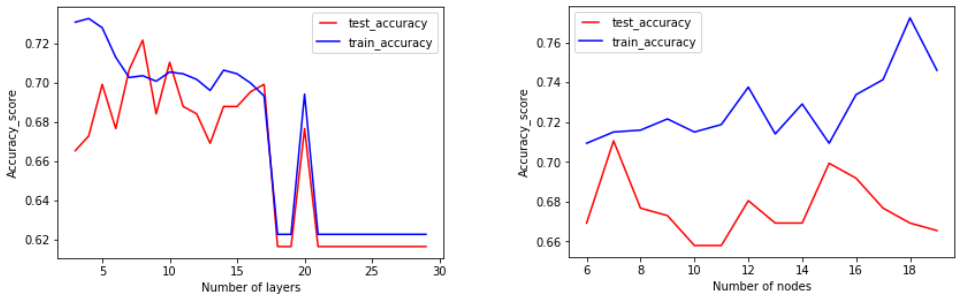

Figure 13. Trend of test and train accuracy for NBA dataset with increasing model complexity (left) for number of nodes =10 and increasing number of nodes (right) with number of layers =7.

For the NBA dataset we analyzed the performance by increasing the number of nodes while keeping number of layers constant (Figure 13, right) and by increasing the number of nodes and keeping number of layers constant (Figure 13, left). The test accuracy fluctuated between 0.67 and 0.72 and was maximum for number of layers=7. It is hard to make a confident judgment from this figure and it is more reasonable to state that there was no significant change in the test accuracy on increasing the number of layers. We then fixed number of layers at 7 and varied number of nodes for each layer. Maximum performance was observed for n=7 nodes. No particular trend is observed for variation with nodes and it is hard to comment if at n=7 nodes is maxima or is just variation in the dataset.

We will use n=7 nodes and 7 layers as optimized parameters for the NBA dataset. With the optimized parameters for NBA dataset and Weather Dataset, we studied trend of different number of iterations for the weather and NBA dataset. Results are displayed in Figure 14. The results indicate that gradient descent step is able to reach its minima in about 300 iterations and there is no significant change in performance for the weather dataset. On the other hand for the NBA dataset which is just ~1300 samples, the optimum performance was observed for 300 iterations. Overall performance did not vary much with varying number of iterations. Another interesting phenomena observed here was that the test performance was higher than the training performance. This phenomena has been previously reported by the scientists for the IRIS dataset while using SVM algorithms [2]. Such a phenomena was observed when dataset was small and the two classes overlapped or were too close to each other. In an effort to generalize and prevent under fitting, data near the boundary is classified better in the test dataset as compared to the training dataset.

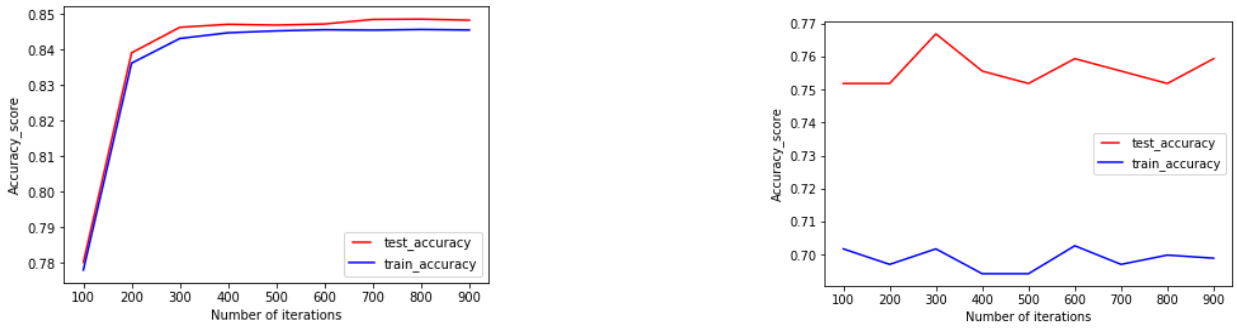

Figure 14. Test and train accuracy for n=8 nodes and 5 layers for the weather datset and n=7 nodes and n=7 layers for the NBA dataset

Table 2. Summary Neural Networks

Parameters	Weather Dataset	NBA Dataset
Number of Layers	5	7
Number of nodes	8	7
Number of iterations	300	300
Test Accuracy	~0.84	~0.76

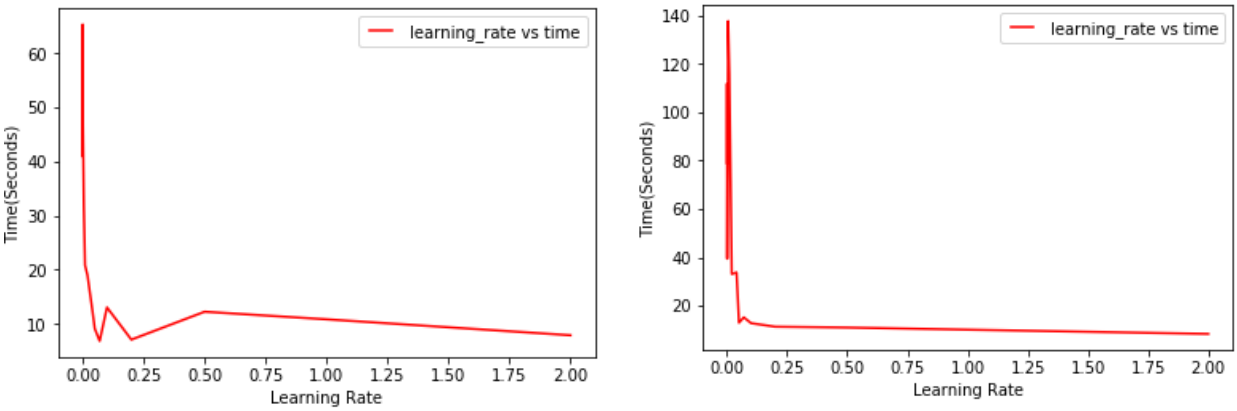

Figure 15. Learning Rate and time for convergence for the Weather Dataset (left) and NBA Dataset (right)

Figure 15 shows the trend of learning rate with time. As we increased the learning rate there was not much improvement in time of convergence of the model. If learning rate is too high, we might have difficulty achieving the convergence in the solution. Although the gradient descent step will move fast however it might miss the minima. Thus a model with high learning rate might not converge as fast as model with moderate learning rate. Therefore we increase the learning rate it might just perform same as small learning rate. For the weather dataset the curve flattened after learning rate of ~0.02 and for the NBA dataset this was observed at learning rate of ~0.05.

3.4Boosting

In this algorithm we varied number of trees, samples to be considered during the split at a node and also the tree depth.

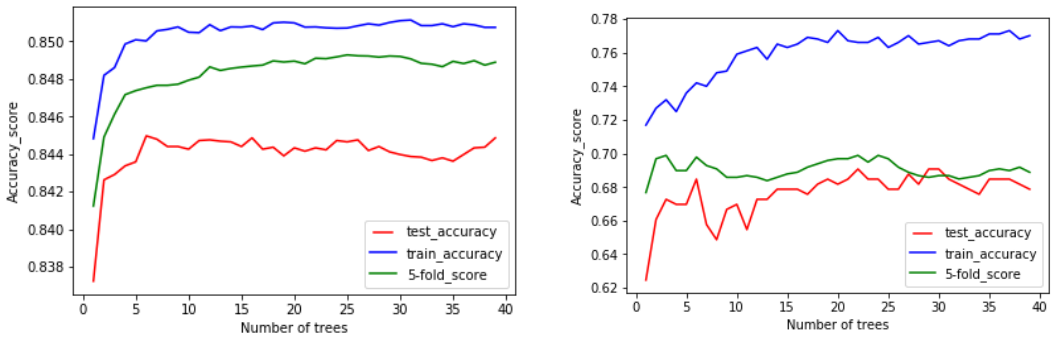

Figure 16. Accuracy trend with varying number of trees, 15 features and 100 samples as Minimum split for Weather Dataset (left) and NBA dataset (right)

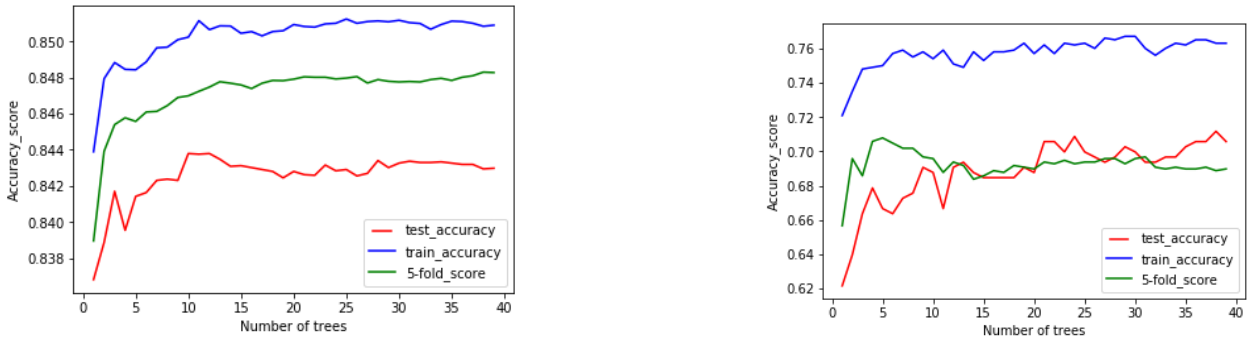

Figure 17. Accuracy trend with varying number of trees, 10 features and 100 samples as Minimum split for Weather Dataset (left) and NBA dataset (right)

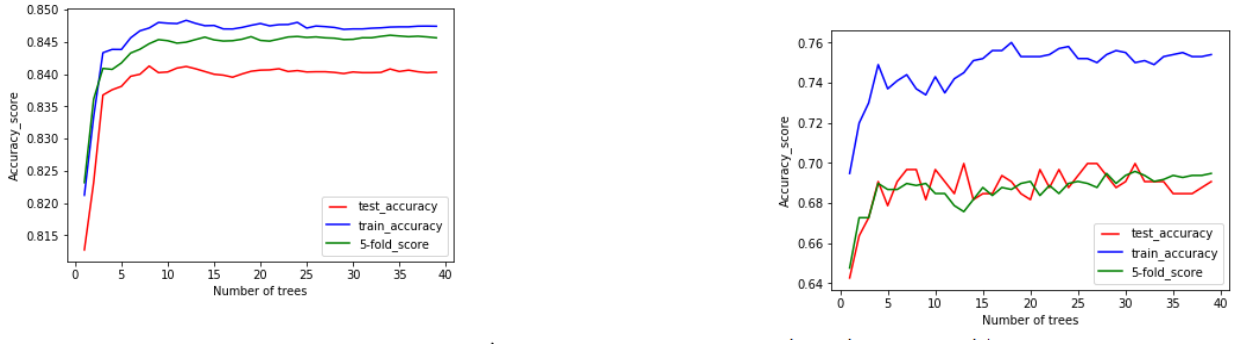

Figure 18. Accuracy trend with varying number of trees, 5 features and 100 samples as Minimum split for Weather Dataset (left) and NBA dataset (right)

From figure 16 we observe that for a random forest model with 7 decision trees, the accuracy score for the weather dataset reached a maximum of 0.845 and then levelled off. For the NBA dataset similar trend was observed at accuracy score reached a maximum of 0.68 when 7 decision trees were included. Comparing the maximum accuracy obtained for both NBA dataset and weather dataset in Figure 16, 17 and 18 we observed that the maximum accuracy scores obtained from the weather dataset decreased slightly (0.845 to 0.840) as number of features decreased. This might be due to the fact that as we reduce number of features, the probability that the attribute which results in the lowest entropy is selected for a split decreases.

Table: Accuracy scores with varying number of samples at split and tree depth for n=10 decision trees and 15 features.

Table 3. Test Accuracy with variation of tree depth and Min_split

Dataset	Tree Depth	Min_Split	Test_Accuracy
Weather Dataset	6	100	0.843
Weather Data	10	100	0.848
Weather Data	15	100	0.849
Weather Data	6	1000	0.841
Weather Data	10	1000	0.842
Weather Data	15	1000	0.843
NBA Dataset	6	100	0.66
NBA Dataset	10	100	0.675
NBA Dataset	15	100	0.675

In the table above we observed that as we increase tree depth there was slight increase in performance but it was not significant. Earlier while using a single tree we have shown that as tree depth increases beyond 6, the performance starts to decline. The performance scores for 1000 samples at split were slightly lower as compared to the 100 samples. This might be due to the fact that if we consider 100 samples for a split, we might be producing a weak learner or in other words a classifier that is more generalized. An ensemble of many such classifiers will give us a model that will perform better during validation. The evaluation of NBA dataset was done for only 100 samples for Min_Split. It was observed that the accuracy score stayed similar with increasing tree depth.

3.4 Support Vector Machines

This algorithm is implemented using three different kernels: linear, rbf and poly with varying values of ‘C, ‘gamma’ and ‘degree’. This was done using the sickit learn.

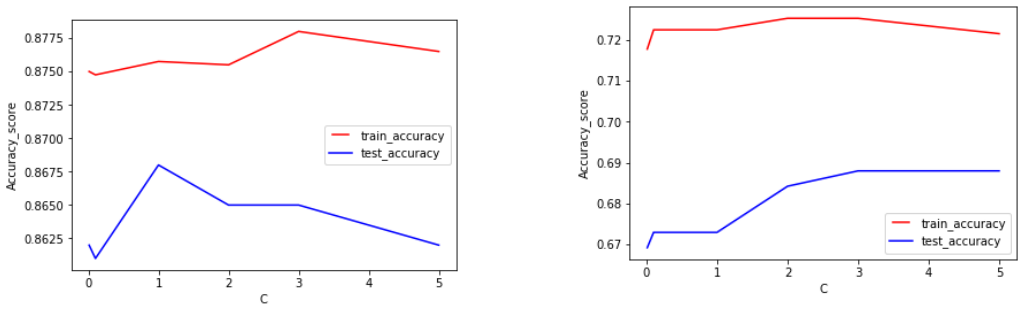

Figure 17 SVM accuracy score for Weather dataset (left) and NBA dataset (right) using linear kernel for different values of ‘C’.

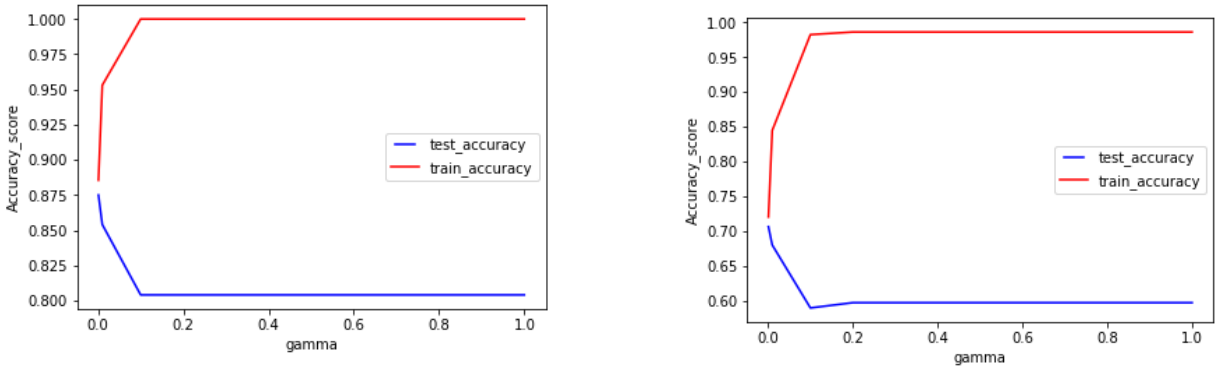

Figure 18 SVM accuracy score for Weather dataset (left) and NBA dataset (right) using rbf kernel for different values of 'gamma'.

Table 4 Optimized hyperparameters for SVM (only best accuracy obtained are listed)

Parameters	Weather Dataset	NBA Dataset
Kernel=Linear	Test accuracy=0.868, C=1	Test Accuracy=0.69, C=3
Kernel= rbf	Test accuracy=0.875, gamma= 0.001	Test Accuracy=0.62, gamma=0.001
Kernel= poly	Test did not converge in reasonable time	Test did not converge in reasonable time

The results of the SVM classification are summarized in Table 4. The parameters C, gamma or degree determines complexity of hyperplane for their respective models. For our data, the best accuracies were obtained for simple models and performance did not increase significantly on increasing the C, gamma or degree values. Also it is worthwhile to note that the SVM has a higher time complexity ($O(n^3)$). For this reason, the weather dataset was truncated to 2000 samples from an original sample size of 142,000 data points. NBA dataset as stated earlier is comprised of 1300 samples. We attempted to use 'poly' kernel with the truncated weather dataset. The algorithm did not converge in reasonable time and results are not reported here.

4.0 Summary and Conclusion

In this paper we attempted to solve a classification problem using five different algorithms. The hyperparameters were varied and performance was analyzed. Optimum set of process parameters were obtained for each algorithm for each dataset. Results for optimum parameters are summarized in Table 5.

Table 5. Summary of optimized parameters for the algorithms

Algorithm	Hyperparameters Weather Dataset	Weather Dataset Test accuracy	Hyperparameters NBA dataset	NBA Dataset test accuracy
Decision Tree	Tree Depth=6, Number of features for split~15	0.84	Tree_Depth=6,7 Number of features for split= 2-maximum attributes	0.67

K-Nearest Neighbors	K=6, weights= distance or uniform, p=1,2 or 3	0.84	K=8, weights= uniform, distance p=2 or 3	~0.75
Neural Networks	Number of layers=5, Number of nodes/layer=8, number of iterations= 300	0.84	Number of layers=7, Number of nodes per layer=7, number of iterations=300	~0.76
Boosting	Number of decision trees= 15, number of samples at split =100	0.85	Number of decision trees=15, number of samples at split =100	~0.67
Support Vector Machines	Kernel= rbf, gamma=0.001	0.88*	Kernel = linear, C=3	~0.69

*Only attempted for truncated dataset of 2000 samples.

Table 5 suggests that SVM performed slightly better on the weather dataset as compared to other algorithms. However this might be due to the smaller data size and increasing datasize might make SVM perform similar to other algorithms. Moreover SVM took significantly longer (~2 hours) as compared to other algorithms and hence might not be the best choice in this with this dataset. For the NBA dataset K-nearest neighbors and Neural Networks performed equally well. With regard to the time of convergence, both neighbors took similar time to converge. Thus Neural Network and K-NN might be better choice for the NBA dataset.

References (Articles referred)

1) An Effective Sampling Method for Decision Trees Considering Comprehensibility and Accuracy, Hyontai Sug, Wseas Transactions on Computers, Issue 4, Volume 8, April 2009.

2) An Analysis on Better Testing than Training Performances on the Iris Dataset, Marten Schutten, Marco A Wiering, Dutch Belgian Artificial Intelligence Conference (BNAIC), At Amsterdam, Nov 2016.

3) A new distance weighted K nearest neighbor classifier, Jianping Gou, Taisong, Journal of Information and Computational Science, November 2011.