**Analysis Of Various Supervised Learning Algorithms**

**Introduction:**

This document explores various machine learning algorithms which fall in the supervised learning category. In particular, the report performs an in depth analysis of the following algorithms:

1. Decision Trees
2. K Nearest Neighbor Algorithm
3. Ensemble Gradient Boosting Algorithm
4. Support Vector Machines
5. Artificial Neural Networks

Each of the above five algorithms were coded in Python using the pybrain and scikit-learn libraries except for pruning of decision trees, which was implemented in R because scikit-learn does not offer such capabilities.

For each of these algorithms, two datasets were used to compare and contrast their behavior under different circumstances. Most of the analysis revolves around exploring the learning curves of these algorithms to see how a range of different input parameters to the algorithms influence the mean square error (MSE) and the predictive capabilities of these algorithms.

**Datasets:**

Two datasets were chosen for this exercise which will be referred as the ‘Adult’ dataset and the ‘Car-Evaluation’ dataset. The training data for these datasets was downloaded from the University of California, Irvine’s website –

<https://archive.ics.uci.edu/ml/datasets/Adult>

<https://archive.ics.uci.edu/ml/datasets/Car+Evaluation>

The Adult and Car-Evaluation datasets were chosen because they contain contradicting number of features, number of training samples and the number of output classes. This would surely give us a lot of talking points in terms of how these factors influence the performance of various algorithms in terms of speed, mean square error and data cleaning operations.

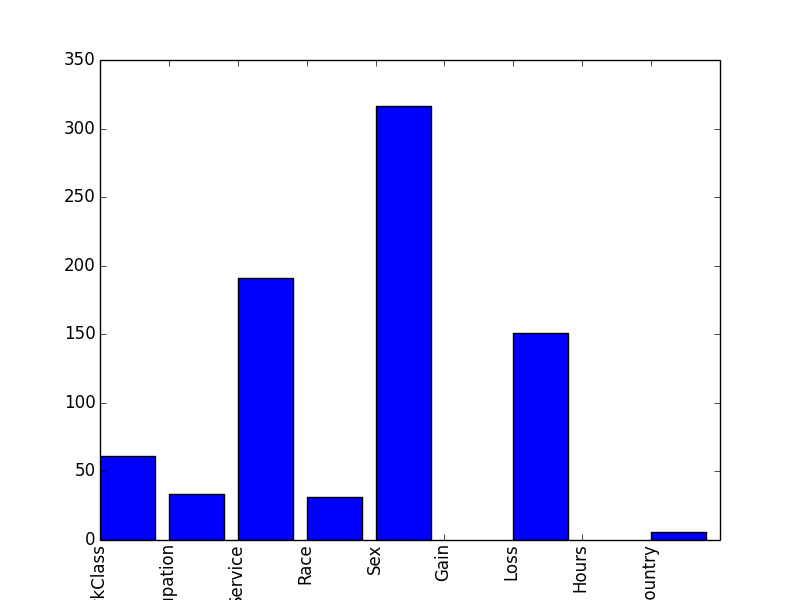
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Dataset Name** | **Features(#)** | **Training Samples(#)** | **Output Classes(#)** | **Missing Values** | **Requires Feature Scaling** |
| Adult | 12 | 30162 | 2 | Yes | Yes |
| Car-Evaluation | 6 | 1728 | 4 | No | No |

The Adult dataset contains a higher number of features and training samples compared to the Car-Evaluation dataset but it is only predicting on two output classes (more on that later) as opposed to four output classes respectively. One more aspect of the Adult dataset, which makes it more complicated is the fact that it contains missing values (denoted by ‘?’) and many features in the dataset require Z-score normalization so that the features can be rescaled to have the properties of standard normal distribution. It is also important to note that the string valued features were transformed to an integer enumeration so that the datasets could be fed into various learning algorithms.

**Adult Dataset:**

The dataset provides a set of thirteen input features describing various characteristics of the population from 1994 census. The job of our learning algorithms is to figure out whether a randomly picked individual is making over $50,000 or not. One feature was removed from the dataset because it did not add any value to the model because this feature manifested itself in other features. The twelve features used for this project were as follows : ['Age', 'WorkClass', 'Education', 'MartialStatus', 'Occupation', 'Service', 'Race', 'Sex', 'Gain', 'Loss', 'Hours', 'Country']. The output feature was denoted as [‘Salary’].

Features such as WorkClass, Education, MartialStatus, Occupation, Service, Race, Sex and Country were initially declared as strings in the dataset. Since each of these features contained a discrete number of elements, they were transformed into an enumeration of integers. The rest of the features such as Loss and Gain were re-scaled using Z-score normalization to values closer to each other and prevent high bias due to unscaled feature sets.

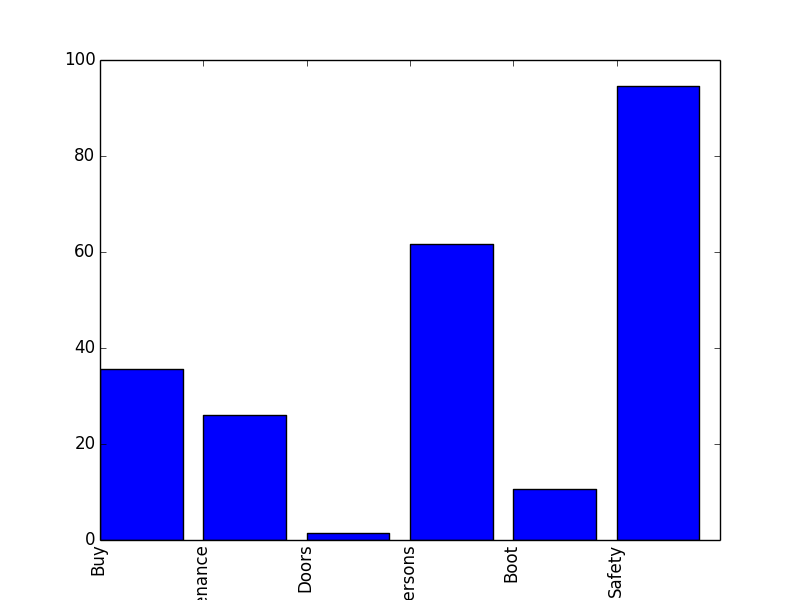


The above diagram illustrates the correlation of each input feature with respect to the output feature (Salary). It was computed using the sklearn’s SelectKBest and f\_classif algorithms. This data gives us insight into which features are the most important ones such as Sex, Loss (capital loss) and Service and the ones which can be discarded such as Hours and Country for certain algorithms.

**Car-Evaluation Dataset:**

The dataset provides a set of six input features describing various characteristics of a collection of cars. The job of our learning algorithms is to figure out whether a randomly picked car belongs to which category, which could be labeled as ‘Unacceptable’, ‘Acceptable’, ‘Good’ and ‘Very Good’ describing the quality of the car. The six features used for this project were as follows: ['Buy','Maintenance','Doors','Persons','Boot','Safety']. The output feature was denoted as [‘Rating’].

All of the input and output features were initially declared as strings in the dataset. Since each of these features contained a discrete number of elements, they were transformed into an enumeration of integers. There was no need to perform Z-score normalization because these values were of the same order.



Using the Python’s f\_classif algorithm, the above diagram illustrates the importance of each feature with respect to the quality of the car. Turns out that safety and the number of people it can accommodate are the two most important factors for a car to be classified as a certain type. Apparently, we can deduce that the number of doors a car features has negligible influence on the quality of the car.

**Decision Trees:**

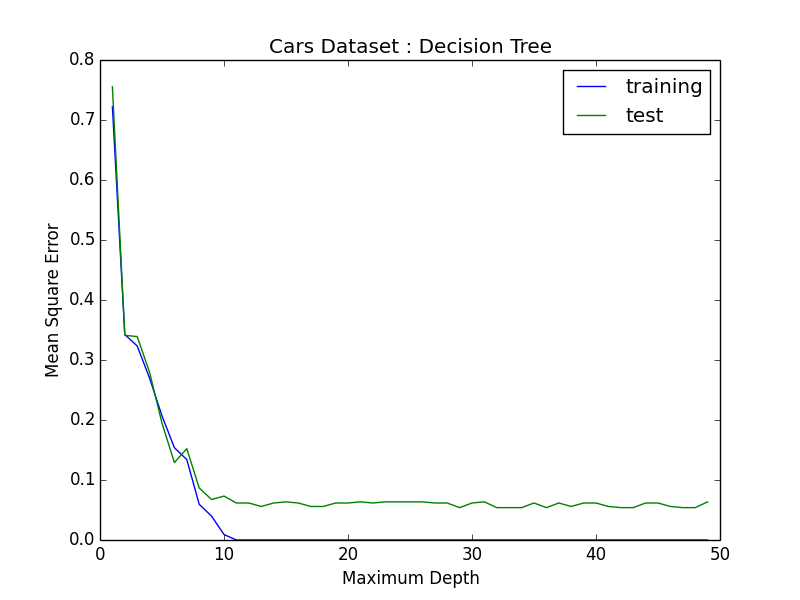
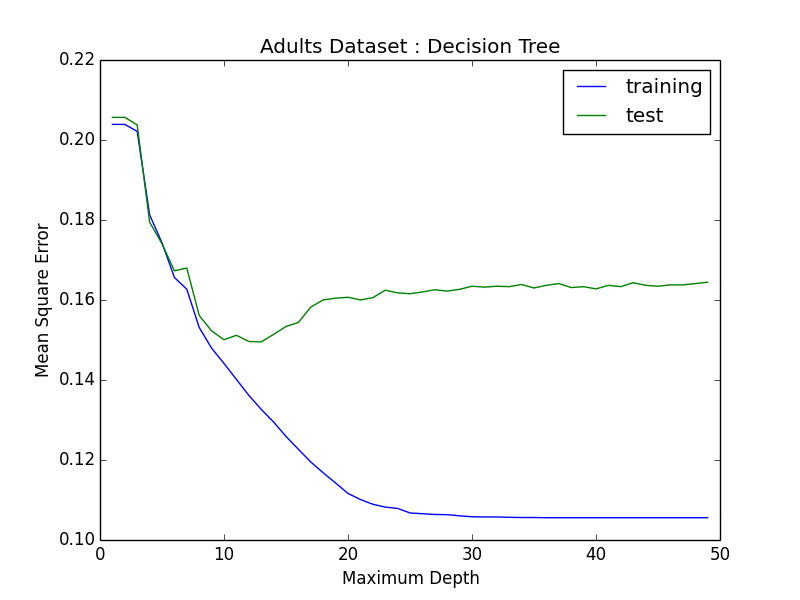
Decision Tree algorithm divides the input features in a structure which mimics tree. The idea is that an individual feature acts as the node and the value that it corresponds to acts as the vertice, which in turn acts as an input to another node in the tree which is one level below the parent note. The following diagram illustrates the decision tree of both datasets without pruning(to be explained later)

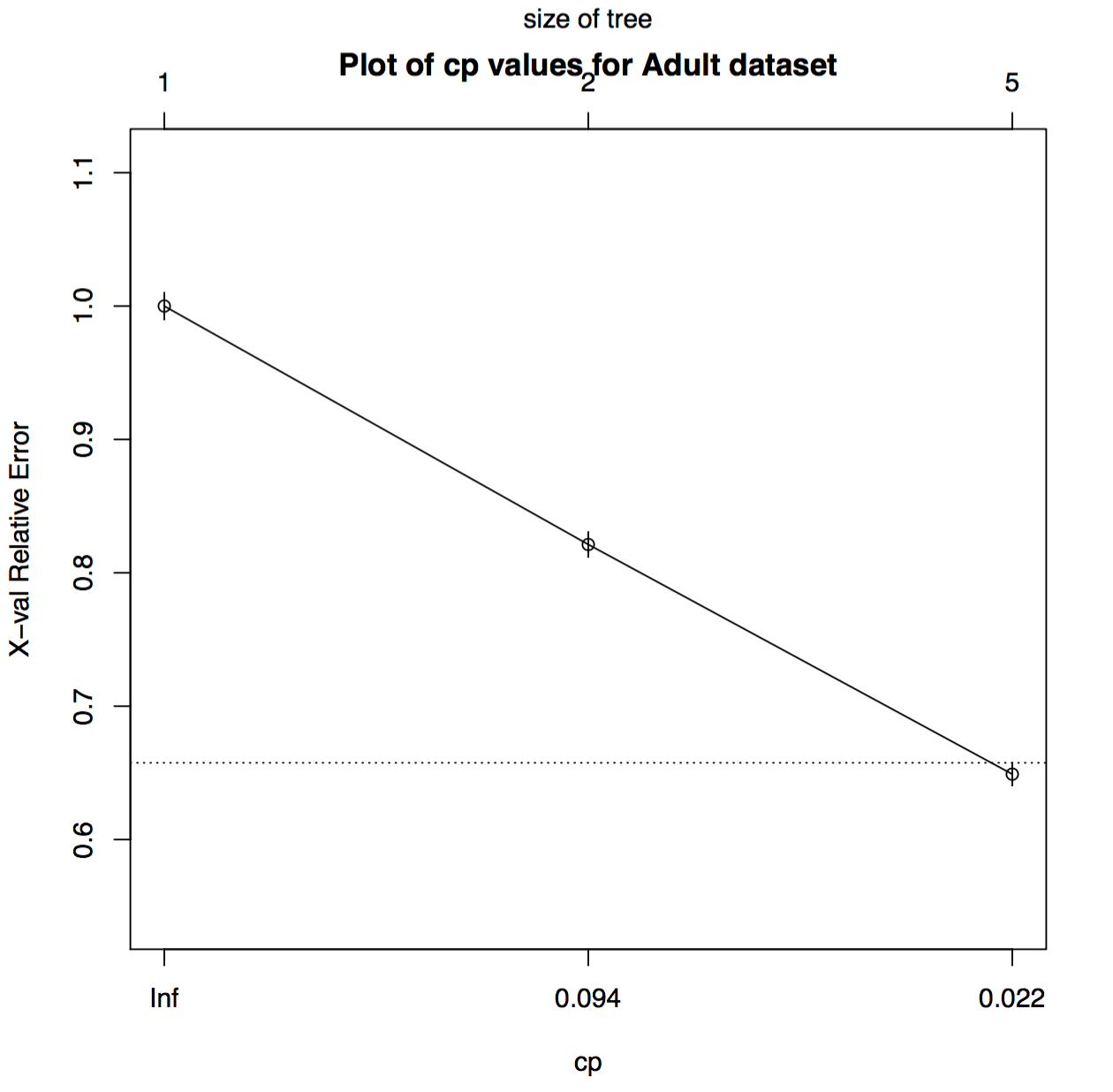
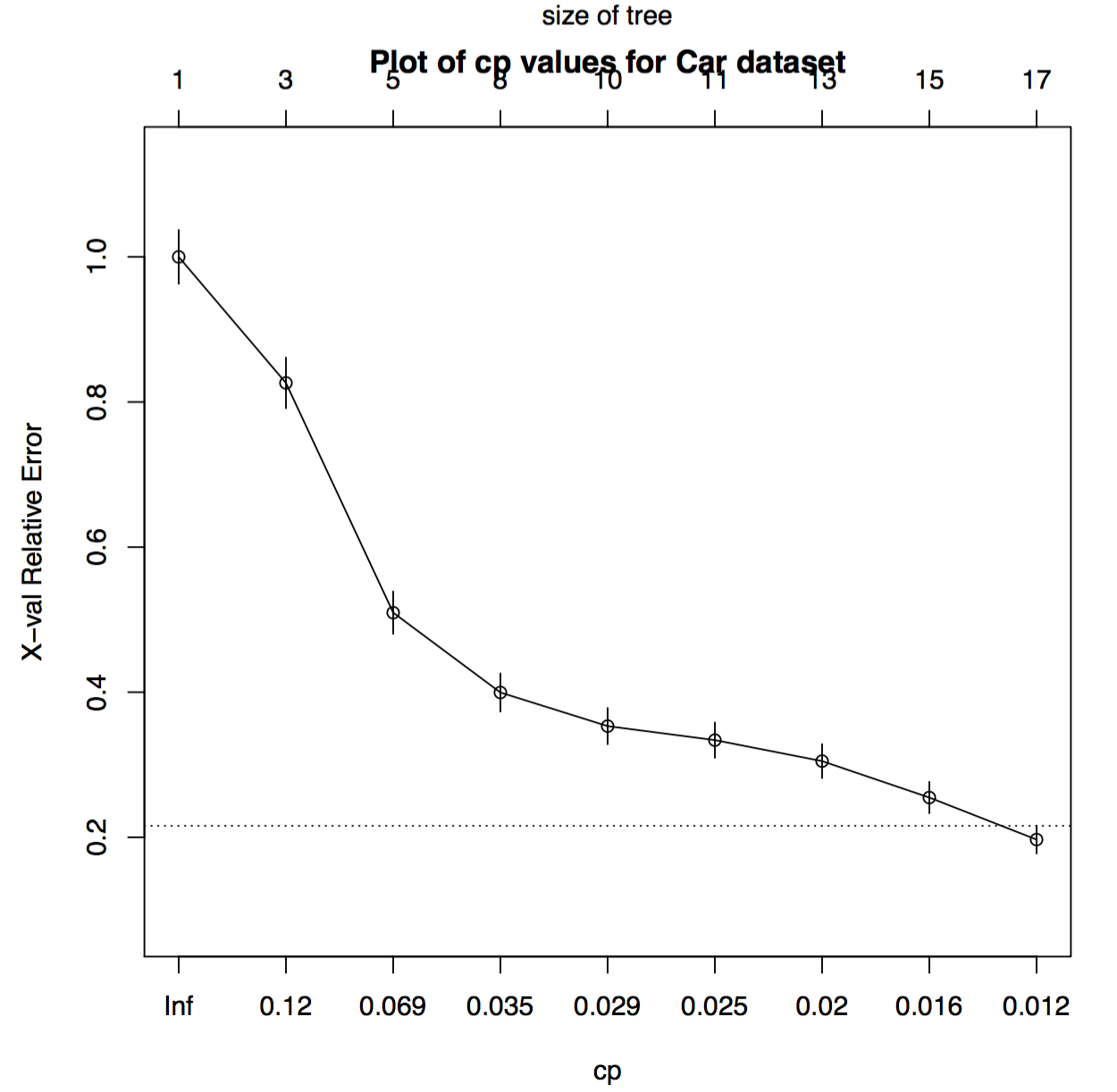
The above two figures show the decision trees for the Adult and Car-Evaluation datasets respectively. It can be observed that the adult dataset is essentially dependent upon only four characteristics, which are ‘Gain’, ‘Martial Status’ and ‘Education’. The first decision point divides our options in a 75/25 split whereby a randomly chosen individual who had capital gains with a normalized value of more than 0.54 was destined to make more than $50,000. This makes sense from a logical point of view also. For example, if an individual had a considerable capital gain throughout his/her life, it can be assumed that they were most likely making huge sums of money in their day job. Of course, there would be some outliers such as individuals who inherited a lot of wealth or won a lottery, which could account for the 95% accuracy of this prediction on the cross validation set. In case an individual did not have considerable amount of capital gain (<0.54 normalized) and his/her martial status had a value of less than 0.5 (normalized), we can predict with 96% confidence that the individual made less than $50,000. Similarly, an individual with less capital gains, higher value of martial status and lower value of education (<0.93) could be assumed to have made less than $50,00 with 71% accuracy. Note that the depth of this tree is not very high.

The cars dataset looks more complex because it has four possible outcomes, namely acceptable, not acceptable, good and very good, which denotes the quality of the car. Following the same logic as we followed for the Adults example, it is easy to see that the number of persons that a car can accommodate plays a major role in splitting our decision tree. Thereafter, factors such as safety, expensiveness and maintenance costs play a major role in splitting the predictions.

By examining these two decision trees, we can conclude that ***more the number of output classes, the more complex is the tree structure***.



The above two plots explore the variation of the maximum depth of a decision tree and its influence on our ultimate goal, the mean square error of the predictive abilities of the training set and the three fold cross validation set. As can be observed above, very low values of maximum depth such as one or two correspond to a very high bias region, where the training dataset is under-fitting the decision tree and results in a high value of mean square error for both the training set and cross validation set. On the other extreme, very high value of maximum depth such as 50 is over-fitting both the Adults and Car-Evolution datasets because the MSE is very close to 0 and almost constant whereas the cross validation error is consistently higher. An interesting region of the ‘Adult’ dataset’s graph is the location where maximum depth varies from 10 to 30. A U-shape for the cross validation set implies that we went from high bias to optimal (the lowest point in the U-shape) and ultimately to a high variance region. For the adults dataset, an optimal value for the maximum depth would be around 15 because this is the point where both cross validation and training errors are minimized simultaneously. As far as the Car-Evaluation dataset is concerned, the graph enters from high bias to an optimal range after reaching a maximum depth of 10 without the U-shape curve. We can attribute this fact to the quirks of the dataset. A pruned version of both the decision trees were implemented in R-language and the result (which we will go over at the end of this document) was that we were able to bring down the cross validation error with pruning. The two graphs below show a plot of complexity parameter (cp) for pruning with respect to the error in the predictions. A cp value of 0.01 will satisfy our pruned tree.

  
**K-Nearest Neighbors:**