Abstract

The paper is intended to compare the most two widely-used classification algorithms in data mining: C5.0 by Quinlan and CART by Bremain. By using three different datasets (Iris flower, Titanic and Pima Indians Diabetes datasets), I have discovered some special cases in which either one of them is significantly outperforming the other.

Introduction

Classification of data into categories is one of the fundamental tasks in data mining. Although classification has been studied extensively, only few of classification algorithms have an ability to extract meaningful knowledge from massive datasets. Among them, in this research it compares on the two most widely – used classification algorithms: CART and C5.0 under two main criteria. First, a comparison of classification capabilities between the two algorithms will ascertain how well they will classify given instances into their pre-determined categories. Second, a comparison of generalization capability will determine how well their trained system is used to predict the categories of unseen data when only the input variables are given. Beyond this point, the structure of the paper is divided into three sections as follows: the first section is *Literature Review*, which concentrates on the developing concepts of CART and C5.0 algorithms while the second sections *Methodology & Results* describes the materials and methodology used in this research and the results of calculations will be presented for further discussion. In this third section, *Discussion & Conclusion*, the obtained results from the second section will be fully analyzed in order to draw the insights between the two algorithms.

Literature Review

1) Decision Tree Induction

Decision Tree induction is a classification pattern recognition approach that is applied in both CART and C5.0 algorithms. A decision tree is an inductive inference method widely used in a supervised classification learning technique for "[classifying] instances by sorting them down the tree from the root to some leaf node, which provides the classification of the instance. Each node in the tree specifies a test of some attribute of the instance, and each branch descending from that node corresponds to one of the possible values for this attribute. An instance is classified by starting at the root node of the tree, testing the attribute specified by this node, then moving down the tree branch corresponding to the value of the attribute. This process is then repeated for the subtree rooted at the new node" (Bahety, 2014). The construction of a decision tree required two conceptual phases: growing and pruning.

By its nature, the decision tree classifier is a greedy algorithm because the tree is grown by "recursively splitting the training set based on local optimal criteria until all or most of the

records belonging to each of the partitions bearing the same class label". With that respect, the tree usually contains overfitting data. Overfitting the data occurs when a predictive model would classify perfectly the known data, but fail to classify anything useful on the yet-unseen data. Thus the attempt to make a tree not too closely taking on inaccurate data that can infect itself with substantial errors and reduce its predictive power is called pruning. The pruning phase handles the problem of over-fitting the data by removing the noise and outliers, which eventually increases the accuracy of the classification.

2) CART Algorithm

CART, an abbreviation of Classification And Regression Trees, was first introduced by Bremain (year) which is a binary tree using GINI Index as its splitting criteria. CART can handle both nominal and continuous attributes to construct a decision tree. Also, it can handle missing values by surrogating tests to approximate outcomes. In the pruning phase, CART uses pre-pruning technique called Cost – Complexity pruning to remove redundant braches from the decision tree to improve the accuracy. In Breiman's explanation, the Cost-Complexity pruning "proceeds in two stages. In the first stage, a sequence of increasingly smaller trees are built on the training data. In the second stage, one of these tree is chosen as the pruned tree, based on its classification accuracy on a pruning set. Pruning set is a portion of the training data that is set aside exclusively for pruning alone". In other words, CART adopts a cross –validated method in its pruning technique.

3) C5.0 Algorithm

C5.0 algorithm is a successor of C4.5 algorithm, both were first developed by Quinlan (1994). A few changes in C5.0 have made it to outperform C4.5 on the efficiency and the memory. Unlike CART only gives a binary tree, C5.0 can generates multi-branch tree in a situation where one or more nominal inputs are given. Similar to its previous versions (C4.5 & ID3), C5.0 use an information -based criterion (a.k.a Entropy or Information Gain) as an attribute selection measure to build a decision tree. For overfitting avoidance, C5.0 use a pessimistic pruning approach, Rule-post pruning, to remove unreliable branches from the decision tree to reduce the size of the tree without any loss of its predictive accuracy. The Rule – post pruning starts off by converting a decision tree to an equivalent set of rules, then based on statistical confidence estimations for error rate it evaluates the rules with the aim of simplifying them without affecting the accuracy. Unlike CART pruning technique in which it uses a portion of the training set for validation, C5.0 pruning technique adopts the Binomial Confidence Limit method in which it allows all of the available labeled data to be used for training. In a case of handling missing values, C5.0 allows to whether estimate missing values as a function of other attributes or apportions the case probabilistically among the results.

Methodology & Results

The both algorithms walk on two different paths to build a decision tree and pruning it. With several respects of the differences between C5.0 and CART, it probably causes one algorithm to outperform the other. In this section, the two algorithms will be put on a ring to compete one another under two comparative criteria: classification capacity and generalization capacity. The materials consist of three particularly chosen datasets: Iris flower data, Titanic data and Pima Indians Diabetes data; which are ranked ascendingly from simplest to hardest respectively. Below is a block diagram describing the research methodology conducted in this paper.

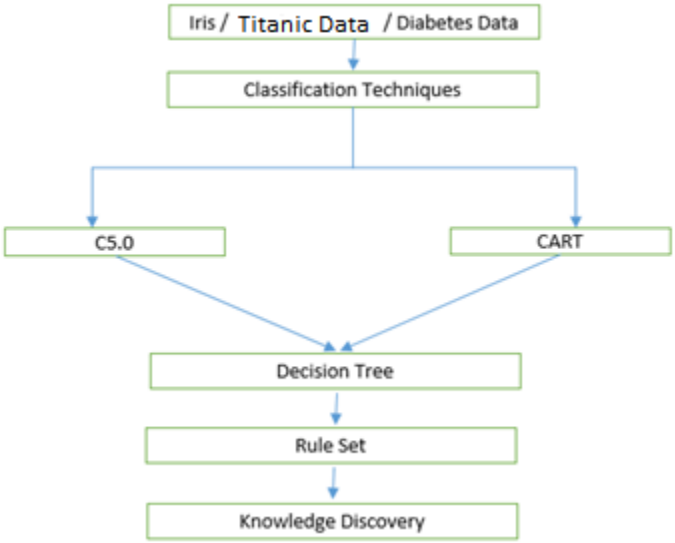


Figure 1: Block Diagram describing the research methodology.

The block diagram can be summarized in three simple steps:

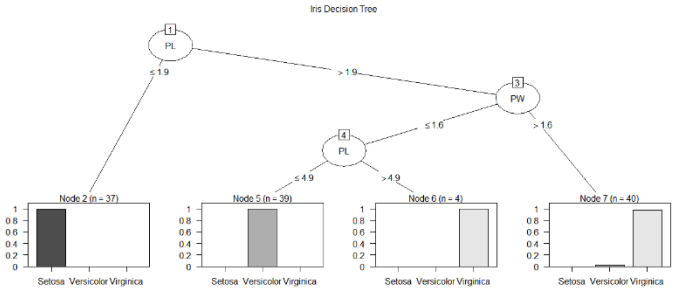
- Step 1: Pre-proceed the datasets. Detailed descriptions and assumptions for each dataset will be provided so that the inputs and output are clearly stated.
- Step 2: Implement the two algorithms on the dataset for building decision trees, then convert them to rule sets for comparison purposes. And measure the performance of C5.0 and CART algorithm under proposed criteria, then records the results of calculations for further comparative analysis.
- Step 3: Analyze the results.

1) Comparative Study of C5.0 and CART using Iris Flower Data

The Iris flower data set or Anderson's Iris data set is a multivariate data set consisting of 50 samples from each of three species (Setosa, Virginica, and Versicolor). And each sample is explained by 4 numerical attributes: Sepal Length, Sepal Width, Petal Length and Petal Width. Similar to human learning behaviors, machine learning relies on past experiences to make judgements. The more experiences it accumulates the better judgements are made.

Following that ideology, I use 120 instances or 80 percent of the Iris data as a training set to construct decision trees for each algorithm, and the remaining 20 percent or 30 instances as a test set to evaluate the accuracy of the trees.

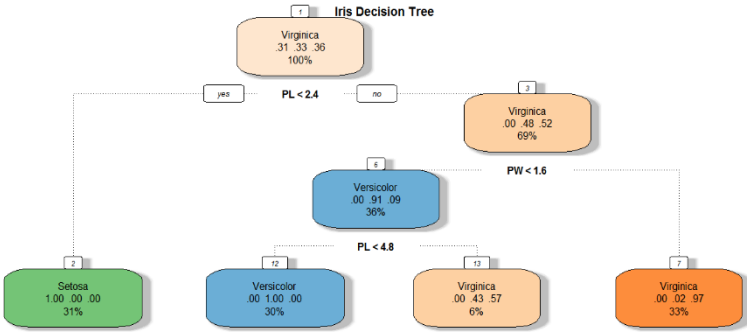
1.1) *The Construction of Decision Trees in CART and C5.0 Using the Training Set.*




```
Decision Tree:

1) root 120 77 Virginica (0.308 0.333 0.358)
   2) PL< 2.45 37  0 Setosa (1.000 0.000 0.000) *
   3) PL>=2.45 83 40 Virginica (0.000 0.481 0.518)
     6) PW< 1.65 43  4 Versicolor (0.000 0.906 0.093)
      12) PL< 4.75 36  0 Versicolor (0.000 1.000 0.000) *
      13) PL>=4.75 7  3 Virginica (0.000 0.428 0.571) *
     7) PW>=1.65 40  1 Virginica (0.000 0.025 0.975) *
```

Figure1: Iris Decision Tree & Ruleset by C5.0 algorithm




```
Decision tree:

PL <= 1.9: Setosa (37)
PL > 1.9:
:...PW > 1.6: Virginica (40/1)
    PW <= 1.6:
    :...PL <= 4.9: Versicolor (39)
        PL > 4.9: Virginica (4)
```

Figure2: Iris Decision Tree & Ruleset by CART algorithm

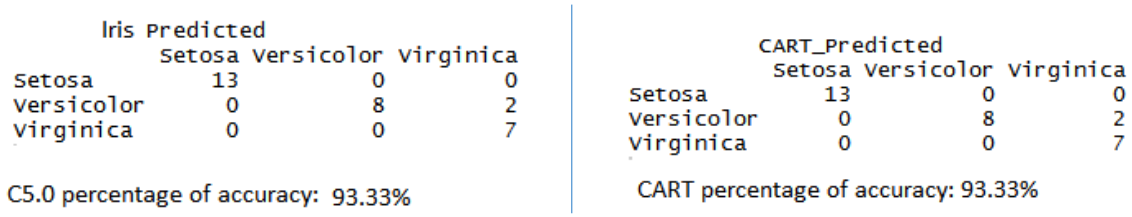
According to "Theoretical Comparison between the Gini Index and Information Gain Criteria" paper by Laura Elena Raileanu (2004), they found that the percentage of disagreement of the Gini Index function and the Information Gain function is never higher than 2%, which explains that there is no significant difference between the two criteria. In other words, it is impossible to empirically conclude which one of the two algorithms to prefer.

Through preliminary analysis of the results in Figure 1 &2, the general structure, size and attribute selections of the decision trees are the same. Both decision trees have a binary structure,

tree size of 4; and their attribute selections include Petal Length as root nodes and Petal Weight as leaf nodes.

However, the values of which selected attributes that C5.0 and CART algorithms decide to do the splitting are relatively different. In C5.0 algorithm, the value of Petal Length in its first split is 1.9 whereas it is 2.45 in its counterpart. The main logic behind this difference is: "the test selection criterion of C5.0 is an information – based criterion (Information Gain), whereas CART's is based on a diversity index (GINI index)". With a respect of classification capacity, it seems like the C5.0's decision tree has done a better classification than its counterpart's. Because the C5.0's decision tree has successfully classified 37 Setosa, 43 Virginica, 39 Versicolor and 1 misclassification, whereas the CART's decision tree has classified 37 Setosa, 43 Virginica, 36 Versicolor and 4 misclassifications.

1.2) *The Predictive Accuracy of Decision Trees Using the Test Set*.



Iris	Predicted		
	Setosa	Versicolor	Virginica
Setosa	13	0	0
Versicolor	0	8	2
Virginica	0	0	7

C5.0 percentage of accuracy: 93.33%

CART_Predicted			
	Setosa	Versicolor	Virginica
Setosa	13	0	0
Versicolor	0	8	2
Virginica	0	0	7

CART percentage of accuracy: 93.33%

Figure 3: Comparison of the predictive accuracy percentage

Interestingly enough, although there is a slight difference in the splitting value of selected attributes between C5.0 and CART algorithms, the both decision trees have yielded the same percentage of accuracy, 93.33%, which means that they are very good at predicting the categories of unseen data when only the input variables are given.

Conclusion:

The result analysis of Iris Data has clearly shown that the two algorithms: C5.0 & CART have a unanimous agreement on attributes selections, but a slight disagreement on the splitting value of selected attributes. Due to this disagreement, the C5.0 decision tree has lesser misclassifications than the CART's with a respect of classification capacity. But with a respect of generalization capacity, they both have the same percentage of predictive accuracy, 93.33%, which explains there is not empirical evidence to conclude which one of the two algorithms works better.

2) Comparative Study of C5.0 and CART using Titanic Data

The Titanic dataset describes the survival status of individual passengers on the Titanic. The dataset frame contains 1309 instances on the following 14 variables: pclass, survived, name, sex, age, sibsp, parch, ticket, fare, cabin, embark, boat, body, and home.dest. The dataset frame only

contains information for the passengers, but not for the crew; and 20 percent information of the passenger age is missing. The below table contains the variable descriptions:

VARIABLE DESCRIPTIONS

Pclass	Passenger Class (1 = 1st; 2 = 2nd; 3 = 3rd)
survival	Survival (0 = No; 1 = Yes)
name	Name
sex	Sex
age	Age
sibsp	Number of Siblings/Spouses Aboard
parch	Number of Parents/Children Aboard
ticket	Ticket Number
fare	Passenger Fare (British pound)
cabin	Cabin
embarked	Port of Embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)
boat	Lifeboat
body	Body Identification Number
home.dest	Home/Destination

For simplicity, the instances that contains missing values will be eliminated from the dataset in order to fairly compare the two algorithms' performance on classifying nominal attributes. Also, some conversions and modifications of the dataset are done to facilitate the process of building decision trees. It is described as follows:

A	B	C	D	E
Field	**Modification**		**Field**	**Modification**
Name	Ignored		Pclass	Normalized into 1st, 2nd and 3rd
Passenger ID	Ignored		Sibsp	Normalized into None, Lessthan3, Morethan3
ticket	Ignored		Parch	Normalized into None, One, and Both
fare	Ignored		Age	Normalized into Child, Adolescent, Adult and Old
cabin	Ignored		Embark	Normalized into S.amp, Cher, and Queen
home.dest	Ignored		Survived	Normalized into YES and NO
boat	Ignored			
body	Ignored			

Upon conversion, the final dataset is left with 1046 instances described by 6 nominal attributes, I use 60 percent of them to train and the other remaining 40 percent to test. The table below is a glimpse of the modified dataset.

	A	B	C	D	E	F	G
1	**sex**	**Age**	**embarked**	**pclass**	**sibsp**	**parch**	**Survived**
2	female	Adult	Southampthon	2nd	None	One	YES
3	male	Old	Cherbourg	1st	None	None	NO
4	male	Adult	Southampthon	3rd	Lessthan3	Both	NO
5	male	Adult	Southampthon	1st	None	None	YES
6	female	Adult	Southampthon	1st	None	Both	YES
7	female	Adult	Cherbourg	1st	Lessthan3	None	YES
8	male	Adult	Queenstown	3rd	None	None	YES

2.1) *The Construction of Ruleset in CART and C5.0 Using the Training Set*

```
Decision tree:

sex = male:
:...Age in {Adolescent,Adult,Old}: NO (370/70)
:   Age = Child: YES (21/6)
sex = female:
:...pclass in {1st,2nd}: YES (143/12)
    pclass = 3rd:
    :...embarked = Cherbourg: YES (13/1)
        embarked in {Queenstown,Southampthon}:
        :...sibsp in {Lessthan3,Morethan3}: NO (32/11)
            sibsp = None: YES (49/23)
```

Figure 4: C5.0 Algorithm's Ruleset (already pruned)

```
1) root 628 265 NO (0.57802548 0.42197452)
  2) sex=male 391  85 NO (0.78260870 0.21739130)
    4) Age=Adolescent,Adult,Old 370  70 NO (0.81081081 0.18918919) *
    5) Age=Child 21   6 YES (0.28571429 0.71428571) *
  3) sex=female 237  57 YES (0.24050633 0.75949367)
    6) pclass=3rd 94  45 YES (0.47872340 0.52127660)
     12) embarked=Queenstown,Southampthon 81  37 NO (0.54320988 0.45679012)
       24) sibsp=Lessthan3,Morethan3 32  11 NO (0.65625000 0.34375000) *
       25) sibsp=None 49  23 YES (0.46938776 0.53061224)
         50) embarked=Queenstown 13   4 NO (0.69230769 0.30769231) *
         51) embarked=Southampthon 36  14 YES (0.38888889 0.61111111) *
     13) embarked=Cherbourg 13   1 YES (0.07692308 0.92307692) *
    7) pclass=1st,2nd 143  12 YES (0.08391608 0.91608392) *
```

Figure 5: CART Algorithm's Ruleset (not pruned yet)

As expected, the both algorithms have indicated that Sex and Age clearly have the most significant relationship demonstrated within the dataset for survival rate. In my opinion, one thing that stands out the most in the both Figure 4 and 5 is that upper class female and children passengers have the highest percentage of survival, whereas male passengers were the least likely to survive.

With a respect of classification capacity, it seems like C5.0 has more misclassifications than its counterpart (19.6% error rates in C5.0 compared to 18.8% error rates in CART). The table below will show the numbers of correct classifications / misclassifications for each algorithm.

Comparsion of Classifcation capacity		
C5.0 / CART	**NO**	**YES**
NO	321 / 330	42 / 33
YES	81 / 85	184 /180

Again, CART uses the Gini Index, which can be understood as a criterion to minimize the probability of misclassification therefore we see that the CART algorithm tends to have less misclassifications than C5.0 in large datasets. However, it is not the case to conclude that CART is more preferable than C5.0 in extracting knowledge from large datasets because of its lower probability of misclassifications. If I am doing an exploratory data analysis, I would prefer the information gain (C5.0) so that I maximize mutual information with my tree.

2.2) *The Predictive Accuracy of Decision Trees Using the Test Set*.

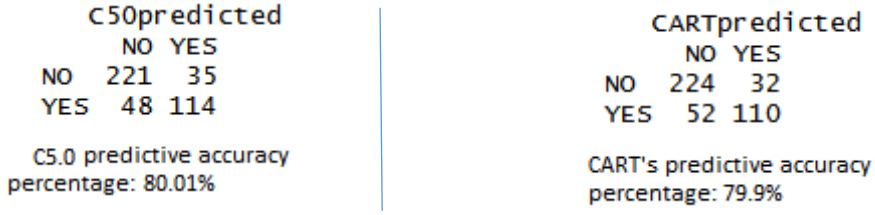


Figure 6: Comparison of the predictive accuracy percentage.

With a respect of generalization capacity, C5.0 algorithm have a slightly stronger predictive power than its counterpart.

Conclusion: We compared C5.0 and CART's performance both in terms of classification, and generalization on the modified Titanic dataset (all inputs are normalized). The results of analysis have found that CART has a better classification but lower generalization accuracies compared with C5.0. And another thing I have also noted in Figure 6 that the prediction errors made by each algorithms are different. Because of that, I ask myself if there could be a possible way of combining these algorithms so that their predictions accuracies could be improved. For now, I

will not discuss any of that matter, but I hope that this matter will present a challenge for future research.

3) ***Comparative Study of C5.0 and CART using Prima Indians Diabetes Database***

A total of 768 instances in Prima Indians Diabetes Database described by the 9 following attributes: number of times pregnant, Plasma glucose concentration, Diastolic blood pressure (mm Hg), Triceps skin fold thickness (mm), Serum insulin (mu U/ml), BMI, Diabetes pedigree function, Age (years), Class variable (Sick or Healthy). Roughly 49% of the dataset contains missing values; which I think if we exclude them from the dataset and then our tree models based on the remaining, the results of our tree models will not be the same. Fortunately, C5.0 and CART algorithms give us two options to deal with an example containing missing values.

Option 1: The both algorithms will first identify missing

values then delete them and trained the trees based on the remaining instances. After deleting these cases there are 392 cases with no missing values (130 Healthy cases and 262 Sick cases).

Option 2: CART looks for "surrogate splits" that approximate the outcomes when the tested attribute has a missing value; whereas C4.5 approaches to missing attribute values by splitting cases with missing attribute values into fractions and adding these fractions to new case subsets.

The dataset is divided into training and test sets using 60-40 ratio.

3.1) Deleting Instances with Missing Attribute Values.

The first comparison test is conducted to measure the generalization accuracies between C5.0 and CART when all missing values are eliminated from the dataset. The below figure gives a summary of their results.

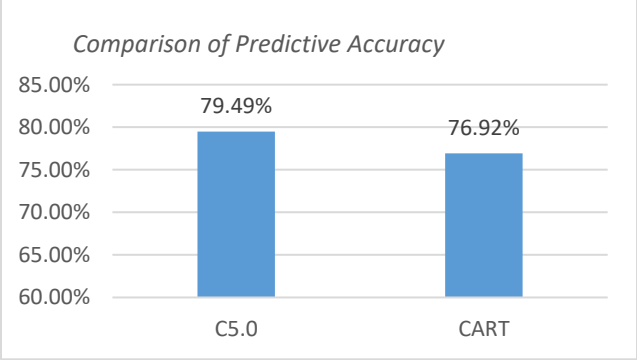


Figure 7: Comparison of the predictive power between C5.0 & CART (without missing values)

Similar to the Titanic dataset, C5.0 outperforms the CART with a respect of generalization capacity.

3.2) *Including Instances with Missing Attribute Values*

The second comparison test is conducted to measure generalization accuracies between C5.0 and CART when all missing values are included.

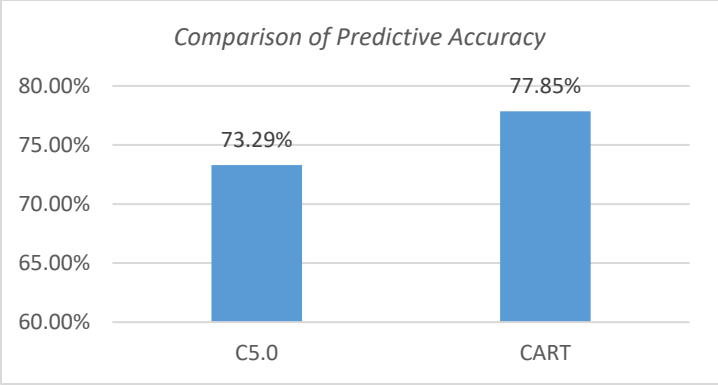


Figure 8: Comparison of the predictive power between C5.0 & CART (with missing values)

Surprisingly, the "surrogate split" method in CART has performed better than the "probabilistic imputation" in C5.0. Through the analysis of Figure 8, when CART and C5.0 have the same complete data performance, surrogate split has outplayed its counterpart. Its advantage is larger because the missing value instances mainly accounts for 49% of the dataset and the CART algorithm is more likely to find a good surrogate when there are more predictors to choose from.

Conclusion: Through the results of Figure 7 & 8, we might conclude that missing data have a tremendous impact on our tree models. Whether we should exclude or include missing data on our dataset is still a complex question to answer. In case of excluding them, CART's tree model has a less predictive power than C5.0's. But in case of including them into our dataset, which of the handling missing data methods we should use. In this paper, I have introduced you the two methods: surrogate splits by CART and probabilistic imputation by C5.0. And I also recommend you to use CART algorithm to handle a missing value dataset only when the missing value data account for more than 40% of the dataset.

Summary

There are well-known differences between CART and C5.0:

1) C5.0 can have a multiway splitting or binary decision tree, whereas CART only gives a binary tree.
2) C5.0 use Information Gain or Entropy as an attribute selection measure to build a decision tree while CART use Gini index.
3) For the pruning process, CART uses pre-pruning technique called Cost – Complexity pruning to remove redundant braches from the decision tree to improve the accuracy,

whereas C5.0 pruning technique adopts the Binomial Confidence Limit method to reduce the size of the tree without any loss of its predictive accuracy.

4) Finally, in a problem of handling missing values CART surrogates test to approximate outcomes while C5.0 apportions values probability among outcomes.

Although they seem to have a lot of differences, their results and accuracy are quite similar. In Iris dataset, we conclude that C5.0 & CART have a unanimous agreement on attributes selections; but a slight disagreement on the splitting value of selected attributes. In the modified Titanic dataset (all the inputs are normalized), we find that C5.0 usually has more misclassifications than CART with a respect of classification capacity. Because of that finding, we conclude if someone is interested doing an exploratory data analysis, he/she should prefer the information gain (C5.0) so that he/she maximize mutual information with his/her tree. And if someone is interested in building a tree model with the lowest probability of misclassification, he/she should prefer CART algorithm. In Pima India Diabetes, we conclude that when the missing values are eliminated, the C5.0 tree model has a stronger predictive power over its counterpart. However, CART algorithm is more preferable to handle a missing value dataset only when the missing value data account for more than 40% of the dataset. There is one concern arisen while I was doing the comparative study of C5.0 and CART algorithms. My concern is indicated in the Titanic dataset's conclusion, because the prediction errors made by each algorithms are different, so I ask myself there is a possible way to combine the two algorithms together so that their predictions accuracies could be improved.

Works Citation

Breiman, L., Friedman, J., Olshen, R., Stone, C., Steinberg, D., & Colla, P. (1983). CART: Classification and regression trees. Wadsworth: Belmont, CA, 156.

Bahety, A. (2014). Extension and Evaluation of ID3–Decision Tree Algorithm. Entropy (S), 2, 1.

Raileanu, L. E., & Stoffel, K. (2004). Theoretical comparison between the gini index and information gain criteria. Annals of Mathematics and Artificial Intelligence, 41(1), 77-93.

Quinlan, J. R. (1993). C4. 5: Programming for machine learning. Morgan Kauffmann.