**Name: VINAY KUMAR SINGH Net ID : vks180002**

**CRICKET DIARIES: MATCH PREDICTION USING DECISION TREE**

**Sports Analytics**

A very interesting branch of analytics which is yet to be explored fully. Sports Analytics was made famous when Statistics and Base probability was applied to the movie Money Ball which is one my favorite movie and a delight to watch. Through these years, technology has been moving fast and fields such as Healthcare Analytics, Retail Market Analytics has shown a lot potential application, algorithms and machine learning has transformed these areas. Though Sports Analytics has shown a lot improvement and advancement but still this interesting field has been lagging in terms of applications in real sports. Take an example about Master’s, there are a very few courses which help get fresh talent in the application based interesting field of Sports Analytics.

**Goal:**  The goal of this article is to make you understand the usage of Machine Learning Algorithm called as Decision Tree or the CART model in Cricket Analytics. Predicting the match outcome (for a particular team) using CART model using a few historical metrics of the team will be our primary focus in this in article. This can be leveraged by the Cricket fans in making prediction about their team winning or losing a match.

**Cricket Overview**

Cricket is a team game consisting of two competing team with 11 players each quite similar to baseball but not exactly baseball. Three major department of cricket include Batting, Fielding and Bowling. Batsmen are responsible for scoring runs for a team by hitting them with a bat after a Bowler of the opposite team bowls. Fielder are placed within the field for getting the batsman out and stopping the number of runs scores by the Batting team.

**Dataset Preparation**

This was the task which is the most time consuming and complex process**.** Usually, when you are building a model from scratch, it takes about 70 % of your total time. This stage is the most important stage of any Model building activity and hence is time consuming. There a lot of steps and iteration that needs to be followed listed below

* Metric collection: In a real world scenario all the data is not readily available, first step is to understand the problem and derive metrics that might affect the model
* Scraping Data: After you are done with the Metrics collection, we need to use web scraping or other sources to get your data about the metrics decided in the first step
* Cleaning Data: There’s a lot of cleaning and parsing the data that needs to be done before you start building the model
* Derived Metrics: Using all the metrics scraped, we need derive a few metrics by combining/ altering data into a more useful format.

We need to use a 70-30 rule here, 70 % data which is scraped is not useful or removed from the dataset.

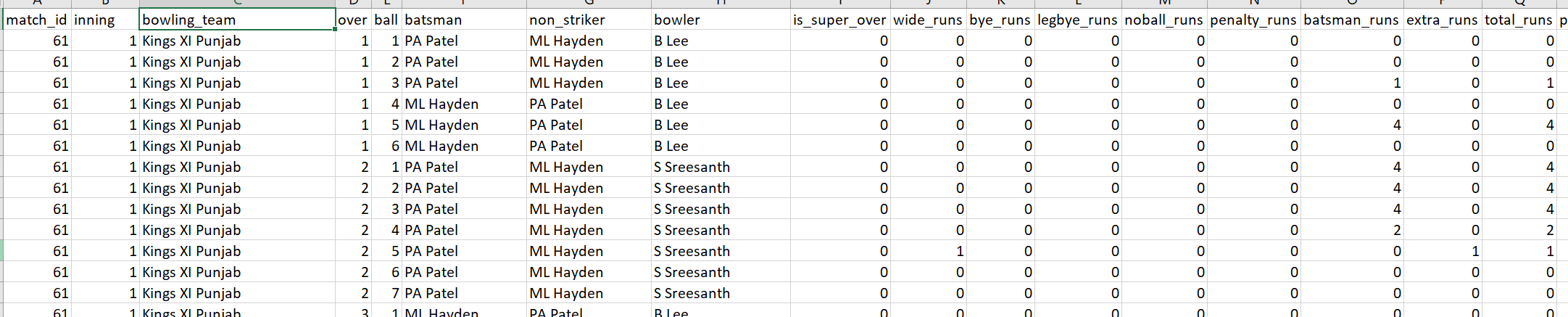
**Cricket Data**

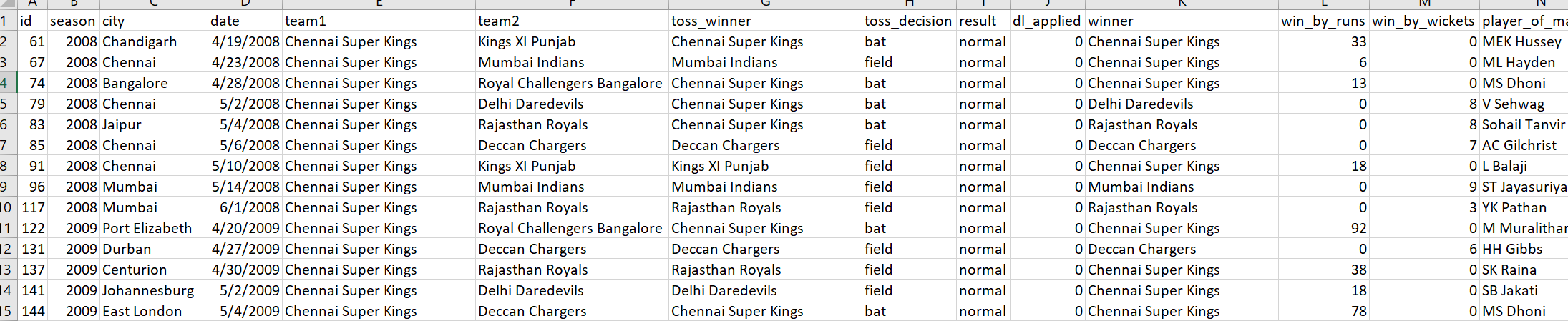
**Raw Data**

Major portion of my data has been scraped from cricket website like espncricinfo and cricsheet.org. I have also taken Indian premier league data from Kaggle competition to run my model. I used the similar approach listed above about first defining a list of metrics, scraping and cleaning the data.

Below is the image and description of the raw data that we have considered:

1. Deliveries data: This dataset consist of ball by ball Indian Premier League data of all the matched from 2008-2017.
2. Matches data: This dataset contains list of all the matches played in IPL with competing team, venue for match, winning team as few of the metrics from 2008-2017

Deliveries Data

Matches Data

**Data Transformation**

We will be only focusing on building a model which will predict the match outcome of Chennai Super Kings. Therefore, we have only included matches played by Chennai Super King team in IPL. Below is the image of the transformed data:

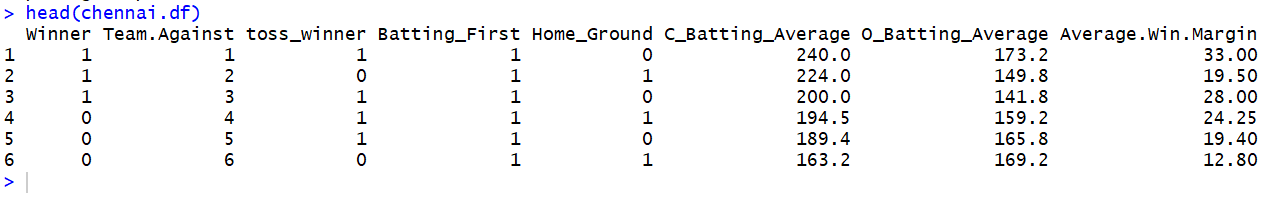
**Metrics Explanation**

Categorical data has been transformed to numerical quantitative variables for simplification:

* Winner: This metrics represent ‘1’ when Chennai Super Kings have won a match and ‘0’ when it lost the match.
* Team Against: This represent the team against which Chennai Super Kings is playing. Below is the index that can be used to get the mapping
* Toss Win: It depicts which team has won the toss. It represents ‘1’ when Chennai Super Kings have won a match and ‘0’ when it lost the toss
* Home Ground : In this case if Chennai Super King is playing in Chennai, then it is considered as Home Ground. Represent ‘1’ if it is a home ground and ‘0’ if it is not a home ground
* Chennai Super King Batting Average : This metrics calculate the average run scored by Chennai Super Kings in last 5 matches played in IPL against any team irrespective of the current match team
* Opposing team batting average: This is like the above metrics but done for the Opposing team which is playing against Chennai Super Kings
* Win\_Runs(WR) : If Chennai Super Kings have won, this metrics tells us about the win margin by runs
* Win\_Wicket (WW): If Chennai Super Kings have won, this metrics tells us about the win margin by wickets
* Average Win Margin(AWM) : It is a combination of two metrics : Win\_Runs, Win\_Wicket

AWM=WR+5\*WW

5 has been taken based on experience of Cricket which attribute 1 wicket to 5 runs for T20 cricket match and there is no analysis done on that. It is one of our model assumption.

Final Dataset

**Match Prediction Model**

We will be using one of a very simple and efficient model of machine learning called as Decision Tree, CART or Trees to predict the outcome of a cricket match for Chennai Super Kings.

**Overview of Decision Tree**

It is a decision support tool which is a tree like structure to build a set of rules that predict the outcome of our metrics that needs to be predicted. It classifies or predict an outcome based on a set of predictors. Set of rules defined can be implemented on both continuous and categorical variables. For example: Classify a “mail as spam or not spam”

**Rules:** If (number of words>90) and (email\_address= marketing) mark as Spam mail is a type of decision tree which predict a mail being a spam or not

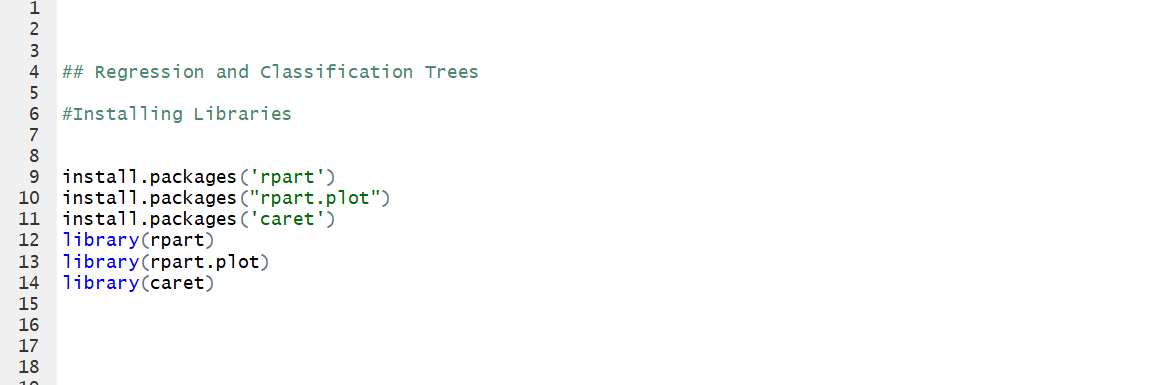
For absolute beginner who want to study more about decision tree, below are the link along with topics description:

* [Recursive Partitioning in Decision Tree](https://newonlinecourses.science.psu.edu/stat555/node/100/)
* [How to split: Continuous and Categorical variables in Decision Tree](https://www.coursera.org/lecture/ml-classification/threshold-splits-for-continuous-inputs-tn6M9)
* [Pruning the Decision Tree](https://www.displayr.com/machine-learning-pruning-decision-trees/)
* [Measuring Impurity : Gini Index and Entropy](https://medium.com/machine-learning-101/chapter-3-decision-trees-theory-e7398adac567)

**Model Building**

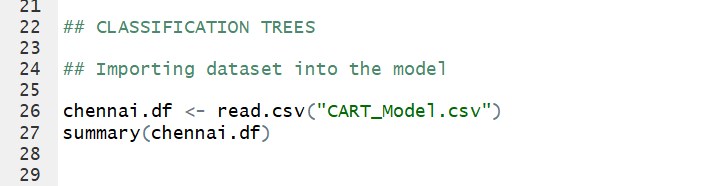
We will be using R Studio for building our model on Match prediction and we will be using the data shown in dataset preparation

* Installing libraries in R: We will be using rpart for building the decision tree classifier in R along with rpart.plot and caret.

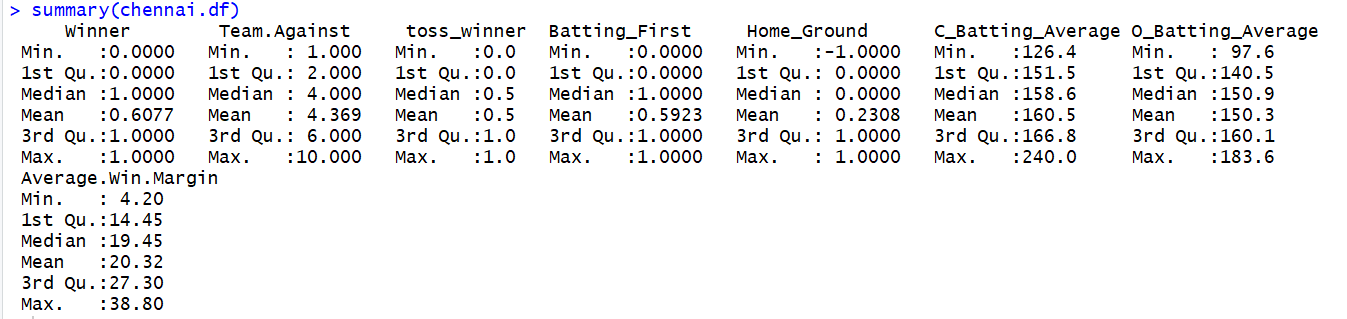
`

* Importing the data frame for the model: Below please find the R syntax for importing data in R and the summary staistics of the variables chosen for the model

Importing data in R



Summary Statistics



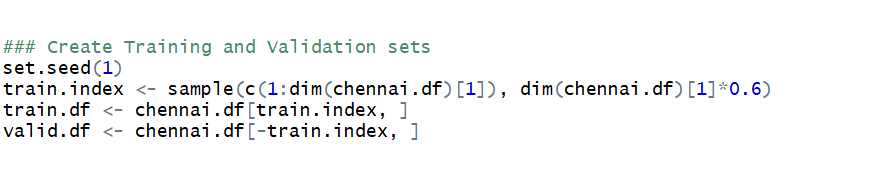
Team\_Against, Home\_Ground, Toss\_Winner metrics will not make any sense from the data since these are qualitative categorical variables converted to quantitative categorical variables.

**Training and Validation Data**

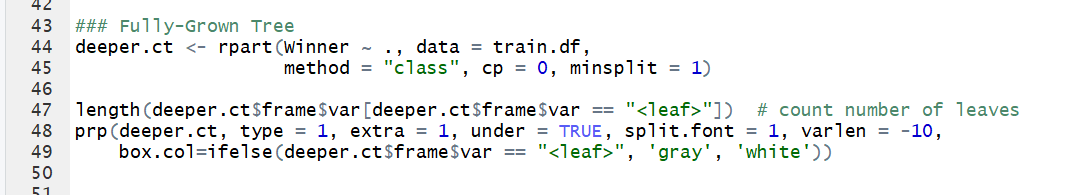
We will be dividing our data into Training and Categorical data.

* Training Data: We will be using this dataset to define the set of rules or build a decision tree from the predictor variables to predict the outcome of winning a match by Chennai Super Kings.
* Validation Data: It is used to test the model accuracy based on dataset which was not used for building the model

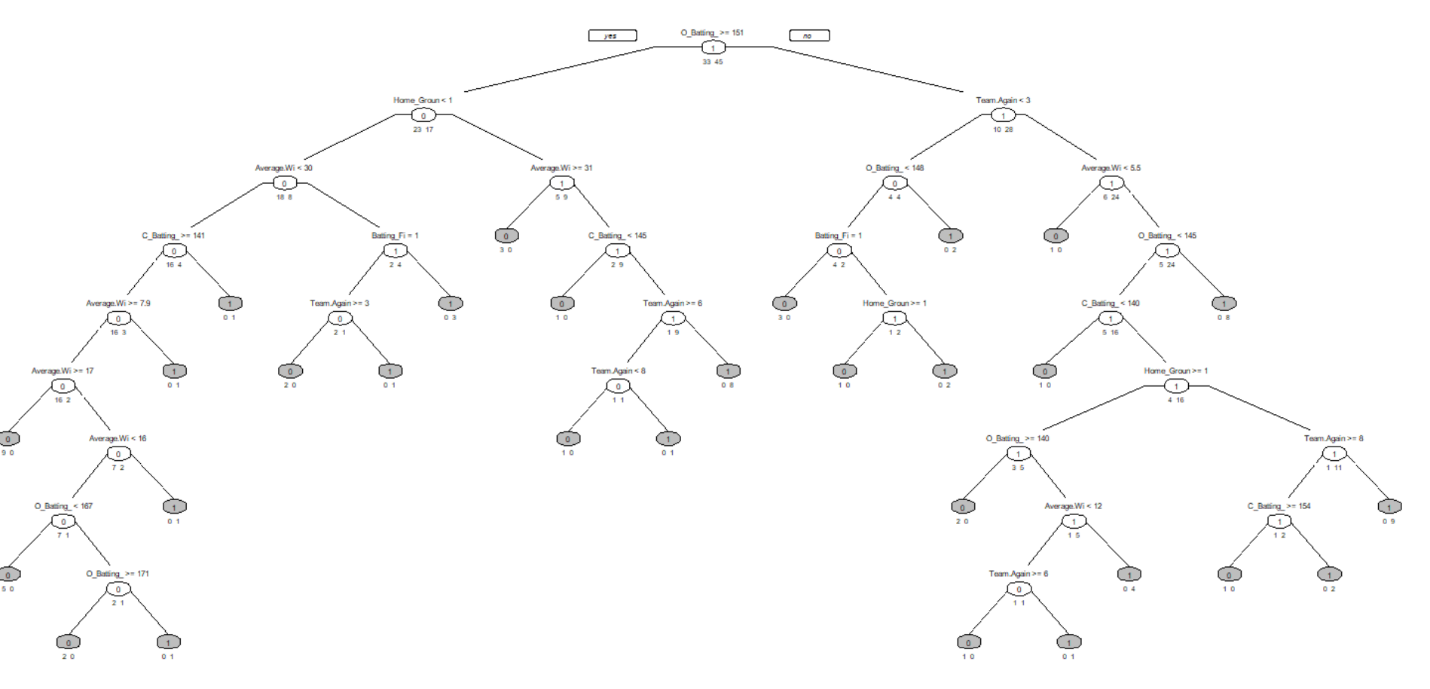
We have split the data into 60% training dataset and 40% validation dataset. But we will be discussing later about getting the best decision tree which will be based on multiple training and validation dataset.

Training and Validation dataset creation

**Generating Decision Tree**

We have generated a fully-grown tree in which every end node will contain a pure class. It will give us 100% accuracy with the training model.

**Fully Grown Tree**



A fully-grown tree will lead to a problem known as Over-fitting. The model will predict the outcome of training dataset by 100%. But we have built the model to predict the outcome of validation dataset not training dataset. An overfit model will not be able to achieve a high accuracy on validation dataset and hence not preferred.

**Pruning Decision tree**

CART/decision tree first lets you build a fully grown tree and then prune it back as the validation error rises.We will generate multiple trees after pruning and decide the best tree based on lowest Cost Complexity.

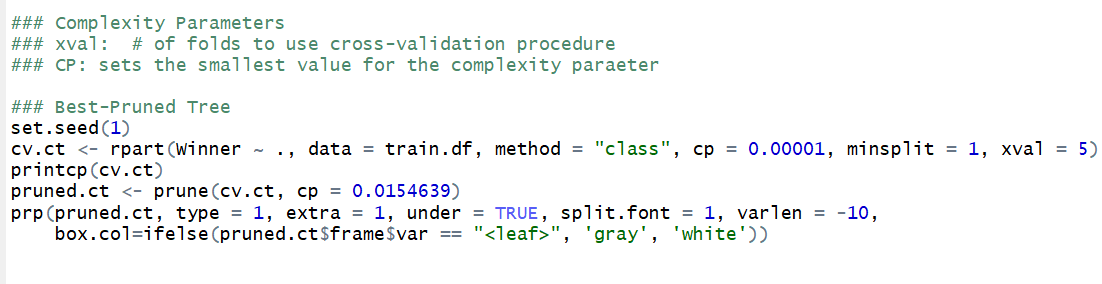
**Best Model creation**

We need to create the best model which will predict the match outcome of Chennai Super Kings. We have followed the following steps:

* Split the dataset into three-part training, validation and test data set
* Pruned the tree
* Reiterated the above two steps for creating multiple pruned decision tree based on multiple training and validation dataset
* Choose the best model based on Cost Complexity

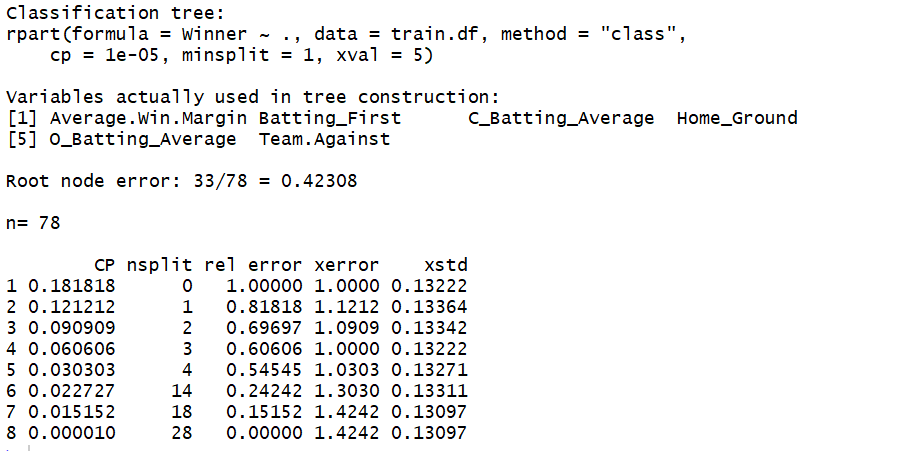
We have created 5 folds to use our cross-validation procedure of choosing the best model of predicting match outcome.

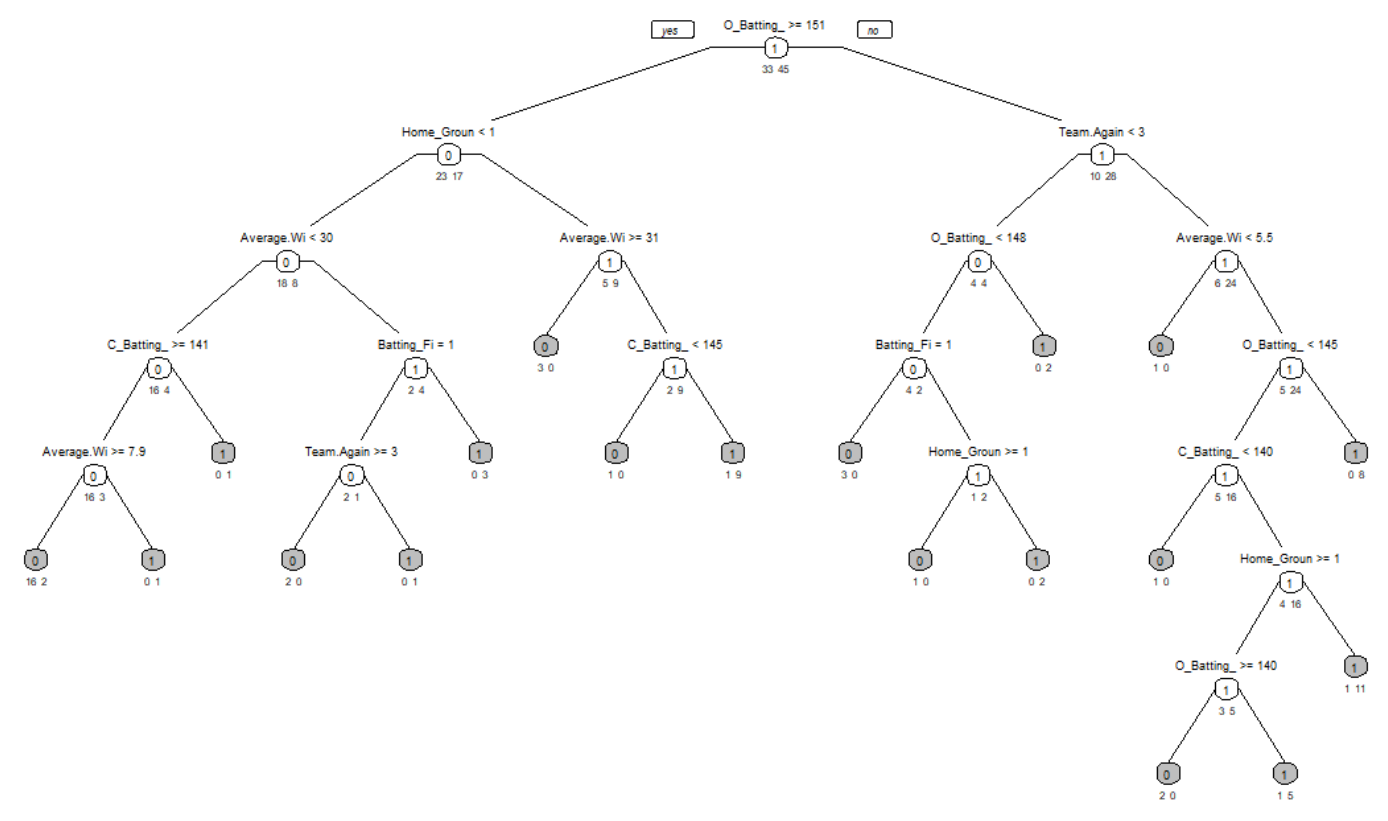
Creating Best Model



The below table illustrate the Cost parameter table along with the split in the decision tree. To select from the below model, we need to choose the model which gives us viable minimum error along with minimum number of splits.

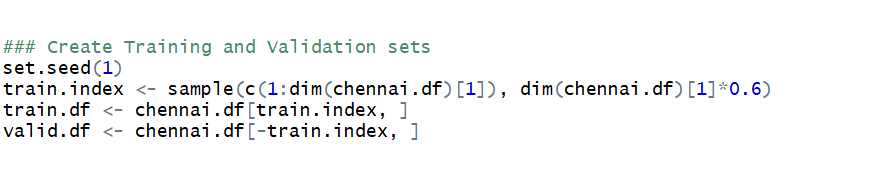
CP table to predict get the best Decision Tree

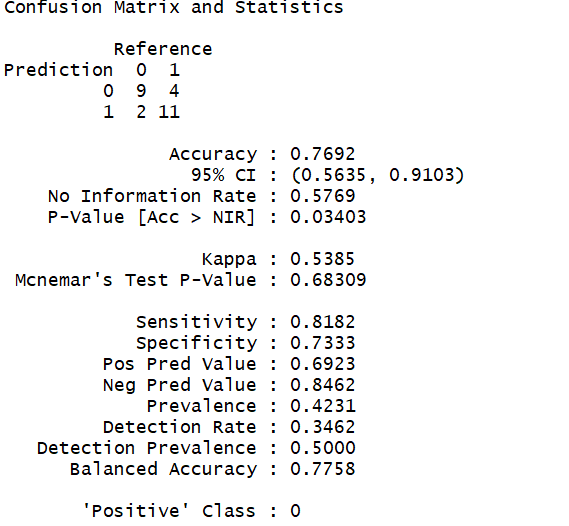


**Best Model Diagram**Below is the diagram generated by our model to define the set of rules that predict Match Outcome when Chennai Super Kings is playing in IPL.

**Model Accuracy**

We will be testing our model accuracy on a testing dataset which is untouched dataset, the dataset which has not been used in building our model i.e. training and validation dataset. We have used the caret function in R to create a confusion metric table depicting the accuracy of the model.



Based is the accuracy result of our model result after all the hassle we have gone through:

**Results**

We have achieved 77 % accuracy in our model, it means if we predict the match outcome when Chennai Super Kings is playing in IPL, the probability of correct prediction will be 0.77.

**Conclusion**

Finally, we have build the model after going through the complex process of decision tree. We segregated the data into training, validation and testing data. The final decision tree was made after successive folds in the training and validation dataset. We came to know that building a fully grown is not the solution to our problem of match prediction, since it led to a problem called as over-fitting. After recursive partitioning and successive iteration, we choose the model faced on Cost Factor metric.

The model build is having an average accuracy of 77%, in order to achieve higher accuracy we need to reiterate the above procedure by adding new metrics to the model.

I will be coming up with new articles about building models in the area of Sports Analytics and will love to help you in case you have any question in the above article.