Jason Larosiliere

Professor Nguyen

Applied Data Mining

18 December 2022

NBA Teams in the Playoff Picture

Abstract

The purpose of this paper is to use different models to find interesting results and patterns in NBA Teams and their stats. Through evaluation, I can get a better idea of what affects their chances of making the playoffs. What did these winning teams do that other teams didn’t? What did they prioritize most of the court and how does coaching play into this? During this project I will use what I’ve learned throughout the year to uncover the aspects of a winning team through predictive modeling.

Machine Learning algorithms were originally used to analyze data and find interesting patterns to gain knowledge. Most of the time there are underlying relationships in data that provide a better insight into the topic. The bottom line is through different algorithms and computer programs we can take data and learn many things that aren’t visible from the surface. This is where supervised and unsupervised learning comes in. These are the 2 basic approaches of machine learning. Supervised learning is when we want to predict outcomes from algorithms while unsupervised learning uses algorithms to discover patterns.

Supervised learning problems with data could either be classification or regression. Classification involves designing data to different categories in which they belong. Regressions are used to predict values based on several predictor variables. It exposes the relationship between dependent and independent variables. This is what makes supervised learning so vital in the real world and people do not even realize its effects. For example, our email apps automatically classify inbox mail from spam mail which relates to classification. Also, you can predict your exercise plan and how much weight you need to burn by entering in weight, height, age, exercise/eating habits and other various predictor variables, which is an example of regression.

The first part of building a predictive model is addressing any missing values or anything against normality with imputation and transformation. We then divide the data into partitions which is known as data partitioning. We validate our data by using 1 partition of the data for training and saving the other partition for validation and testing. One problem that may arise during predictive modeling is overfitting which is when a model corresponds exactly with the dataset or training data. As a result, the model would not be effective in fitting additional new data.

A Decision Tree is an example of supervised learning. This model allows you to decide about several questions and/or processes. It usually starts with a single node and branches off to possible outcomes to the first decision. The more leaves a tree has the more complex it is and the tree with the most is the maximal tree. One pro to decision trees is it requires less effort and is very versatile. One con is it is prone to overfitting. A Random Forest is simply a forest made up of multiple trees. In this case, a forest will gather a majority vote or average from all the decision trees. This is very useful since it deals with the potential inaccuracy of a decision tree by creating more. The more trees and variables the more accurate. A con to this method would be like the decision tree in terms of prone to overfitting.

A tree with one node and two leaves is a stump. Adaboost is simply a forest of stumps rather than trees. It learns from the mistakes of weak binary decisions and creates new stronger ones. Essentially, it attempts to reduce the bias error. By increasing the # of rounds we can adjust the accuracy of the model. The learning rate is an interpretation of how fast the model learns and the higher it is the more contribution from each classifier. One pro to this model is immunity from overfitting and one con is it requires high quality datasets to perform well. Another type of boosting model is Gradient Boosting. This forms a model from weak decision trees. The difference between this and a Random Forest is the decision trees in Gradient Boosting are added together to form the best. Essentially the same as Adaboost but accounts for loss and is more flexible, however one con to this model is its sensitive to outliers.

A Regression model predicts the response variable with a linear combination of predictor variables. There are 3 ways of selecting the variables which are forward, backwards and stepwise. A logistic regression model is the same, but the dependent variable could either be the values 0 or 1. For example, the dependent variable could be whether you pass a test or fail. Lasso makes this model simpler by shrinking the data values. One con to this is it requires the dependent variable to be normal. KNN algorithm uses proximity to make classifications about the grouping of datapoints/observations. It eliminates the likelihood that a new data point will become a part of another group. The number of nearest neighbors tells our model how far out radius-wise. This model is simple, easy and can solve classification and regression problems, however it does not do well with datasets with many variables.

Misclassification is when an individual is classified in the wrong category

(FALSE NEGATIVE COUNT + FALSE POSITIVE) / TOTAL

Sensitivity is the proportion of positive individuals correctly classified

TRUE POSITIVE / ACTUAL POSITIVE

Specificity is the proportion of negative individuals correctly classified

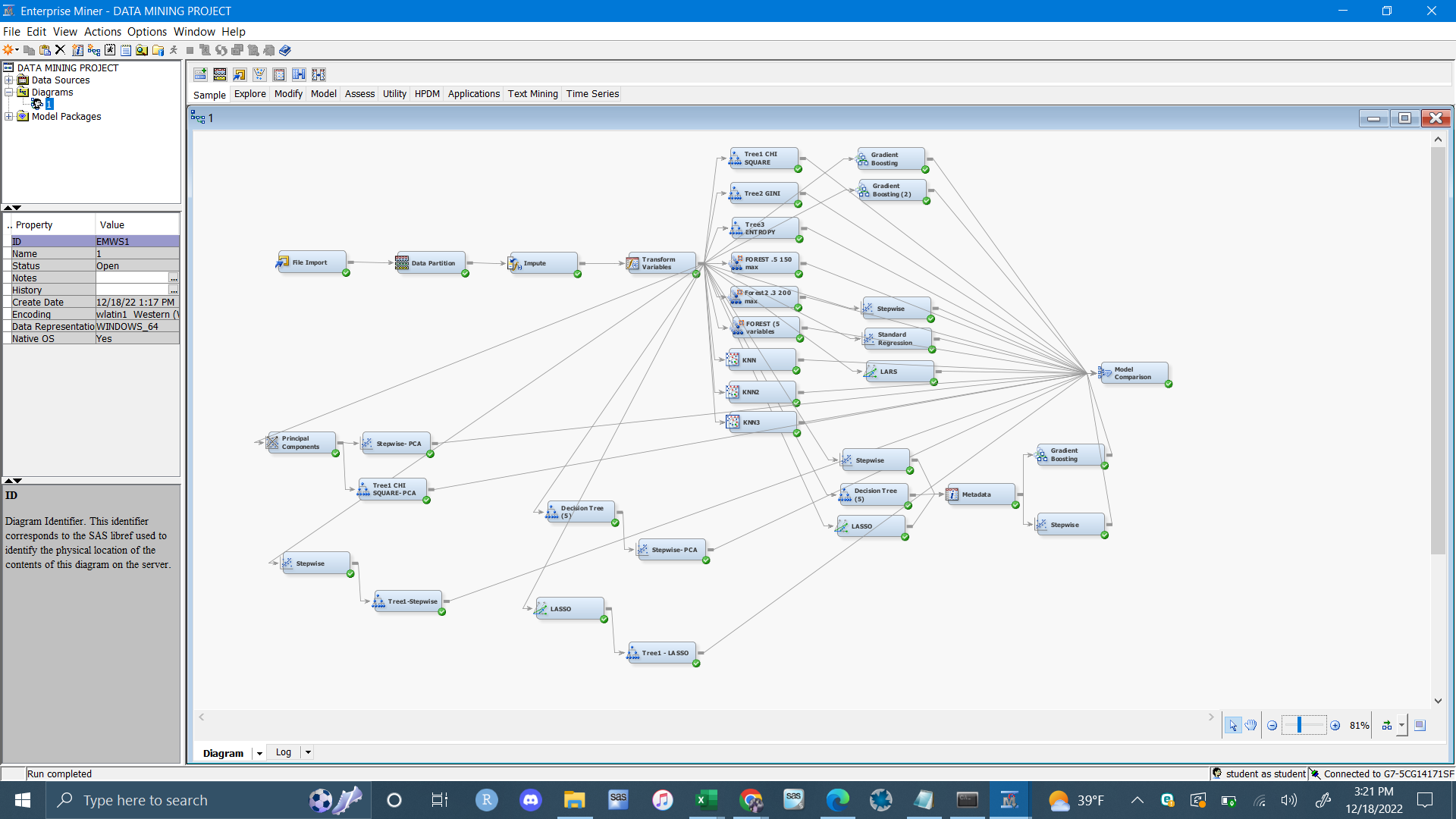
TRUE NEGATIVE / ACTUAL NEGATIVE

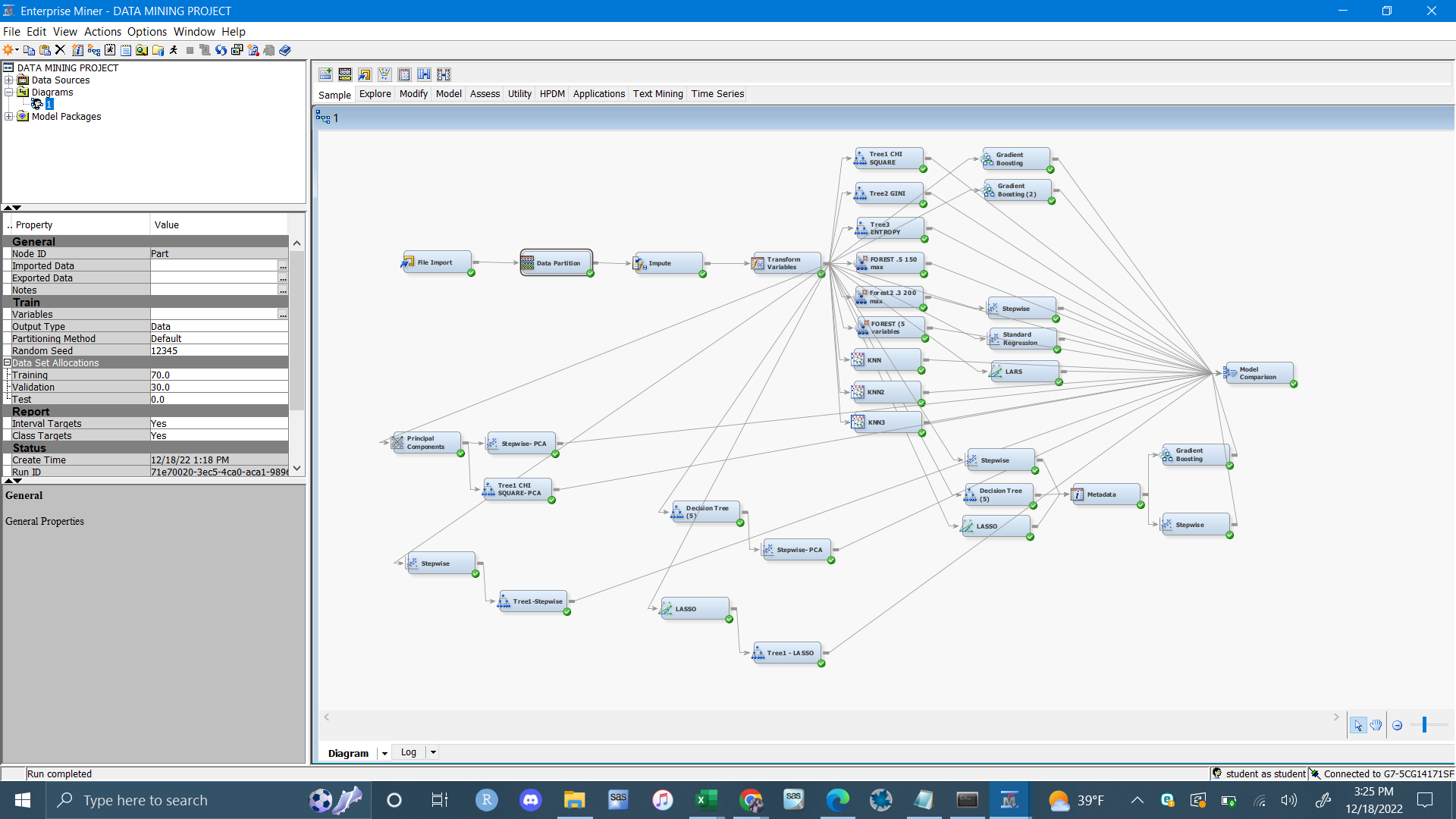
F1Score is used to measure the performance of classification and measures the model’s accuracy

(PRECISION \* SENSITIVITY) / (PRECISION + SENSITIVITY)

ROC is a graph that displays the performance of the classification model. ROC Index is the area under the ROC Curve. Cumulative lift compares the models of different lifts to tell you which is the best. It shows the improvement of each model. The % response is the proportion of the total observations captured.

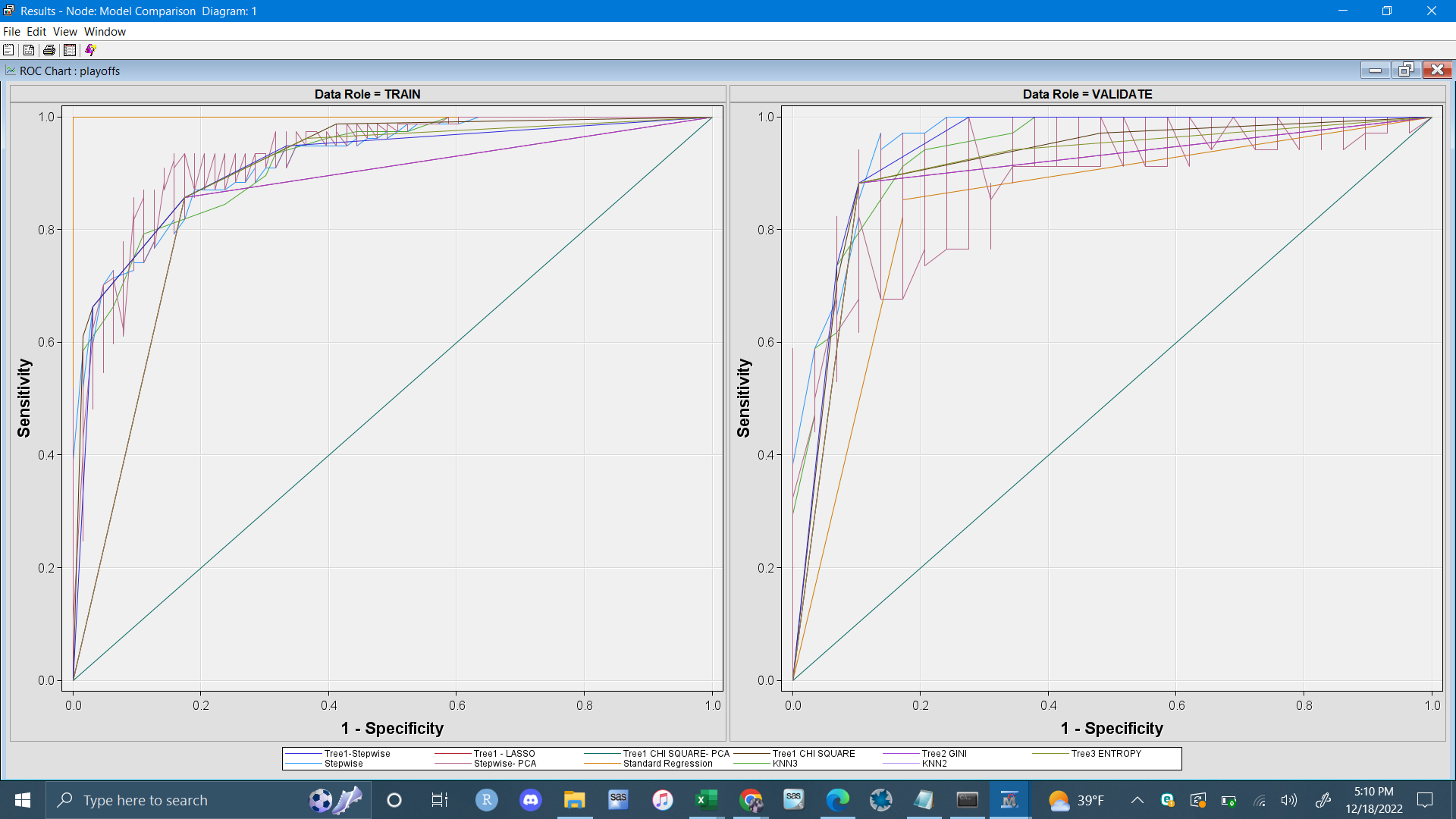
The dataset that I’m using includes 203 observations and 11 variables. The topic of my study is the stats of 200 NBA teams from the year 2011-2017. I combined data from 2 datasets found on Data.World and the independent variables I have are points, rebounds, assists, steals, blocks, turnovers 2pt %, 3pt %, efficiency, def. efficiency, and I have whether they made the playoffs or not as the target binary variable. By doing this, we can get a better understanding of what goes into a winning team and what did the playoff contenders do on the court that teams who didn’t make the playoff picture, didn’t do.

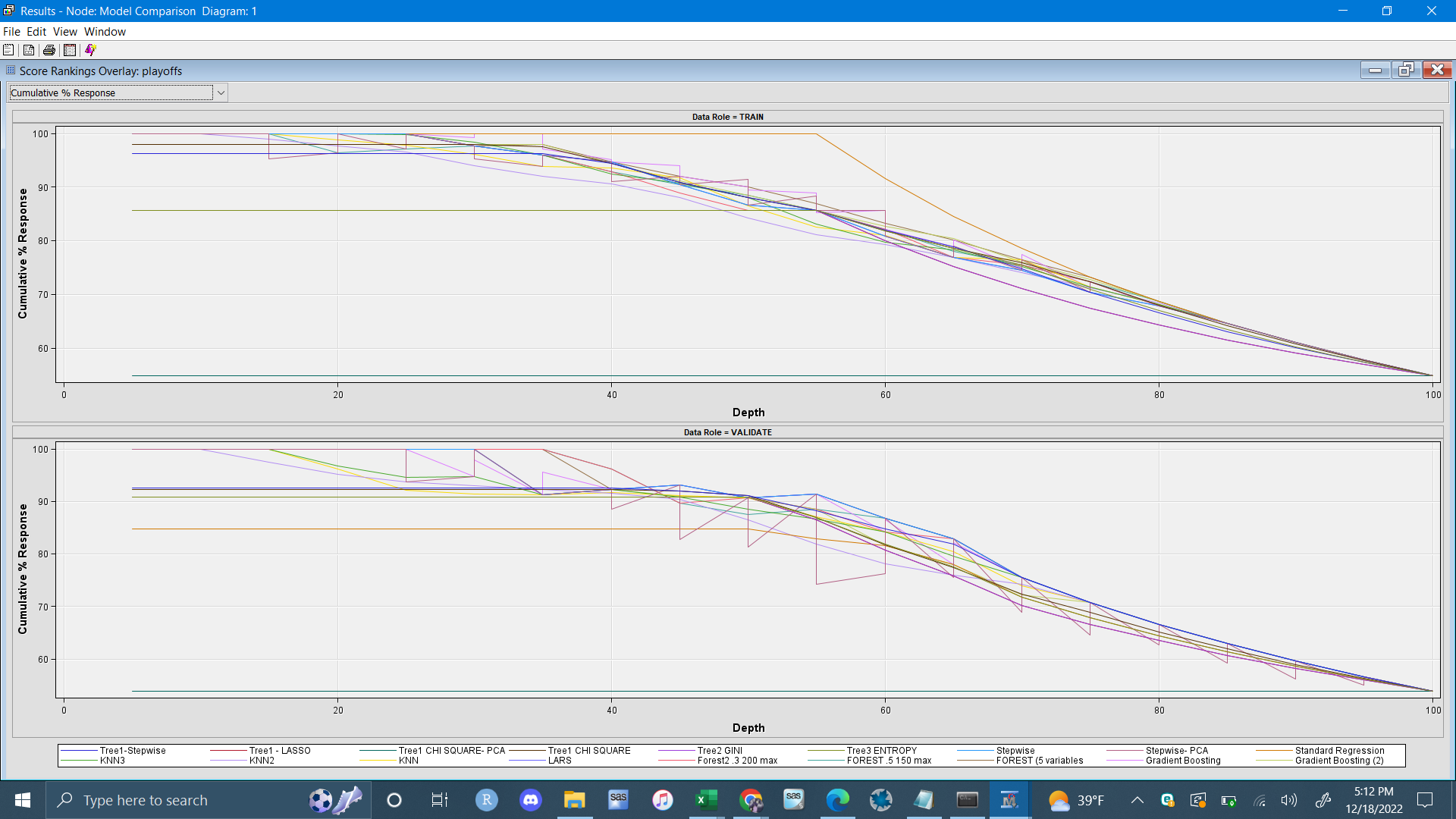




For Decision Tree, I created 3 different models with different values/hyperameters. The first Tree is a chi square. Second tree is Gini and the third is entropy. I also created 3 more decion trees with techniques to improve the model. These three decision trees use data that was ran through stepwise regression, LASSO regression and principal components analysis first. In terms of misclassification rate and ROC index, the decision tree that in my opinion, that performed the best is the chi square criterion tree that was ran through a stepwise regression first. The misclassification rate is .111 and the ROC index is .912. For this tree I used 20 leaves and a leaf size of 5. and the variables that came up as the most important is the transformed variable, defensive efficiency. For Random Forest, I created 3 different models with different values/hyperparameters. In terms of misclassification rate, the random forest that performs the best had 100 max trees and 0.6 proportion of observations. The number of variables that were considered at each split was 5. The three most important variables were Standard deviation of defensive efficiency, Sd of efficiency and Sd of Turnovers.

The number of rounds and learning rate I used for the best Gradient Boosting model was 50 rounds with a .1 learning rate. This model was made using the technique of Metadata. I ran the model after combining the variable selection process of stepwise regression and decision tree and LASSO regression to achieve more accuracy. The only variable that came up as important is the standard deviation of defensive efficiency. The misclassification rate is 0.079 and the ROC index is .953. For KNN, 3 different models of hyperameters were created. The model that performed the best had 16 neighbors and a classification rate of .127. In terms of misclassification rate the best model overall is the stepwise regression. To improve the model, I tried principal components analysis of the data before a stepwise regression again. I also ran another regression model but used a decision tree to select the variables. However, the first stepwise with PCA before it gave the best results. The misclassification rate is 0.157. The MSE of the regression is 0.079 and the MAE is 0.85.





In conclusion, I have implemented the above models to further explain the NBA teams and their success in making the playoffs. Through different hyperparameters and techniques in selecting variables, I worked to achieve the best possible model. What I got out of this is that defensive efficiency, regular efficiency and turnovers are the key indicators in making the playoffs and that teams should prioritize protecting the ball and making it harder for the opposing team to get open shots. This project could definitely be improved by including more teams, maybe bringing in teams from 2018 to right now. This can help get a better understanding of the game of basketball today since it is a sport that changes a great deal over time. Another thing is including more variables that differ from the stats that come up on the box score. For example, the attendance of the home team and how that plays a part and how many all-stars per team as a dummy variable. Lastly, would be to include any variable regarding the coach of the team for that year.

References

*NBA Team stats - dataset by etocco* (2022) *data.world*. Available at: <https://data.world/etocco/nba-team-stats> (Accessed: December 18, 2022).