Project 3 Report

Mahad Farah

January 5, 2023

### Project#3 Final Report

### NHL Salary Prediction

### BA 4420 Data Analytics for Managers

### by

### Mahad Farah

### December 9, 2022

# Executive Summary

How do professional sports teams determine how to distribute their money for the right players? This is the million-dollar question when it comes to dealing with finances in professional sports, and in this case, the NHL. Business analytics are used everyday in the sports industry, so this is only one example of how they can be used to make predictions that influence business decisions. Many people are very engaged in all sorts of sports betting, which is based on statistics to predict the outcome. These predictive models can also be used within professional sport departments. For example, successful NHL teams know how to spend their money wisely in order to get players that will create wins for the team. Salary predictions can be measured not only by how well certain players perform, but also by how they contribute to team success. The more successful the team is, the more money is made, which is the most important goal.

Correctly paying your athletes is very important when it comes to being a successful team. My prediction models will not only help you evaluate players salaries to see if their salaries are reasonable, but also help decide what an accurate salary should be. Despite some outliers or position differentiation, the more time on ice by a player equals a higher salary. Through exploratory data analysis I discovered that the specific measures that have the greatest influence on player salary are draft year, expected team goals scored while player is on the ice, and faceoff losses. I also found that time plays a significant role in player salaries. Rookie contracts are usually lower, and then fluctuate based on performance. Older players tend to keep a higher mean salary than rookies, but have less outliers when it comes to salary received. All of these factors must be taken into consideration when determining player salary, and the prediction models will make this even easier.

I have put in a lot of work and thought into finding and creating successful models that may be able to predict an NHL player’s earned salary, and in turn help financial teams in the NHL fill their salary cap efficiently in order to make a successful team. I have created three models, these being a linear regression, regression tree, and random forest model. Although the regression tree has the highest prediction accuracy, I believe that all three models used together would create the best understanding among an NHL financial department. The visual aspect of the regression tree, along with the importance plot of the random forest model will help everyone understand just how much each statistic factors into each prediction. I truly believe that the model will be successful in helping any NHL team determine a reasonable salary base. I have created a linear regression model, regression tree model, and a random forest model to base the predictions.

# Table of Contents

[***Business Understanding 4***](#Xcd0b4b56f2d0fdb40a1a1f64fa02cc32277e3b9)

[***Data Understanding and Exploratory Data Analysis (EDA) and/or Data Preparation 5***](#Xe9f8b27d51f2dc3304a948807b8535c8df1f4cd)

[**Data acquisition/collection 5**](#X7f5e8f33c5dbee31886616c24d6f0ec5183fc82)

[**Data description and quality 5**](#Xf44962d0e60cbfefd324ae7dd8e3618e69f8d59)

[**EDA 8**](#Xbb10e33a3a02a5078bf17f686e86e59ec4fcf61)

[**Dimension reduction 12**](#Xe288d55e2190a8f0fef4be0a95ee92556bfa8bd)

[**Preliminary insights 18**](#X5a99c0901bcdfad427db295dfb2093eee9fe962)

[**Data cleaning 19**](#X2266138a37358d5d8a664a038576505a91a9d6c)

[**Data Preparation 20**](#Xb212552f1882b8ca85f4e2dc8f3bdb72c1213fc)

[***Expanded Final Modeling and Predictive Analytics and their Evaluation 22***](#X1fe7e6fbb831cfb32a1cec972c6ba776f0ee3bd)

[***Deployment 33***](#Xab810916925cb77f3ba663bb940c7842a26c4bd)

[***Comparisons with documented results 34***](#Xe10d6d3027c3635ff2d68adb33f83a1991914ca)

# Business Understanding

### Application domain

I was very interested in the analytics behind determining the salaries of professional athletes and thought that professional hockey would be an interesting and relatable topic to explore being I are in Duluth, Minnesota. The business analytics project is taking place within the NHL and more specifically the financial department within each team throughout the league. This data set would also be applicable to people predicting results for sports betting; however the project will use the salary target variable to relate it to sports finance. I will use predictive analytics to determine salaries that NHL players should be earning based on a variety of factors. Predicting player salaries will allow financial departments to use salary budgets more wisely, and get the most valuable players at the right price.

### The project’s primary objective as a question:

How do NHL financial departments determine sufficient and efficient salaries of the players on roster based on informative performance measures?

### Criteria for Success

Business success for this project is detected by how financial departments spend their money wisely, while saving time and money at the same time. Based on league averages, the goal is to create contracts that are high in value. Meaning that they are correct in predicting player salaries based on their performance, so they are under or over paid. If a team overpays an athlete, and that athlete isn’t contributing to the team, it is considered a waste of money. If a team underpays an athlete, however, they may want to leave the organization, so the team would be losing important talent. Business success in sports is simply defined as win and losses. The more your team wins the more money you bring in and the more “success” you can claim. Regarding the goals of this modeling project, it will be deemed successful if an NHL team can increase winning percentage by 15% over a two year period of utilizing the salary prediction models.

### Problem Type

The specific problem type of the project is making predictions. This prediction dataset is supervised because I have a defined target variable, which is the player’s salary. I will use predictor variables like plus/minus, minutes, goals, shots, games played, etc. to create successful prediction models for the target variable salary. I chose this problem type because being able to predict these aspects within sports is a newer idea, and is extremely effective for the team managers whose jobs rely on their decision making. Creating models that can aid them through analytics, will only make their jobs easier and help them make better business decisions.

# Data Understanding and Exploratory Data Analysis (EDA) and/or Data Preparation

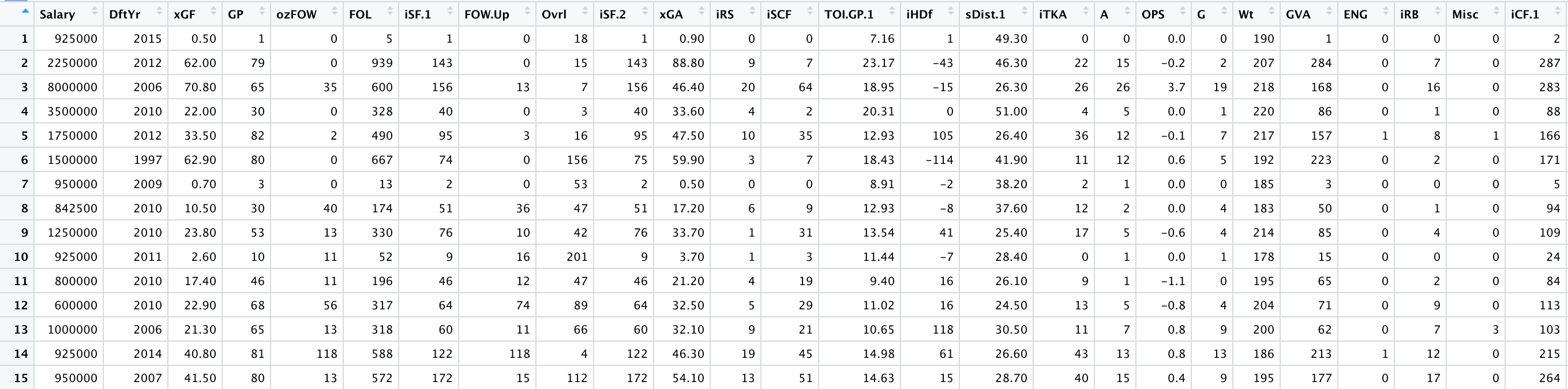
## Data acquisition/collection

I found the dataset on kaggle.com. It is acquired of 874 observations, which in this case mean individual NHL hockey players, and 154 predictor columns. This data is from games played from 2017 and earlier, which provides us with reasonable values that aren’t altered by the COVID-19 pandemic. I understand that this is a massive amount of variables for real NHL financial departments to be able to handle and utilize, so I will use dimension reduction to narrow these variables to the most important. In order to perform necessary dimension reduction and data preparation, I decided to combine the given train and test dataset into the full data set. They also provided the test salary separate from the rest of the test variables, so I combined that dataset as well. I had to complete this process, becuase there was a lot of cleaning that needed to be done. I renamed this combined csv file “TotalHockeySalaryData.csv”.

train <- read.csv("train.csv")  
data2 <- read.csv("test.csv")  
tstsal <- read.csv("test\_salaries.csv")  
test <- merge(tstsal, data2, by = 'row.names')  
test <- test[,-1]  
totaldata <- rbind(train, test)  
write.csv(totaldata, file = "TotalHockeySalaryData.csv", row.names = FALSE)

Some of these variables aren’t relevant to the prediction process, which I will acknowledge and remove through dimension reduction. I must include all of these variables when performing dimension reduction techniques like correlation analysis and forward or backward reduction, because I need to preserve the most important variables. Although some of the variables may be confusing to the normal person, NHL finanial departments, who will be working with the data hands on, will understand each and every statistical category.

## Data description and quality



Screenshot of raw data.

The data instance is each specific row of data, which is represented by an NHL player in this dataset. The target variable, in the specific project, is the player’s salary (first column of the dataset). Each variable is described in the following table. Given that there are so many variables in the original dataset, I described just the ones that I based the three models from.

| Variable | Description | Notes |
| --- | --- | --- |
| Salary | Salary of each player in the dataset | Min. of $575,000 & Max. of $14,000,000. No missing values. The target variable. |
| xGF | The team’s expected goals (weighted shots) while this player was on the ice, which is shot attempts weighted by location | Min. of 0.1 & Max. of 115.70. No missing values. |
| DftYr | Year drafted | Ranges from 1990 to 2016. 125 missing values. Some players aren’t drafted. |
| GP | Games Played in the season | Ranges from 1 to 82. (there are 82 games in an NHL season). No missing values. |
| ozFOW | Faceoffs won in the offensive zone | Ranges from 0 to 420. No missing values. |
| FOL | The team’s faceoff losses while this player was on the ice | Ranges from 1 to 1196. No missing values. |
| isF.1 | Shots on goal taken by this individual | Ranges from 0 to 313 shots on goal. 11 missing values. |
| FOL.UP | Faceoffs lost when the team was in the lead | Ranges from 0 to 385. |
| Ovrl | Where the player was drafted overall | Ranges from 1 to 287. Missing values. |
| iSF.2 | Shots on goal taken by this individual | Ranges from 0 to 313. |
| xGA | Expected goals allowed (weighted shots) while this player was on the ice, which is shot attempts weighted by location | Ranges from 0 to 108. No missing values |
| iRS | Shots off the rush taken by this individual | Ranges from 0 to 35. |
| iSCF | All scoring chances taken by this individual | Ranges from 0 to 139. |
| TOI.GP.1 | Time on ice divided by games played | Average time on the ice per game appeared in. Ranges from 3.95 minutes to 27.25 minutes. |
| iHDf | The difference in hits thrown by this individual minus those taken | Good measure of aggressiveness. Ranges from -111 to 227. Large range of values. |
| sDist.1 | The average shot distance of shots taken by this player | Shows shot tendencies. Ranges from 0 to 68.10. May be skewed based on number of shots taken by each player. |
| iTKA | Takeaways by this individual | Ranges from 0 to 96. Shows aggressiveness on defense. |
| A | First assists, primary assists | Ranges from 0 to 70. Sum of A1 and A2. No missing values. |
| OPS | Offensive point shares, a catch-all stats that measures a player’s offensive contributions in points in the standings | Ranges from -1.70 to 10.50. Measures offensive effectiveness. |
| G | Goals | Total goals scored by player. Ranges from 0 to 44. |
| Wt | Weight | Measured in pounds. Ranges from 157 to 265. No missing values. |
| GVA | The team’s giveaways while this player was on the ice | This is a negative statistic, however those that don’t play cannot contribute. May skew data. Values range from 0 to 153. No missing values. |
| ENG | Empty-net goals | Ranges from 0 to 4. Difficult to achieve, shows skill. |
| iRB | Rebound shots taken by this individual | Ranges from 0 to 41. No missing values. |
| Misc | Misconduct penalties | Ranges from 0 to 4. Very rare. |
| iCF.1 | Shot attempts (Corsi, SAT) taken by this individual | Ranges from 0 to 609. 11 missing values. |
| Wide | Shots that went wide of the net | Ranges from 0 to 135. May skew data, would be better as a percentage. |
| GS.G | The player’s average game score | Measure of overall performance. Ranges from -0.81 to 1.28 |
| CA | Shot attempts allowed (Corsi, SAT) while this player was on the ice | Measure of defensive effectiveness. Ranges from 1 to 2273. |
| FA | Unblocked shot attempts allowed (Fenwick, USAT) while this player was on the ice | Measure of defensive effectiveness. Ranges from 1 to 1765. |
| HF | The team’s hits thrown while this player was on the ice | Ranges from 0 to 926. |
| G.Wrap | Goals scored with a wraparound | Skillset value. Ranges from 0 to 2. |

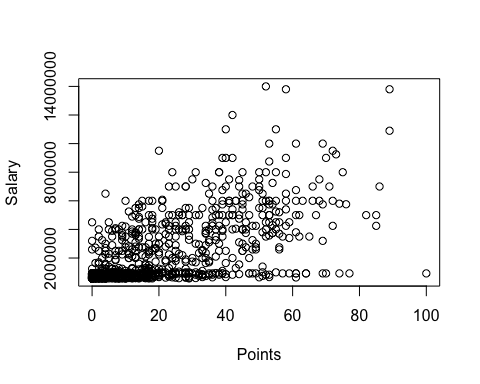
The data is relevant to this project and being used within real NHL financial departments because it includes all real NHL statistics from 2017 and earlier. Most recent data in many different genres has been altered by the COVID-19 pandemic, which can make it distorted. Since the dataset is both recent, but unaltered by the pandemic, it is trustworthy. The dataset is also easy to work with, as once more data becomes available in the future, it will be easy to add into the models and use to make even better predictions as time goes on and trends change.

This dataset will never be able to perfectly predict sports decisions as hockey like any sport has a certain amount of randomness and any player could bounce back after an off year or could be a one year wonder. With sport part of the fun is the randomness of it, without that randomness almost no one would watch the games. Will also have an issue with evaluating younger players as they will be bound to rookie deals and lower value younger player deals for their first few years in the league. If there wasn’t this level of uncertainty within hockey, people wouldn’t watch or bet on it. However, this dataset includes everything I need to know to make good predictions about player salaries.

## EDA

### First chart

options(scipen = 999)  
plot(totaldata$PTS, totaldata$Salary, xlab = "Points", ylab = "Salary")



This visualization of Points vs. Salary shows us that there is a linear relationship between the two variables. Relatively, the more points a player scores, the higher the salary of the player. This visualization is important because I will definitely dig deeper into the relationship between these variables, as well as others with the same relationship.

### Second chart

totaldata$DftYr[is.na(totaldata$DftYr)] <- 0

totaldata$DftYr[totaldata$DftYr > 2010] <- "Young"

totaldata$DftYr[totaldata$DftYr == 0] <- "Undrafted"

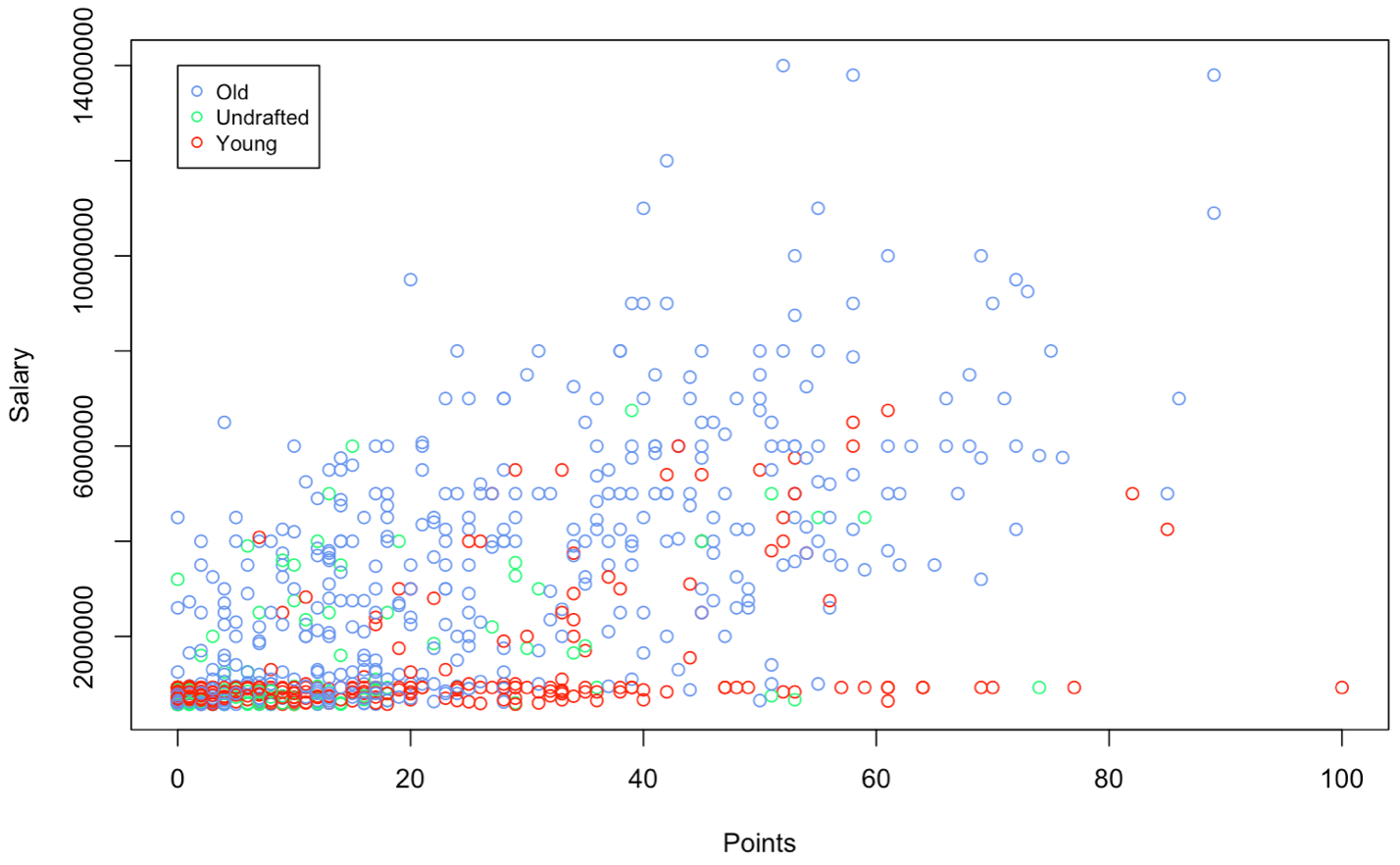
totaldata$DftYr[totaldata$DftYr <= 2010] <- "Old"

totaldata$DftYr <- as.factor(totaldata$DftYr)

plot(totaldata$PTS, totaldata$Salary, col=c('cornflowerblue','springgreen', 'red')[totaldata$DftYr], xlab = "Points",

ylab = "Salary")

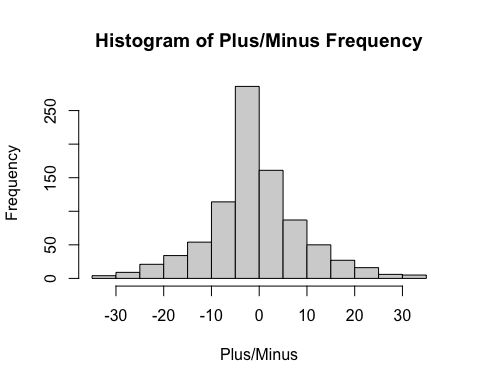
legend(x=0,y=14000000,c("Old","Undrafted","Young"),cex=.8,col=c('cornflowerblue','springgreen', 'red'),pch = c(1,1,1))



The scatter plot above shows the deeper analysis I did after the original two variable scatter plots. The red dots are players drafted before 2010, blue is drafted after 2010, and green are undrafted players. As you can see, The draft year is a great correlation to salary, because many of the higher paid players have been in the league longer. This shows two very impactful variables to the target variable, salary, all in one graph. As I dig even deeper into the dataset, I can make many other data visualizations similar to this one to show visually how much these statistics can be used to predict salary.

### Third chart

hist(totaldata$X..., xlab = "Plus/Minus", main = "Histogram of Plus/Minus Frequency")

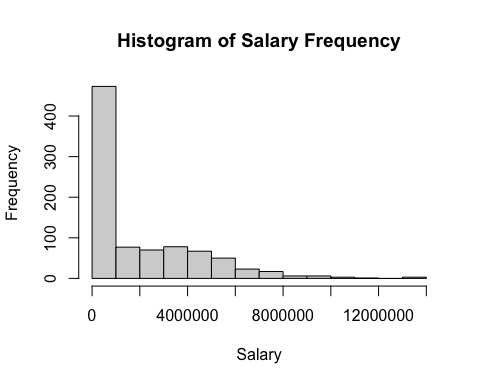


This Histogram visualization of player Plus/Minus shows us the distribution of variables. This shows us the frequency distribution of Plus/Minus among players in the NHL. There is a nice distribution, which is important because it should correlate nicely with the other variables. As you can see, there are a lot of players that have a pretty ineffective Plus/Minus between -5 and 5. This visual goes hand in hand with the previous scatter plots and next histogram, with the higher concentration of low salary and small performing players.

### 

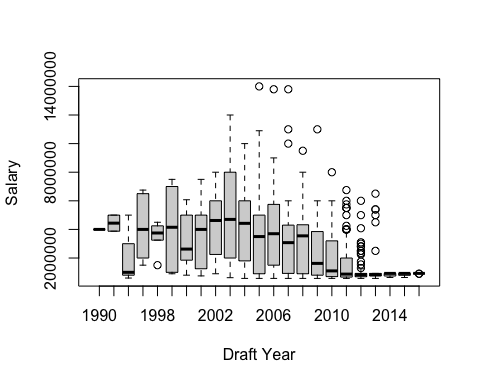
### Fourth chart

hist(totaldata$Salary, xlab = "Salary", main = "Histogram of Salary Frequency")



I decided to add this Histogram visualization of player salary that shows us the overall salary distribution. There are about five times the number of players with lower salaries (<$1,000,000) compared to any other salary bucket. This directly shows the larger amount of low salary players, and how these small salaries may be just as important, if not more important to NHL financial departments.

boxplot(totaldata$Salary ~ totaldata$DftYr, xlab = "Draft Year", ylab = "Salary")



The above boxplot of salary in each draft year shows us the salary distribution between each individual draft year. This tells us that the older players have less outliers and a higher mean salary than the younger players, who are in their rookie contracts. There are more outliers shown in the more recent draft years, probably showing the salaries of breakout young players. This graph shows a great story in order to understand the dataset.

## Dimension reduction

When I first started dimension reduction, I thought to run a simple correlation analysis and PCA, as well as remove some variables just because they weren’t common, or they seemed to overlap one another, like Assists1 and Assists2, for example. I first removed 127 variables, leaving us with just 27 variables. I performed correlation analysis and PCA on this data, which is shown below:

wrongdata <- totaldata[,c(1,7,8,9,15:23,25,26,27,41,44,55,57,59,60,111,115,116,124,138)]  
wrongdata <- wrongdata[,-c(4,5,6)]  
cor(wrongdata)

pcs.cor <- prcomp(na.omit(wrongdata), scale. = T)  
summary(pcs.cor)

## Importance of components:  
## PC1 PC2 PC3 PC4 PC5 PC6 PC7  
## Standard deviation 3.8046 1.7495 1.20530 1.02397 0.97916 0.76699 0.69782  
## Proportion of Variance 0.6031 0.1275 0.06053 0.04369 0.03995 0.02451 0.02029  
## Cumulative Proportion 0.6031 0.7307 0.79119 0.83488 0.87483 0.89934 0.91963  
## PC8 PC9 PC10 PC11 PC12 PC13 PC14  
## Standard deviation 0.6699 0.5478 0.52451 0.4700 0.4327 0.41231 0.37369  
## Proportion of Variance 0.0187 0.0125 0.01146 0.0092 0.0078 0.00708 0.00582  
## Cumulative Proportion 0.9383 0.9508 0.96229 0.9715 0.9793 0.98638 0.99220  
## PC15 PC16 PC17 PC18 PC19 PC20 PC21  
## Standard deviation 0.26066 0.22204 0.16749 0.12804 0.10452 0.07654 0.07171  
## Proportion of Variance 0.00283 0.00205 0.00117 0.00068 0.00046 0.00024 0.00021  
## Cumulative Proportion 0.99503 0.99709 0.99825 0.99894 0.99939 0.99964 0.99985  
## PC22 PC23 PC24  
## Standard deviation 0.05954 0.004439 0.001848  
## Proportion of Variance 0.00015 0.000000 0.000000  
## Cumulative Proportion 1.00000 1.000000 1.000000

pcs.cor$rot[,1:4]

## PC1 PC2 PC3 PC4  
## Salary 0.17578904 -0.086737644 0.16523518 -0.009347034  
## Ht 0.02523462 0.275013365 0.64774411 0.021953210  
## Wt 0.04288367 0.316824645 0.59862019 -0.025557742  
## GP 0.24061386 0.062876052 -0.08933063 0.150146376  
## G 0.20314777 -0.152478926 -0.03380414 -0.243646556  
## A 0.23403869 -0.181858083 0.06069908 -0.157477130  
## A1 0.22057966 -0.189230006 0.04363114 -0.201163859  
## A2 0.22422870 -0.151115187 0.07624568 -0.082971265  
## PTS 0.23596646 -0.180926883 0.02433275 -0.204685098  
## X... 0.04608227 -0.080642742 0.11567433 -0.687157185  
## PIM 0.15952829 0.394036325 -0.20167923 -0.193663438  
## Shifts 0.24949398 0.019420731 -0.01138224 0.195594677  
## TOI 0.25134670 -0.001902758 0.01278028 0.185813681  
## iCF 0.24880411 -0.091210056 0.01467592 0.033489958  
## iSF 0.24410694 -0.105466551 -0.00930152 -0.016889253  
## iHF 0.14794576 0.346952953 -0.12704310 0.069645108  
## iHA 0.20599323 0.101330243 -0.10779499 0.201644864  
## iMiss 0.24278262 -0.078630264 0.01661965 0.039551147  
## iGVA 0.22289561 -0.035396588 0.08187287 0.140162120  
## iPENT 0.19717617 0.316582562 -0.14191630 -0.087798226  
## Min 0.20677361 0.251385644 -0.10148296 -0.028224354  
## Maj 0.03779968 0.422139399 -0.22954367 -0.321059923  
## SF 0.25610559 -0.056035567 0.02476570 0.099511446  
## FOW 0.24896504 -0.019509105 0.02304221 0.175312432

wrongdata <- wrongdata[-c(2, 9, 15, 22, 23, 24, 26)]

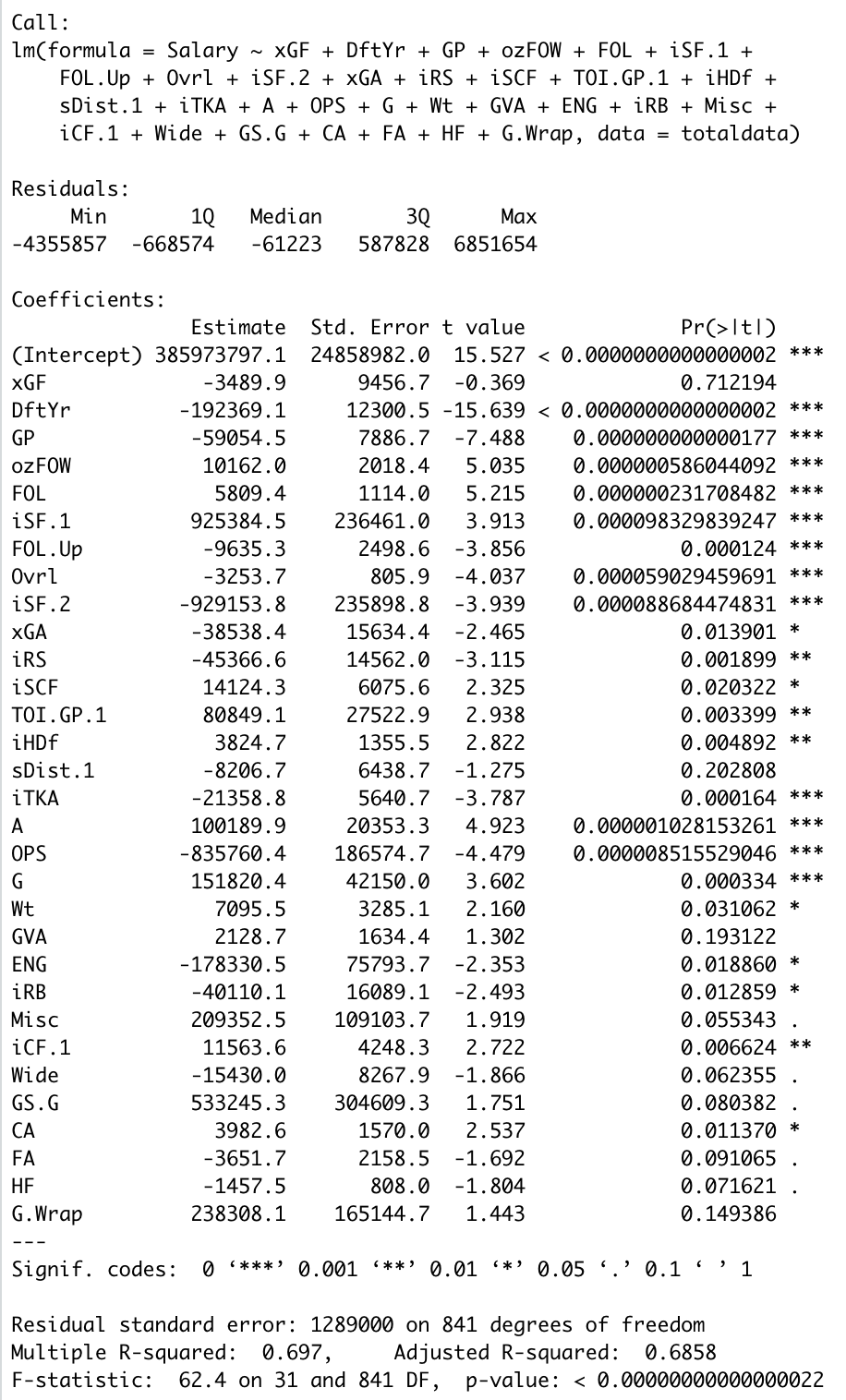
Through these methods of correlation analysis and PCA I further removed Ht, A, iGVA, iPENT, SF from the dataset because they were highly correlated with multiple other varibles in the set. After performing the initial regression and regression tree models with this dataset, I realized that I should go about dimension reduction differently. As shown in the beginning of the Final Model and Predictive Analysics section, you can see the poor models that this dataset created.

On the second round of dimension reduction, I started over with the whole dataset and did forward, backward, and stepwise selection techniques. I compared these results to one another and then compared them to the PCA analysis to see how they align. My goal was to create a set of variables that would predict NHL salaries accurately.

set.seed(1)  
train.rows <- sample(rownames(totaldata), dim(totaldata)[1]\*0.6)  
valid.rows <- setdiff(rownames(totaldata), train.rows)  
train.data <- totaldata[train.rows, ]  
valid.data <- totaldata[valid.rows, ]  
  
hockey.lm3 <- lm(Salary ~., data = totaldata)  
  
hock.lm.null <- lm(Salary~1, data = totaldata)

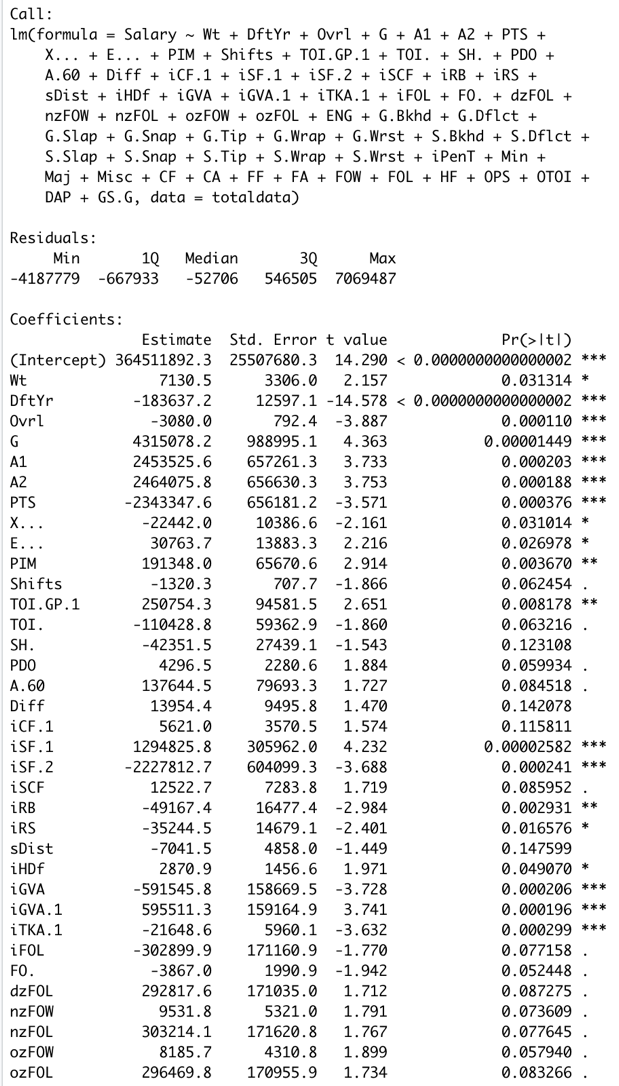
hock.lm.step <- step(hock.lm.null, scope=list(lower=hock.lm.null, upper=hockey.lm3), direction = "forward")

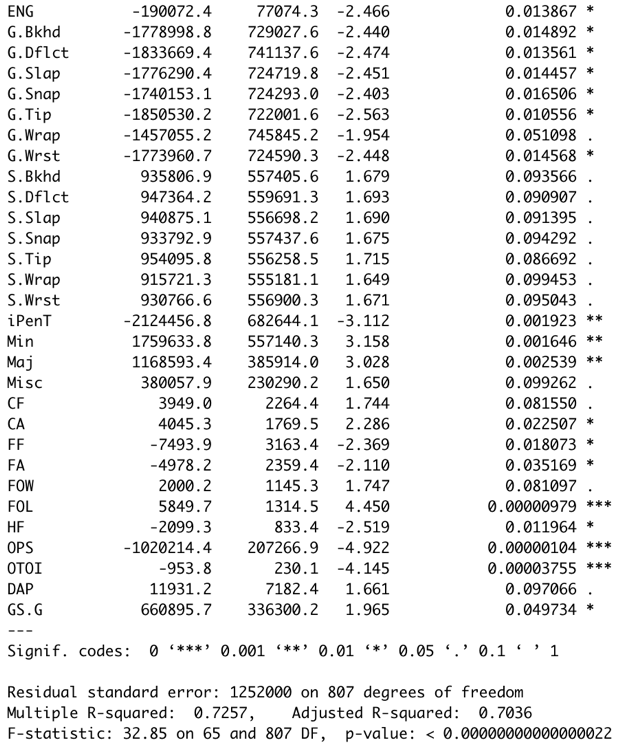
summary(hock.lm.step)



hock.lm.backstep <- step(hockey.lm3, direction = "backward")

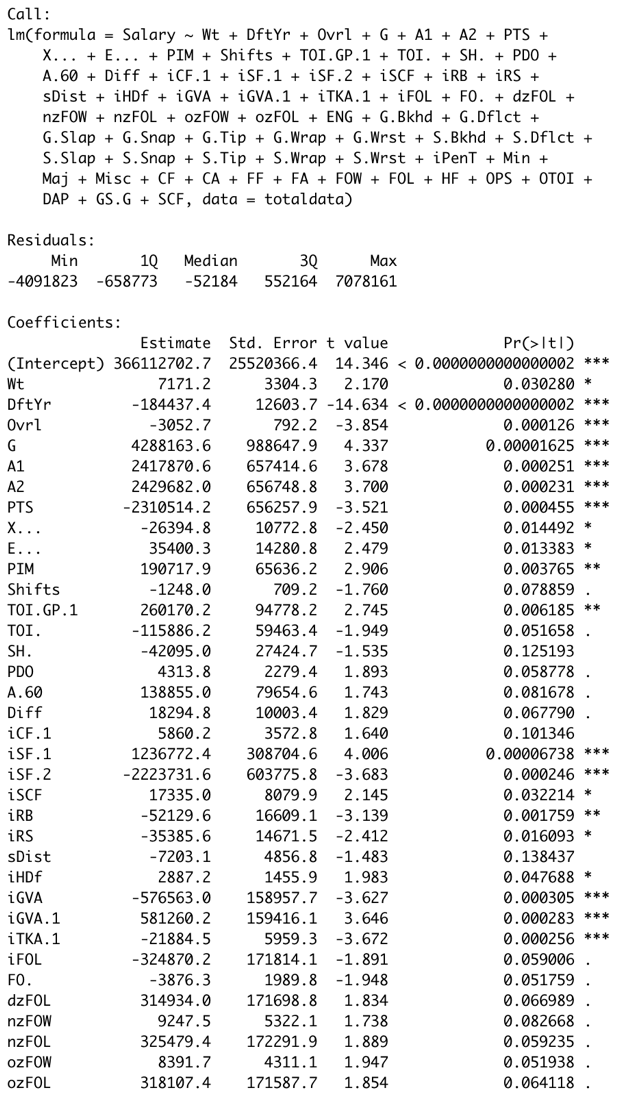
summary(hock.lm.backstep)





hock.lm.stepboth <- step(hockey.lm3, direction = "both")

summary(hock.lm.stepboth)



Table

Description automatically generated

> accuracy(hock.lm.step)

ME RMSE MAE MPE MAPE MASE

Training set 0.0000000001726422 1264781 897074.9 -25.44259 63.25859 0.4875605

> accuracy(hock.lm.backstep)

ME RMSE MAE MPE MAPE MASE

Training set -0.000000005739844 1203271 865932.7 -23.67119 61.34498 0.4706348

> accuracy(hock.lm.stepboth)

ME RMSE MAE MPE MAPE MASE

Training set 0.00000000004986698 1201867 866071.1 -23.54794 61.3689 0.47071

I decided to use the set of variables chosen by forward selection to be used in all of the models. This selection of variables gave the highest accuracy with the most reasonable amount of variables, which I then believe would help create models that would best be able to answer the business question. What should NHL players be being paid, and have they previously been over or undervalued.

pcs.cor <- prcomp(na.omit(totaldata), scale. = T)

summary(pcs.cor)

Three of the first 6 variables that have the most impact according to the PCA analysis were also selected in the forward regression model. These three are xGF, FOL, and DftYr. This is not surprising, as I explain further into the dataset, as these variables are at the top of the random forest, regression tree, and linear regression models. This means that these varibles do have a lot of importance when predicting NHL player salaries.

## Preliminary insights

I described some initial insights underneath each graph, that were specific to each visualization. However, I also determined some larger picture insights to further analyze. For example, I see in the boxplot of salary over different draft years. You can see that in the more recent years (~ 5 years), the salary distribution is much smaller, and the average salary is much lower. This is due to rookie contracts, and as time goes on the better players earn more and more. I must analyze this to determine how it affects the predictions. There are also a lot more players with a lower salary. How will this affect the predictions? In the scatterplot of Salary and Points, I see a good relationship in that the higher the points the higher the salary. I thought this would create a good predictor variable for salary, but I actually found this to not be the case. The values that actually helped us the most were those that are altered by the amount of playing time the player actually gets.

## 

## Data cleaning

sum(which(is.na(totaldata)))

## [1] 11967975

I first ran the above code, to get a feel for the missing values in the dataset. As you can see, there are a lot, which I will handle throughout the data cleaning and preparation steps. The first element of data noise I thought to bring to the attention was out of date players, drafted in years before 1990. This data would skew the prediction because of the extreme increase in NHL and all professional sport salaries since then. However, this data was already cleaned to remove players drafted before 1990, so no extra cleaning was necessary in that area.

### Position Variable

totaldata$Position[totaldata$Position=="C/LW/C"] <- "Flex"  
totaldata$Position[totaldata$Position=="C/LW/RW"] <- "Flex"  
totaldata$Position[totaldata$Position=="C/RW/LW"] <- "Flex"  
totaldata$Position[totaldata$Position=="LW/C/RW"] <- "Flex"  
totaldata$Position[totaldata$Position=="LW/RW/C"] <- "Flex"  
totaldata$Position[totaldata$Position=="RW/C/LW"] <- "Flex"  
totaldata$Position[totaldata$Position=="RW/LW/C"] <- "Flex"  
totaldata$Position[totaldata$Position=="C/D"] <- "C"  
totaldata$Position[totaldata$Position=="D/LW"] <- "D"  
totaldata$Position[totaldata$Position=="D/RW"] <- "D"  
totaldata$Position <- as.factor(totaldata$Position)  
summary(totaldata$Position)

## C C/LW C/RW D Flex LW LW/C LW/RW RW RW/C RW/LW   
## 143 56 42 295 41 75 47 32 90 23 30

Originally, the Position variable of this dataset had many different labels (which are shown directly in the code above), in which only had one or two players in each category. These players that played multiple positions, I changed to a “Flex” position. With a couple other changes, I reduced the position variable into only 11 different categories, ranging from 23 to 295 players per position.

### Born Variable

totaldata$Born <- as.Date(totaldata$Born)  
  
totaldata$Born <- as.numeric(format(totaldata$Born, "%Y"))  
totaldata$Born[totaldata$Born==0] <- 100  
totaldata$Born[totaldata$Born==1] <- 101

The Born variable was presented as character, so I first had to change that to date, and then only record the year to include age as a variable. If I made this variable categorical, the model techniques would be difficult because of the massive amount of categories. I changed players born in 2000 to 100 and 2001 to 101, to kept the order throughout the data.

### Hand Variable

totaldata$Hand[totaldata$Hand== "L"] <- 1  
totaldata$Hand[totaldata$Hand== "R"] <- 0  
totaldata$Hand <- as.integer(totaldata$Hand)

I created dummy variables for the dominant hand variable, which will help us with the regression model.

## Data Preparation

totaldata <- totaldata[-c(3:6, 13:16)]  
  
totaldata$iCF[is.na(totaldata$iCF)] <- median(totaldata$iCF, na.rm = TRUE)   
totaldata$iSF[is.na(totaldata$iSF)] <- median(totaldata$iSF, na.rm = TRUE)  
totaldata$iHA[is.na(totaldata$iHA)] <- median(totaldata$iHA, na.rm = TRUE)  
totaldata$iPENT[is.na(totaldata$iPENT)] <- median(totaldata$iPENT, na.rm = TRUE)  
totaldata$SF[is.na(totaldata$SF)] <- median(totaldata$SF, na.rm = TRUE)  
totaldata$FOW[is.na(totaldata$FOW)] <- median(totaldata$FOW, na.rm = TRUE)  
totaldata$DftYr[is.na(totaldata$DftYr)] <- median(totaldata$DftYr, na.rm = TRUE)  
totaldata$DftRd[is.na(totaldata$DftRd)] <- median(totaldata$DftRd, na.rm = TRUE)  
totaldata$Ovrl[is.na(totaldata$Ovrl)] <- median(totaldata$Ovrl, na.rm = TRUE)  
totaldata$iFF[is.na(totaldata$iFF)] <- median(totaldata$iFF, na.rm = TRUE)  
totaldata$iRB[is.na(totaldata$iRB)] <- median(totaldata$iRB, na.rm = TRUE)  
totaldata$iDS[is.na(totaldata$iDS)] <- median(totaldata$iDS, na.rm = TRUE)  
totaldata$sDist.1[is.na(totaldata$sDist.1)] <- median(totaldata$sDist.1, na.rm = TRUE)  
totaldata$Ovrl[is.na(totaldata$Ovrl)] <- median(totaldata$Ovrl, na.rm = TRUE)  
totaldata$iHDf[is.na(totaldata$iHDf)] <- median(totaldata$iHDf, na.rm = TRUE)  
totaldata$GS.G[is.na(totaldata$GS.G)] <- median(totaldata$GS.G, na.rm = TRUE)  
totaldata$IPP.[is.na(totaldata$IPP.)] <- median(totaldata$IPP., na.rm = TRUE)  
totaldata$Pass[is.na(totaldata$Pass)] <- median(totaldata$Pass, na.rm = TRUE)  
totaldata$PDO[is.na(totaldata$PDO)] <- median(totaldata$PDO, na.rm = TRUE)  
totaldata$SH.[is.na(totaldata$SH.)] <- median(totaldata$SH., na.rm = TRUE)  
totaldata <- totaldata[-291, ]  
which(is.na(totaldata))

## integer(0)

write.csv(totaldata, file = "TotalHockeySalaryData.clean.csv", row.names = FALSE)  
regdata <- totaldata[,c(1,5,118,9,69,131,37,71,7,38,119,42,40,22,50,45,53,11,138,10,4,134,80,41,110,34,93,146,113,115,132,88)]  
str(regdata)

## 'data.frame': 873 obs. of 32 variables:  
## $ Salary : num 925000 2250000 8000000 3500000 1750000 ...  
## $ DftYr : int 2015 2012 2006 2010 2012 1997 2009 2010 2010 2011 ...  
## $ xGF : num 0.5 62 70.8 22 33.5 62.9 0.7 10.5 23.8 2.6 ...  
## $ GP : int 1 79 65 30 82 80 3 30 53 10 ...  
## $ ozFOW : int 0 0 35 0 2 0 0 40 13 11 ...  
## $ FOL : int 5 939 600 328 490 667 13 174 330 52 ...  
## $ iSF.1 : int 1 143 156 40 95 74 2 51 76 9 ...  
## $ FOW.Up : int 0 0 13 0 3 0 0 36 10 16 ...  
## $ Ovrl : num 18 15 7 3 16 156 53 47 42 201 ...  
## $ iSF.2 : int 1 143 156 40 95 75 2 51 76 9 ...  
## $ xGA : num 0.9 88.8 46.4 33.6 47.5 59.9 0.5 17.2 33.7 3.7 ...  
## $ iRS : int 0 9 20 4 10 3 0 6 1 1 ...  
## $ iSCF : int 0 7 64 2 35 7 0 9 31 3 ...  
## $ TOI.GP.1: num 7.16 23.17 18.95 20.31 12.93 ...  
## $ iHDf : num 1 -43 -15 0 105 -114 -2 -8 41 -7 ...  
## $ sDist.1 : num 49.3 46.3 26.3 51 26.4 41.9 38.2 37.6 25.4 28.4 ...  
## $ iTKA : int 0 22 26 4 36 11 2 12 17 0 ...  
## $ A : int 0 15 26 5 12 12 1 2 5 1 ...  
## $ OPS : num 0 -0.2 3.7 0 -0.1 0.6 0 0 -0.6 0 ...  
## $ G : int 0 2 19 1 7 5 0 4 4 1 ...  
## $ Wt : int 190 207 218 220 217 192 185 183 214 178 ...  
## $ GVA : int 1 284 168 86 157 223 3 50 85 15 ...  
## $ ENG : int 0 0 0 0 1 0 0 0 0 0 ...  
## $ iRB : int 0 7 16 1 8 2 0 1 4 0 ...  
## $ Misc : int 0 0 0 0 1 0 0 0 0 0 ...  
## $ iCF.1 : int 2 287 283 88 166 171 5 94 109 24 ...  
## $ Wide : int 1 51 51 15 20 28 1 21 21 8 ...  
## $ GS.G : num -0.38 0.18 0.57 0.2 0.27 0.26 0.18 0.21 0.04 0.21 ...  
## $ CA : int 12 1992 1051 605 992 1281 23 379 724 91 ...  
## $ FA : int 10 1423 826 467 720 972 15 291 548 73 ...  
## $ HF : int 1 749 340 198 512 348 1 192 292 52 ...  
## $ G.Wrap : int 0 0 0 0 0 0 0 0 0 0 ...

write.csv(regdata, file = "TotalHockeySalaryData.regression.csv", row.names = FALSE)

I used imputation for almost all of the missing values in the dataset. I noticed that observation number 291, barely had any values filled in, which would skew the model prediction data, so I deleted that player entirely. This leaves us with 873 observations in the dataset. I also got rid of the identification variables that aren’t needed in model creation, these variables include name, birthplace, and team. They include multiple variables for their name and where they were born, so this subtracts 6 unneeded variables from the dataset.

Now after dimension reduction, cleaning, and preparation, I have created an accurate and trustworthy dataset to base the prediction models off of. I created a new csv file for the cleaned and reduced dataset named “TotalHockeySalaryData.regression.csv” to make modeling easier.

# Expanded Final Modeling and Predictive Analytics and their Evaluation

## Trial Models

The first run of dimension reduction created mixed results with the prediction techniques. I ran both the linear regression and regression tree prediction model with this dataset, before realizing that I would need to look into dimension reduction further. I improved on these first models in order to provide the best possible prediction model for NHL teams to be successful in evaluated player salaries and building a successful team.

set.seed(1)  
train.rows.tr <- sample(rownames(wrongdata), dim(wrongdata)[1]\*0.6)  
valid.rows.tr <- setdiff(rownames(wrongdata), train.rows)  
train.data.tr <- wrongdata[train.rows, ]  
valid.data.tr <- wrongdata[valid.rows, ]  
  
hockey.lm.trial <- lm(Salary ~ ., data = train.data.tr)  
summary(hockey.lm.trial)

##   
## Call:  
## lm(formula = Salary ~ ., data = train.data.tr)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -6231768 -572217 -186338 451008 7259058   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -754230.32 1058707.88 -0.712 0.47655   
## Wt 8721.82 5300.62 1.645 0.10051   
## GP -54460.57 10027.92 -5.431 0.000000088 \*\*\*  
## G 30583.34 17472.39 1.750 0.08067 .   
## A -2169506.60 1590209.93 -1.364 0.17310   
## A1 2216658.88 1589938.03 1.394 0.16389   
## A2 2208792.93 1590744.95 1.389 0.16560   
## X... -7698.14 7459.76 -1.032 0.30260   
## PIM 6091.90 9375.22 0.650 0.51613   
## Shifts 165.09 997.21 0.166 0.86857   
## TOI 64.02 19.47 3.288 0.00108 \*\*   
## iCF 6202.98 2812.70 2.205 0.02789 \*   
## iHF 2354.66 2155.76 1.092 0.27525   
## iHA -7890.00 3198.10 -2.467 0.01396 \*   
## iMiss -19132.78 11334.36 -1.688 0.09203 .   
## iGVA -13322.33 7860.28 -1.695 0.09072 .   
## iPENT -37440.55 59626.28 -0.628 0.53035   
## Min 43397.32 47795.86 0.908 0.36433   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1566000 on 496 degrees of freedom  
## (10 observations deleted due to missingness)  
## Multiple R-squared: 0.5244, Adjusted R-squared: 0.5081   
## F-statistic: 32.17 on 17 and 496 DF, p-value: < 0.00000000000000022

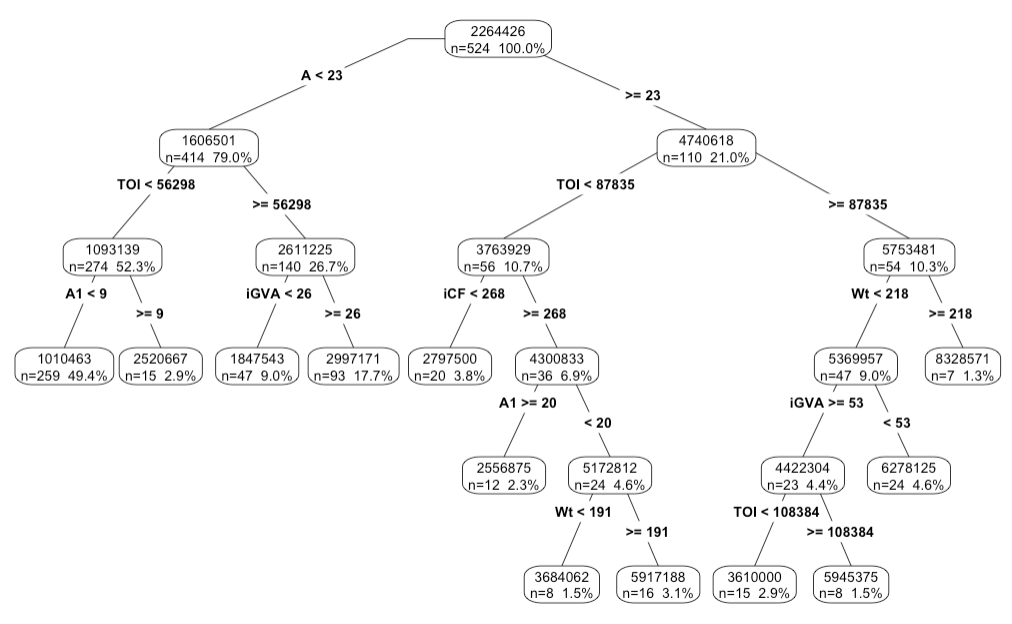
library(forecast)

## Registered S3 method overwritten by 'quantmod':  
## method from  
## as.zoo.data.frame zoo

hockey.lm.pred <- predict(hockey.lm.trial, valid.data.tr)

hockey.lm.pred.tr <- predict(hockey.lm.trial, valid.data.tr)  
accuracy(hockey.lm.pred.tr, valid.data.tr$Salary)

## ME RMSE MAE MPE MAPE  
## Test set 87024.77 1656290 1130648 -36.97942 66.60028

library(rpart)  
library(rpart.plot)  
reg.tree.trial <- rpart(Salary ~ ., data = train.data.tr, method = "anova")  
options(scipen = 999)  
prp(reg.tree.trial, type = 4, extra = 101, digits = -3)

accuracy(predict(reg.tree.trial, valid.data.tr), valid.data.tr$Salary)

## ME RMSE MAE MPE MAPE  
## Test set 164322.6 2048579 1311861 -41.9396 70.29484

The first prediction models came up with an RMSE of 1728677 for the linear regression model and 2111280 for the regression tree model. Through further exploratory data analysis and research I were able to reduce the average RMSE to around 1400000. The improved models are shown below.

## First Model

I chose to build a linear regression model to describe the relationship between the target variable, salary of NHL professionals, and multiple other dependent, predictor variables. This model will allow NHL team commissioners and owners to use certain predictor variables to determine a reasonable salary for their players, helping to solve the business problem. This is helpful to keep finances under control, and to find successful players for the right price. The goal to increase win percentage by 15% will be attainable if you can prevent overpaying the players that don’t perform for the team. Teams will be able to use this model to evaluate their players past performances also, and make decisions about salary or roster changes based on their findings.

cleandata <- read.csv("TotalHockeySalaryData.regression.csv")  
  
set.seed(1)  
train.rows <- sample(rownames(cleandata), dim(cleandata)[1]\*0.6)  
valid.rows <- setdiff(rownames(cleandata), train.rows)  
train.data <- cleandata[train.rows, ]  
valid.data <- cleandata[valid.rows, ]  
  
hockey.lm <- lm(Salary ~ ., data = train.data)  
summary(hockey.lm)

##   
## Call:  
## lm(formula = Salary ~ ., data = train.data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3943938 -643915 -59532 595501 6560617   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 334264151 31780575 10.518 < 0.0000000000000002 \*\*\*  
## DftYr -166425 15719 -10.588 < 0.0000000000000002 \*\*\*  
## xGF 21248 12540 1.694 0.090837 .   
## GP -46210 10025 -4.609 0.00000515 \*\*\*  
## ozFOW 9302 3713 2.505 0.012557 \*   
## FOL 4034 1438 2.806 0.005208 \*\*   
## iSF.1 1131592 321492 3.520 0.000472 \*\*\*  
## FOW.Up -9068 4095 -2.215 0.027249 \*   
## Ovrl -3424 1029 -3.328 0.000941 \*\*\*  
## iSF.2 -1139479 321639 -3.543 0.000434 \*\*\*  
## xGA -28456 20016 -1.422 0.155753   
## iRS -21901 19527 -1.122 0.262589   
## iSCF 6339 8485 0.747 0.455352   
## TOI.GP.1 93928 33232 2.826 0.004899 \*\*   
## iHDf 5494 1776 3.093 0.002091 \*\*   
## sDist.1 -5570 8183 -0.681 0.496398   
## iTKA -23368 7696 -3.036 0.002522 \*\*   
## A 119512 27537 4.340 0.00001731 \*\*\*  
## OPS -914955 258469 -3.540 0.000438 \*\*\*  
## G 141256 56659 2.493 0.012992 \*   
## Wt 3458 4242 0.815 0.415330   
## GVA -1241 2238 -0.554 0.579638   
## ENG -155475 101235 -1.536 0.125237   
## iRB -21044 21099 -0.997 0.319062   
## Misc 158608 150155 1.056 0.291352   
## iCF.1 13543 5785 2.341 0.019626 \*   
## Wide -29242 11444 -2.555 0.010912 \*   
## GS.G 568944 370655 1.535 0.125435   
## CA 5794 1964 2.950 0.003330 \*\*   
## FA -6154 2825 -2.178 0.029875 \*   
## HF -3079 1050 -2.933 0.003515 \*\*   
## G.Wrap 457111 240012 1.905 0.057425 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1266000 on 491 degrees of freedom  
## Multiple R-squared: 0.675, Adjusted R-squared: 0.6545   
## F-statistic: 32.89 on 31 and 491 DF, p-value: < 0.00000000000000022

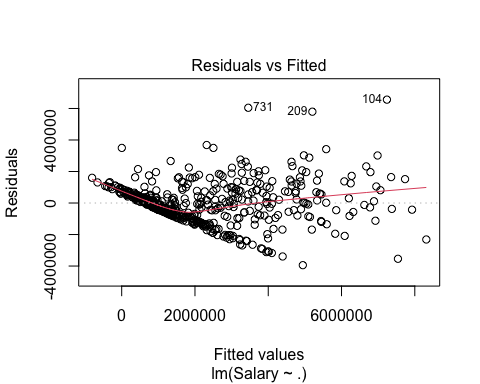
library(forecast)  
hockey.lm.pred <- predict(hockey.lm, valid.data)

accuracy(hockey.lm.pred, valid.data$Salary)

## ME RMSE MAE MPE MAPE  
## Test set 193754.4 1428035 959225.2 -17.01215 56.64067

options(scipen = 999)  
some.residuals <- valid.data$Salary[1:20] - hockey.lm.pred[1:20]  
data.frame("Predicted" = hockey.lm.pred[1:20], "Actual" = valid.data$Salary[1:20],  
 "Residual" = some.residuals)

## Predicted Actual Residual  
## 6 2825450.7 1500000 -1325450.73  
## 9 1668455.1 1250000 -418455.10  
## 10 339687.8 925000 585312.24  
## 12 529642.1 600000 70357.91  
## 13 1177674.1 1000000 -177674.14  
## 14 1357354.5 925000 -432354.48  
## 17 454067.4 590000 135932.57  
## 18 918393.1 650000 -268393.08  
## 21 3504613.5 5000000 1495386.49  
## 23 2569076.3 925000 -1644076.28  
## 24 1583318.4 1600000 16681.55  
## 26 2839759.1 925000 -1914759.15  
## 30 1113375.6 600000 -513375.63  
## 32 3605403.2 5500000 1894596.85  
## 34 595123.8 725000 129876.25  
## 36 953299.1 800000 -153299.10  
## 38 1715440.0 742500 -972940.00  
## 46 5633444.0 925000 -4708443.95  
## 47 959473.5 700000 -259473.46  
## 50 799290.7 925000 125709.27

plot(hockey.lm) 

The first measure I used for evaluating the linear regression model is the Multiple R-Squared measure which is shown in the regression summary. This value represents the amount of variance in the target variable that can be explained by the predictor variables in the model. The model produced a Multiple R-Squared of 0.675, which you can see in the summary above. The second measure I used for evaluating the linear regression model is the RMSE value by using the accuracy function. The RMSE measures the Euclidean distance between predictions and the true values. The larger the error, the more it is represented in the RMSE value because the value is squared. The RMSE value for the linear regression model came out to be 1428035. This may seem like a large amount, but with the highest paid NHL player coming in at 12.6 million, this number is pretty reasonable.

Partitioning the data when running the linear regression model reduces overfitting from the model. Using this model in the business world would be extremely easy being that you wouldn’t even need to partition, and you would still be able to predict without overfitting. It would be easy to update the dataset as the years go on and new data becomes available. It is also important to realize that data will become outdated, so reevaluating the dataset would be essential to creating good predictions.

## Second Model

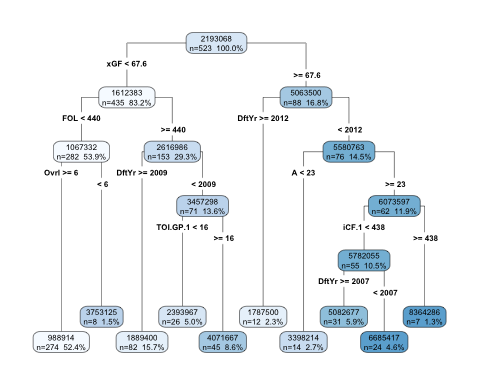
I also decided to use a regression tree model to make a visual representation of the predicted values of a player based on the most important stats to the model. This model fits the problem because it gives us a quick estimated salary and produces a visual representation of the model. This model will make it easier for members of the organization who have a poor understanding of data analytics to be able to gain some understanding of how to look at a player and their stats from an analytical standpoint with the visual and easy to follow nature of the regression tree. The hope is that I can create a baseline understanding of player value amongst the staff and make easier decisions based on players.

I envision that the tree will have the most important data point with a maximum of 10 branches and will give us a close data prediction to what the player’s salary should be. Just like the linear regression model, this will help NHL companies to solve their business problems by allowing them to evaluate players and their salaries. They will be able to budget their finances and learn how to create the most successful team within their salary cap. The great thing about this model is that it shows the visual representation of the prediction. Some may not “buy in” to the other models, but this one will allow them to see where the predictions are specifically coming from. The colors shown in this model also helps show where the higher salary players will fall, those that end in a darker blue terminal node.

library(rpart)  
library(rpart.plot)  
  
reg.tree <- rpart(Salary ~ ., data = train.data, method = "anova")  
reg.tree

## n= 523   
##   
## node), split, n, deviance, yval  
## \* denotes terminal node  
##   
## 1) root 523 2422327000000000 2193068.0   
## 2) xGF< 67.55 435 950025000000000 1612383.0   
## 4) FOL< 440 282 248950800000000 1067332.0   
## 8) Ovrl>=5.5 274 132118700000000 988914.3 \*  
## 9) Ovrl< 5.5 8 57439300000000 3753125.0 \*  
## 5) FOL>=440 153 462885400000000 2616986.0   
## 10) DftYr>=2008.5 82 150593300000000 1889400.0 \*  
## 11) DftYr< 2008.5 71 218748000000000 3457298.0   
## 22) TOI.GP.1< 16.03 26 56853310000000 2393967.0 \*  
## 23) TOI.GP.1>=16.03 45 115512000000000 4071667.0 \*  
## 3) xGF>=67.55 88 600556400000000 5063500.0   
## 6) DftYr>=2011.5 12 42675620000000 1787500.0 \*  
## 7) DftYr< 2011.5 76 408760000000000 5580763.0   
## 14) A< 22.5 14 34253080000000 3398214.0 \*  
## 15) A>=22.5 62 292758800000000 6073597.0   
## 30) iCF.1< 437.5 55 200079600000000 5782055.0   
## 60) DftYr>=2006.5 31 113050000000000 5082677.0 \*  
## 61) DftYr< 2006.5 24 52281150000000 6685417.0 \*  
## 31) iCF.1>=437.5 7 51273570000000 8364286.0 \*

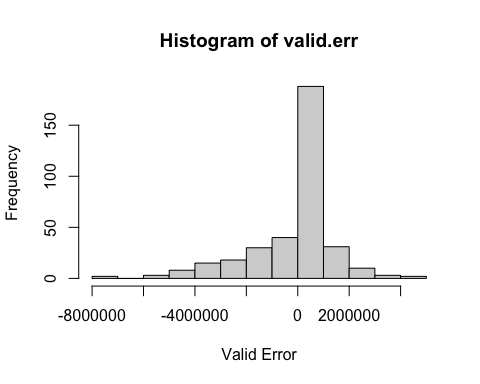
options(scipen = 999)  
rpart.plot(reg.tree, type = 4, extra = 101, digits = -3)



library(forecast)  
accuracy(predict(reg.tree, valid.data), valid.data$Salary)

## ME RMSE MAE MPE MAPE  
## Test set 237156.6 1635965 1065488 -23.99215 50.42687

library(ggplot2)  
train.err <- predict(reg.tree, train.data) - train.data$Salary  
valid.err <- predict(reg.tree, valid.data) - valid.data$Salary  
hist(valid.err, xlab = "Valid Error")



The first measure I used for evaluating the regression tree is the RMSE values. The RMSE for the validation set was higher than the training set by about 400,000, which makes total sense because the model was based off of the training set. The second measure I used for evaluating the accuracy of the regression tree is a histogram showing the distribution of error. I completed these accuracy measures for the training and validation datasets. As you can see in the visuals above, the error is highly concentrated closer to 0. I are happy with these results, as it shows the model has a larger amount of small errors than large errors. This is important so that NHL teams aren’t severely over or under paying their athletes, because this is when problems occur involving players leaving or wasting the salary cap.

For this specific model there isn’t an option to use boosting or ensemble, but the third model is a random forest model which is an ensemble boosting model based on multiple regression trees. Rather than just performing the single regression tree, the random forest model creates a tree for many random samples and combines the predictions to make the best one.

## Third Model

I decided to use a random forest model for the prediction as well because it works well with regression prediction. Random Forest builds many trees based on different samples of the data, and then uses the averages of the results of each sample to determine the results of the prediction. Random forest is a form of bagging, which includes taking resamples, as replacements, from the original data.

This random forest model will not only solve the business problem by created an accurate model to predict player salaries, but it will also provide NHL teams with a solid understanding of where the predictions are coming from. This model directly shows the most influential stats to determining salary, which should then be focused on by scouts trying to find a team for next season. In the past, scouts would assign high value to categories like weight and age, but I want to change that thinking to valuing players based on other, more important variables, which are displayed by the randomForest plot.

Random forest models are popular in many different prediction situations in real life. Some examples may be to identify components of medicine and in e-commerce to determine whether or not customers like a product. Random forest can predict continuous data as well, like the salary prediction situation. The random forest regression model will create a successful model without a ton of fine tuning, that if used correctly, will help NHL teams correctly predict salaries and create a more successful and financially stable team.

library(randomForest)

## randomForest 4.7-1.1

## Type rfNews() to see new features/changes/bug fixes.

##   
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':  
##   
## margin

rfdata <- cleandata  
  
rf <- randomForest(Salary ~ xGF + DftYr + GP + ozFOW + FOL + iSF.1 +   
 Ovrl + iSF.2 + xGA + iRS + iSCF + TOI.GP.1 + iHDf +   
 sDist.1 + iTKA + A + OPS + G + Wt + GVA + ENG + iRB + Misc +   
 iCF.1 + Wide + GS.G + CA + FA + HF + G.Wrap, data = train.data, ntree = 500, mtry = 4, nodesize = 5, importance = TRUE)  
print(rf)

##   
## Call:  
## randomForest(formula = Salary ~ xGF + DftYr + GP + ozFOW + FOL + iSF.1 + Ovrl + iSF.2 + xGA + iRS + iSCF + TOI.GP.1 + iHDf + sDist.1 + iTKA + A + OPS + G + Wt + GVA + ENG + iRB + Misc + iCF.1 + Wide + GS.G + CA + FA + HF + G.Wrap, data = train.data, ntree = 500, mtry = 4, nodesize = 5, importance = TRUE)   
## Type of random forest: regression  
## Number of trees: 500  
## No. of variables tried at each split: 4  
##   
## Mean of squared residuals: 1871930671702  
## % Var explained: 59.58

accuracy(predict(rf, valid.data), valid.data$Salary)

## ME RMSE MAE MPE MAPE  
## Test set 190992.5 1612251 1048167 -30.91069 54.19301

The first measure I used for evaluating the random forest model is the percentage of variance explained, which is shown in the random forest call output. This model shows 58.08% of variance explained. The second measure used for evaluating the random forest model is the RMSE. The model based on the validation data comes in at 1611905. I knew that this wasn’t exactly what I were hoping for, but altered the model a bit to create a better outlook, still using the same variables. In the next run I decide to use 20 variables tried at each split, rather than 4.

set.seed(1)  
train.rows2 <- sample(rownames(regdata), dim(regdata)[1]\*0.6)  
valid.rows2 <- setdiff(rownames(regdata), train.rows)  
train.data2 <- regdata[train.rows2, ]  
valid.data2 <- regdata[valid.rows2, ]  
  
random <- randomForest(Salary ~ ., data = train.data, mtry = 20)  
print(random)

##   
## Call:  
## randomForest(formula = Salary ~ ., data = train.data, mtry = 20)   
## Type of random forest: regression  
## Number of trees: 500  
## No. of variables tried at each split: 20  
##   
## Mean of squared residuals: 1715719588698  
## % Var explained: 62.96

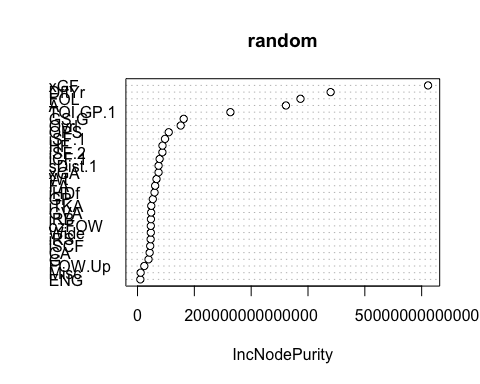
accuracy(predict(random, valid.data), valid.data$Salary)

## ME RMSE MAE MPE MAPE  
## Test set 211937 1474153 922368.4 -22.40262 44.76885

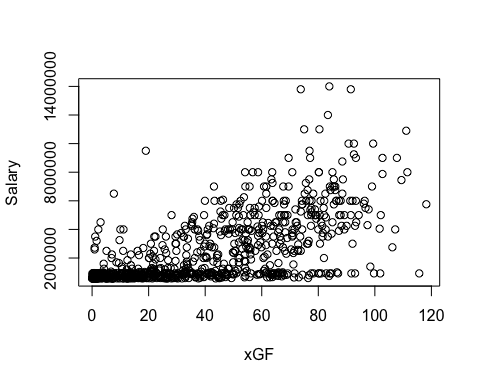
As you can see the % of variance explained has improved to 63.47% and the RMSE is down to 1474844. This is significantly better than what I originally had, and improving on the model will better the predictions. As I further evaluate the model, I must acknowledge how I are staying away from overfitting. With this model, if I predict salary based on the full dataset, rather than partitioning, the RMSE still comes out to about 700,000. This number is still large, but would represent an overfitted dataset. In hockey, there is so much variance and unknown, that accuracy measures tend to be slightly lower.

The visual below is important for each NHL team, because it shows which values are the most important when it comes to prediction. As mentioned before, NHL teams should focus more on a players xGF (the teams expected goals while player is on the ice) rather than measures like age and weight. In the scatter plot, you can see exactly how correlated these two measures are. The higher the xGF the higher salaries each player tends to have.

varImpPlot(random)



plot(regdata$xGF, regdata$Salary, xlab = "xGF", ylab = "Salary")



# Deployment

This model can be deployed multiple different ways based on what the front office would like. One way would be the method of evaluating players inside of an organization and using the model to give each player a total value of their performance and compare this to how much they were paid during the year and if this player has been over or under paid. Taking this difference I can mark each player as a deal that is underpaid, properly paid, or a money drain on the organization that might need to be traded or released. This will help the organization streamline managerial decisions and reduce friction within the organization when it comes to personal choices.

This model can also be highly effective when it comes to finding players outside of the organization that I might want to sign because they are undervalued by the rest of the league. I can take a list of the players in the league that might be looking to change teams or organizations are willing to trade for and create a list of the players that on their current contract are being paid far less than their actual worth.

For an example I took a random player from the data set and plugged it into the linear regression. This was the result.

playerdata <- read.csv("TotalHockeySalaryData.csv")  
testplayer <- playerdata[6,]  
pred <- predict(hockey.lm, testplayer)  
pred

## 6   
## 2825451

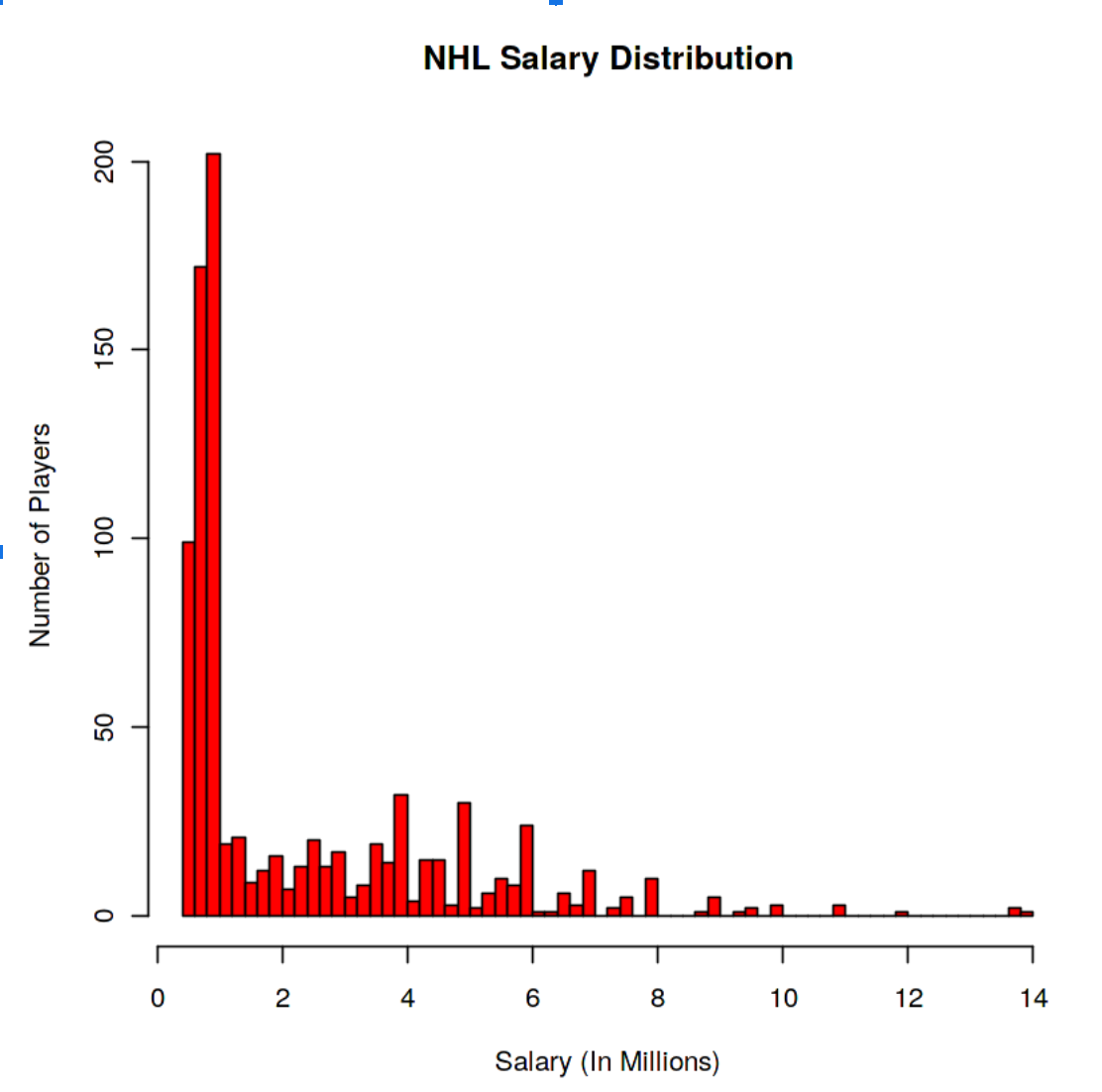
This result is $881,902 higher than the actual salary the player was paid. Looking at the player I tested this underpayment makes sense. The player is Brian Cambell and this was his final season in the NHL playing with the Blackhawks. He openly took a much lower salary so the Blackhawks were able to pay other players more money so they could contend for the NHL championship. While he took a lower salary he still played like a player deserving of a much higher salary.

One of the best tools I can use to help members of the front office who don’t have a strong understanding of the data analytics world would be to use a regression tree. The regression tree gives us an easy to understand visualization of the model that I can give to scouts who might be interested in how I are evaluating players. While it is not as accurate as the other models the ease of understanding makes regression trees valuable to the business understanding.

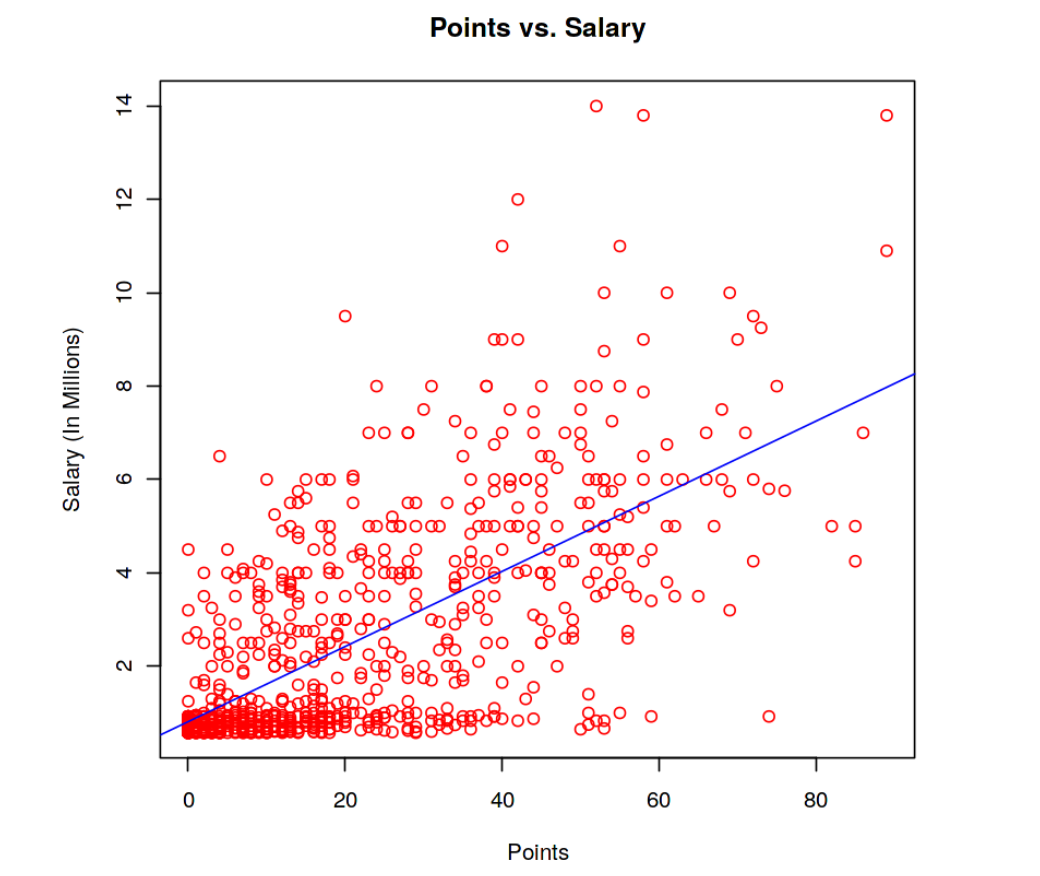
I don’t foresee many ethical issues with the dataset and models as every piece of data is given voluntarily. I think that some NHL owners and GMs might have an issue with possibly over relying on the models and not taking into account that hockey by nature is a sport that has aspects of randomness. Because of the model they might overpay a player that under performed because of personal issues or they just had an unlucky season. Overall, using these models can give NHL owners the ability to compensate employees more precisely. One potential downside that may be concerning in the use of the models is using it to underpay players that may be supposed to be getting paid higher. With most players not having access to these salary calculating methods, this leaves room for players to be taken advantage of and not paid justly. Again, it is important to remember that these models should be used as a leverage to help pay players/employees fairly rather than create injustices in salary within the NHL.

# Comparisons with documented results

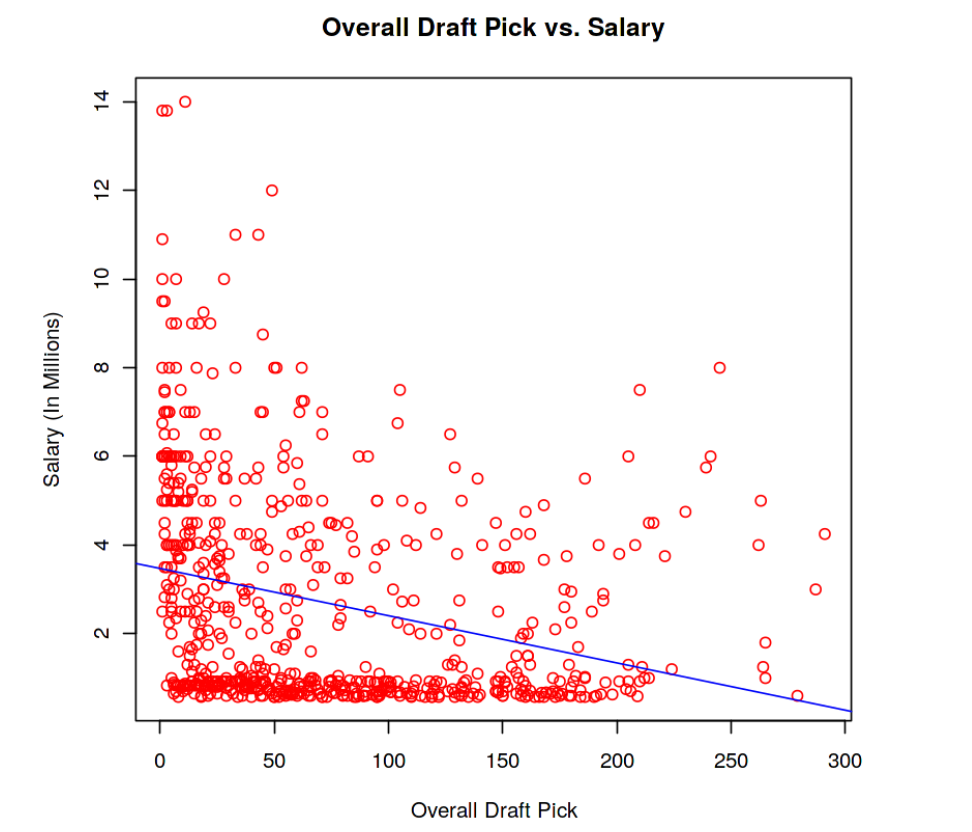
Looking at the results from the first similar project published by Brad Kassen, who is a data scientist, I can see that although their model seems to be slightly more accurate, they are very close. The goal with their project is to create models that can help us evaluate players from a managerial perspective for decision making by using the models to determine their worth, which is the same as the story that I are trying to tell with this project. In terms of visuals, I were also pretty similar with his project. I both visualized important statistics such as draft year and points in comparison with salary, although the visualizations themselves were unique to each other. Another visual that I had in common was looking at the distribution of the salaries throughout the league, and this one is actually the exact same type of chart, just simply in a different color. From both visuals I can see that vast majority of players are not getting paid crazy amounts, while the top players in the league make substantially more, which can definitely creates difficulties in getting the most accurate model possible.



The first visual I will discuss is his histogram on the distribution of the money in the NHL. I did virtually the same thing in the project just with slightly different formatting and this visual shows us how skewed the data is, as vast majority of people in the NHL are getting paid on the lower end, while the high earners are really outliers in the grand scheme of the whole league.



Another visual that I will discuss from his project is scatterplot comparing points (goals/assists) to salary. I had the same idea as that is what common sense would tell you should be directly correlated with salary, and as I can see from both of the visuals there is definitely a correlation, although it is not as strong as one may think.



The final visualization he used that I will be discussing is his scatter plot of draft year comparing it to salary. This is a visual where I differentiate in terms of type of visual as he had a scatter plot while I made a box plot. Both work well for the purpose and as I can see there is definitely a correlation between when you are drafted and how much money you make, which makes sense from an outside standpoint.

In terms of accuracy of the models ran by the similar project examples, they are quite like mine. However, they ran several different types of regression models, that were different from the simple linear regression. The RMSE’s that he got for all five of the different regression models were in the range of 1,440,000 to 1,520,000, and the RMSE of the linear regression was $1,589,654. This difference most likely stemmed from the slight difference in the variable choice and the fact that I ran a linear regression, while he ran multiple slightly different forms of regression for this project.