STAT 444: Final Project

Winter 2018

Team F

Salary Prediction for National Hockey League Players

Authors:
Marcus Di Renzo
Alessandra Iaboni
Galen Wray

Professor:
Kun Liang

April 10, 2018

Contents

1 Introduction

The central purpose of our analysis is to predict the salary being paid to professional hockey players in the NHL (*National Hockey League*) based on the circumstances surrounding their signings. All players are given contracts for different lengths of time and for different amounts of money paid out over those time periods. When a player's contract comes to an end, they are required to sign a new contract, referred to as a *player signing*. The purpose of this report is to predict the player's salary, independent of which team they sign with. This will provide individual players with a 'market value' that can be used to help guide trading decisions, and funding allocations for particular teams.

Each year within the NHL, based on the revenue of the previous year, each team is assigned a *salary cap*, which is their maximum allowable budget to spend on player salaries. Thus, when selecting players for a particular team, managers must determine which players will provide the greatest asset to the team, while also having an affordable salary. However, there is often a discrepancy between the salary a player is given, and the salary a player deserves based on their historical on-ice performance.

External bias can arise as a consequence of a player's behaviour off the ice, possible existing relationships between particular players and managers, or expectations of future performance, which can make player signing an imperfect process. Designing a statistical model that is capable of objectively predicting player salaries can reduce potential sources of bias. A recruiter with access to this model would be able to predict the salaries that a player will be valued at, and leverage this information to help develop negotiations, identify players of interest for the season, and develop a reasonable salary allocation to their players for the season. Free agents in the league with metrics comparable to existing players can be assigned a fair and justifiable salary, providing opportunity to recruit skilled players with low cost. In addition, the expected salary of a player (as predicted by their performance) versus their actual salary can be compared, giving insight into which players are being historically under or overpaid.

Though the topic of NHL salary prediction has been explored in previous literature, there is no equivalent studies that analyzed an identical set of covariates and modelling techniques outlined in this report. This occurs largely as a consequence of the data collection methods used in creating the initial data set. The data analyzed in this report is derived from multiple sources that were manually joined, and thus differs from previous studies in which data was commonly pulled from a single source. Though previous Kaggle competitions and theses have provided reasonable starting points, the problem we have chosen to analyze is unarguably more complex. We have not restricted our data to originate from a single source, considered a larger breadth of potential models, and analyzed the relationship between salary and a larger subset of potential covariates. In previous approaches, similar predictors have been analyzed with the intent of explaining player characteristics that contribute to salary earnings. These analyses, however, were based on same year statistics and were predominantly aimed at being explanatory rather than predictive. Our problem, is thus different since our aim is to use previous years' data as indication of the salaries players will earn in the future. More information regarding the previous work on NHL salary prediction can be found in the literature review within Appendix B.

2 Data

2.1 Data Collection

For this paper, the data are collected from numerous sources. First, we collect data on NHL player signings by scraping information from CapFriendly. From this source, we take the player name, date of signing, and the cap hit, our target variable. Only standard contracts are analyzed in this report because of the restrictions on other contract types, such as entry-level contracts and 35+ contracts. For example, entry-level contracts not only have low maximum salaries, but they are also the first contract they sign in the NHL, so there would be no previous NHL data to predict from. As predictors for our analysis, we gather data from Hockey Abstract, which produces annual tables of combined player statistics from many sources. The available variables vary each year depending on which statistics were 'popular'. As a result, some manual

matching must be done when combining these files. That being said, the typical statistics, such as points, are consistent year over year. The last variables we collect are the annual NHL salary caps and final dates of each NHL season. Such data can be collected from news sources, such as CBC and ESPN.

Once the data were collected, we linked the files to achieve our final dataset. To do so, we had to match by name between the signings data and previous seasons. We considered the last date for a NHL season's signings to be the last day of the season. Signings after that day are considered to be signings for the next year. We acknowledge that some signings may not be for the next year and thus lead to some inaccuracies within our final dataset. However, signings for year(s) ahead are infrequent. Moreover, statistics will likely be relatively consistent between the years. During the matching process, any signing that does not have a match in the previous NHL data is removed. As a result, we must ensure that names are consistent when doing the matching (i.e. nicknames and accents are the same) so that we do not unnecessarily remove players from our final dataset. Once signings and past player statistics are merged, we add our final variable, salary cap for the year of the signing, by year.

With our merged dataset, we limit the first observation date to start with predictor variables from the 2013-2014 NHL season such that we avoid the 2012-2013 half lockout in the NHL. Including a year without a full season would require some adjustments in order to be consistent with other seasons. Rather than risking the integrity of our model by using possibly incorrectly adjusted data, we simply start our dataset the following year. The dataset includes signings since the end of the 2013-2014 season until February 20, 2018.

2.2 Data Summary

In order to predict salaries, a total of 15 covariates were considered. This section of the report defines each of the 15 initial covariates included within all of the models, along with the motivation of the inclusion of each particular variable. All plots and summaries within this section refer to the raw data, prior to pre-processing, unless otherwise specified. A more detailed analysis of potential themes of interest within the data set are explored within the data section of the Appendix following this report. Appendix A will also discuss potential variables that were excluded from our analysis.

Below is a table outlining some descriptive statistics for each of the numerical covariates included in the models:

Table 1: Summary of Numerical Predictors

	0%	25%	50%	75%	100%	Mean	St. Dev.
Salary	550000.00	650000.00	815000.00	2200000.00	12500000.00	1729528.38	1775711.33
Games Played	1.00	13.00	49.00	73.00	84.00	44.34	29.08
Games Injured	0.00	0.00	0.00	8.00	68.00	5.82	10.06
TOI/G (Min)	2.72	11.03	13.64	16.82	25.86	13.91	4.02
Points	0.00	2.00	9.00	24.00	100.00	15.62	17.35
Goals	0.00	0.00	3.00	9.00	41.00	5.99	7.48
Age (Days)	7161.00	8690.00	9277.00	10272.00	14116.00	9562.08	1150.87
Takeaways	0.00	2.00	10.00	23.00	98.00	15.03	15.22
Giveaways	0.00	3.00	12.00	27.00	102.00	17.49	17.55
Rel. Corsi	-122.85	-5.07	-0.03	2.37	48.10	-1.74	10.87
Penalty Min.	0.00	4.00	17.00	32.00	238.00	23.47	26.04
Hits	0.00	16.00	44.00	86.00	365.00	58.44	54.53
+/-	-38.00	-4.00	-1.00	3.00	39.00	-0.61	8.56

2.2.1 Target (TARGET)

An individual player's cap hit, measured in dollars, which we refer to as the 'salary' or the 'target' for the remainder of this report. This variable was used as the response variable within the data set.

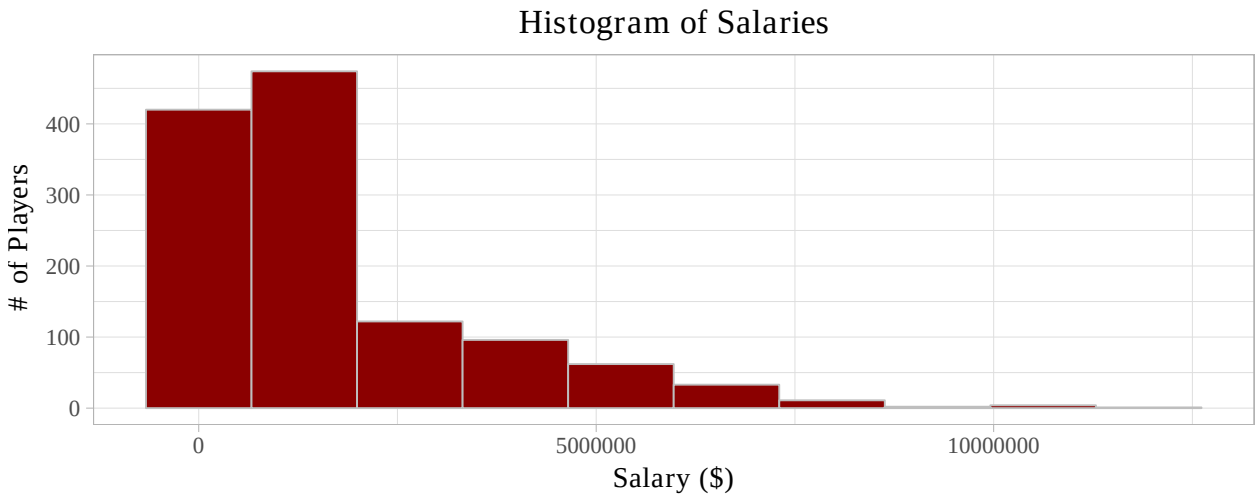


Figure 1: Descriptive Plot of Salaries

2.2.2 Salary Cap (SALARY_CAP)

Salary cap refers to the NHL salary cap imposed on each team within the league, as measured in millions of dollars. The salary cap is the maximum amount of salary that a team can distribute among its players, and is the same for all teams within a given year. The salary cap may vary year-to-year as a consequence of the revenue attained by the NHL in the previous year, and indirectly provides information on the year in which the player was signed to their current team.

Salary cap was included in the data set since there is a difference of nearly 10 million dollars between the first and last year. The increase in salary cap means there is more funds available to allocate to player salaries, which may lead to differences in individual player salaries. As opposed to using a categorical variable to indicate the year of signing, which is not extendable to the following year, we use the salary cap, a continuous variable, as a proxy for the year each player was signed.

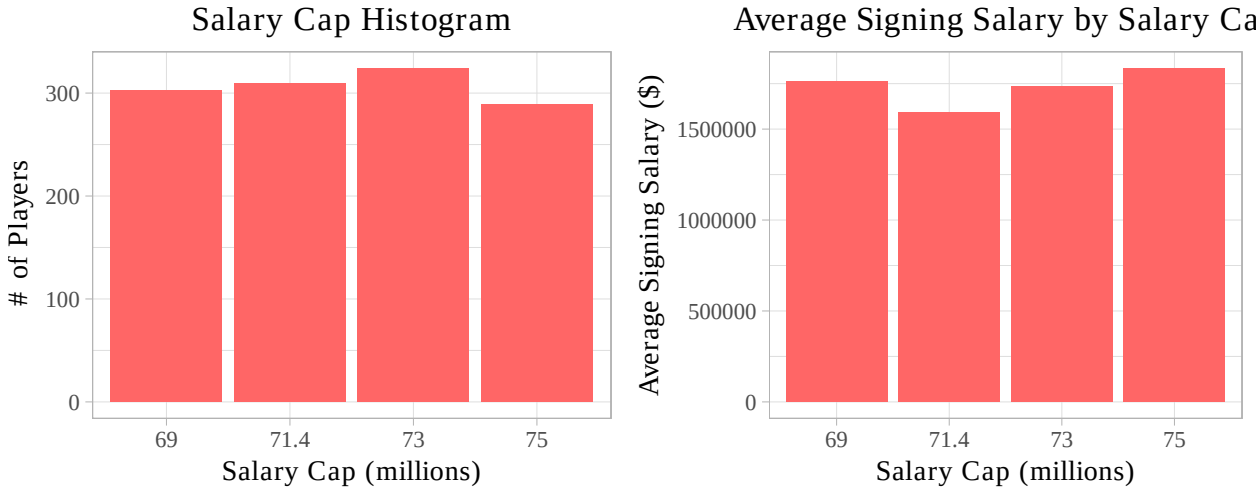


Figure 2: Descriptive Plots of Salary Cap

2.2.3 Games Played (GP)

Quantitative variable referring to the number of games played by that individual during a season. A game is considered player by an individual player only if the player had at least one shift on the ice. All players included in the data set had at least a minimum of one game played.

Players may have a higher number of games played than others will tend to have more experience, are selected to play in a larger proportion of games than their teammates, or are less prone to injury. As a result, these differences in player quality and experience may be reflected in salary.

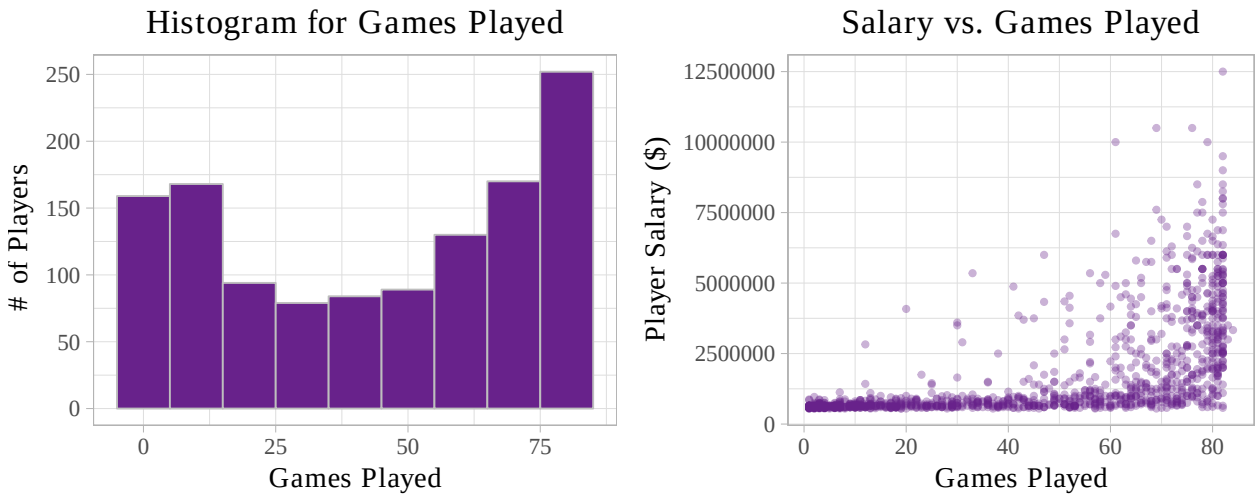


Figure 3: Descriptive Plots of Games Played

2.2.4 Games Injured (INJURED_GAMES)

Quantitative variable referring to the number of games an individual player was unable to participate in as a direct consequence of an injury.

Players who are frequently injured may be undesirable to sign, as their impact within the team may be limited. Moreover, games missed due to injury should be treated differently than those that were missed if a player was a 'healthy scratch' (i.e. they sat out of the game because they haven't been playing well). As a consequence, it seems reasonable that the salary of a player may be influenced by their historical injury frequency, particularly relative to the number of games played.

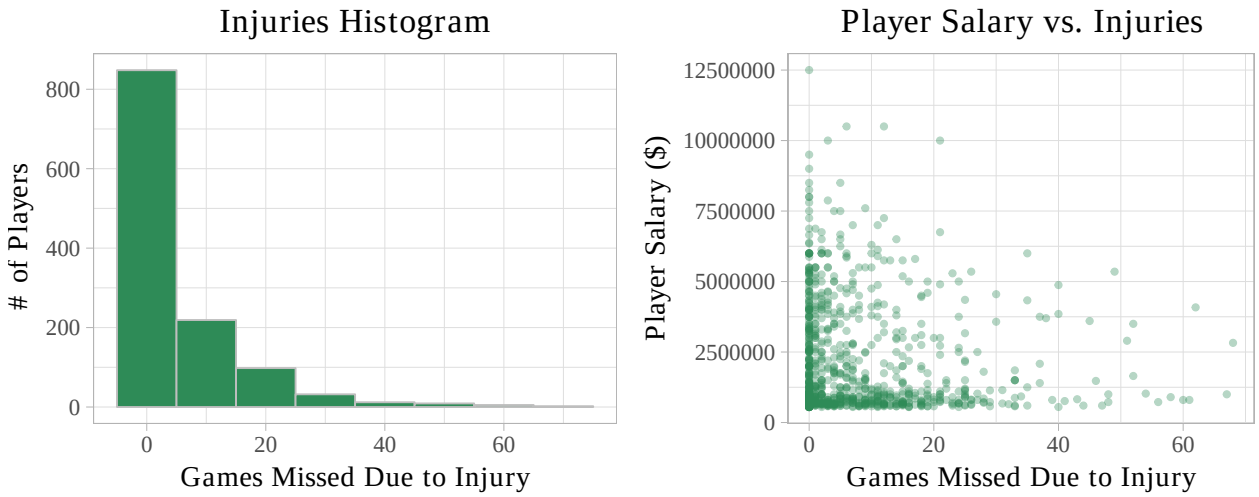


Figure 4: Descriptive Plots of Games Injured

2.2.5 Time on Ice per Game (TOI.GP)

This variable refers to an individual player's average time on ice per game, measured in minutes. Similar to games played, time on ice was included since it is often an indirect insight into the strength of a player. If a player is valuable asset to the team, they are likely to be chosen to be on the ice more often than others, and thus a larger salary seems reasonable.

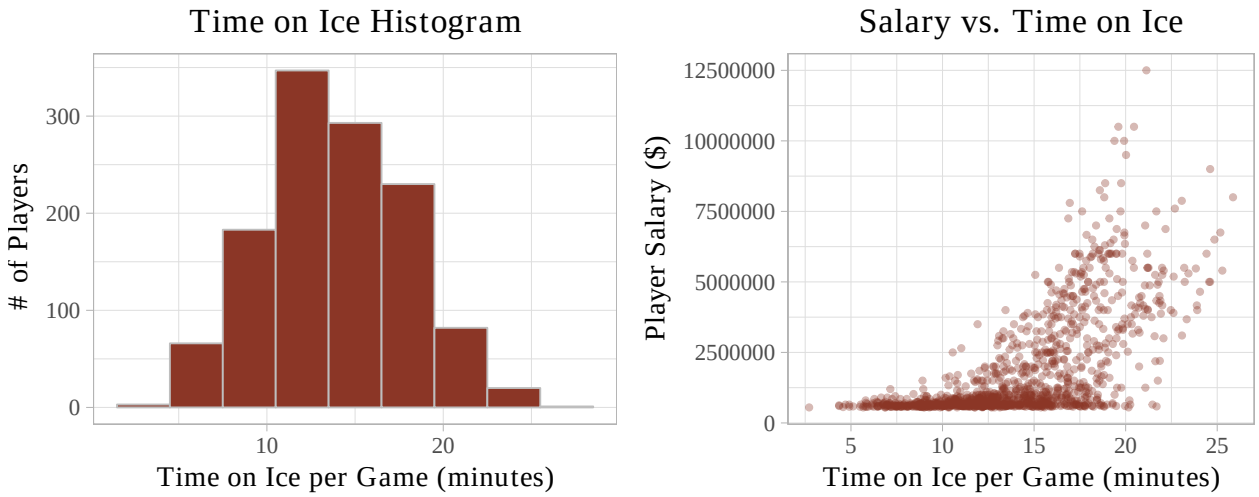


Figure 5: Descriptive Plots of Time on Ice per Game

2.2.6 Points (P)

Denotes the total number of points accumulated by an individual player. Points refer to the total number of *goals* (the last player to touch the puck before the puck enters the opposing goal line is awarded a goal) and *assists* (awarded to up to 2 players per goal who had touched the puck prior to the goal scorer, assuming no opposing players touched the puck between players) a player achieves.

Players capable of achieving a higher number of average points are beneficial, since they improve the likelihood of their team winning. As a consequence, these players tend to be highly desirable, which may drive bidding wars between managers and leading to higher salaries.

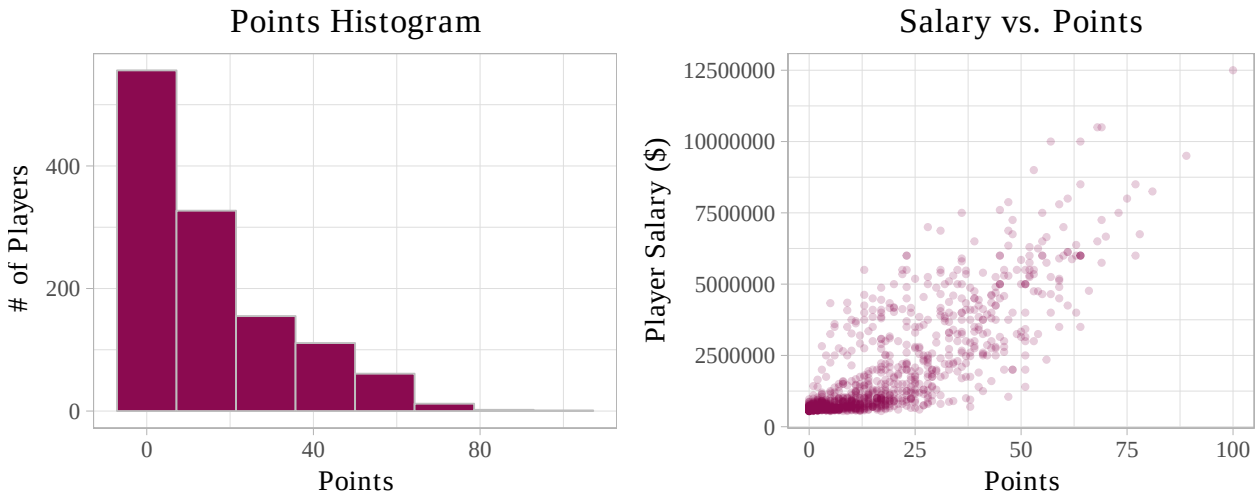


Figure 6: Descriptive Plots of Points

2.2.7 Goals (G)

Total number of goals achieved by an individual player. Similar to points, players with a high number of average goals are desirable, and it is sensible that there would be some positive relationship with regards to salary worth investigating. Although goals has similar characteristics as points, both variables were included because goals are valued differently than points. For example, the distribution of goals and points can tell us whether a player is a 'sniper', a player who scores a lot of goals, or more of a 'playmaker', a player who will have more assists, and these players may be paid differently.

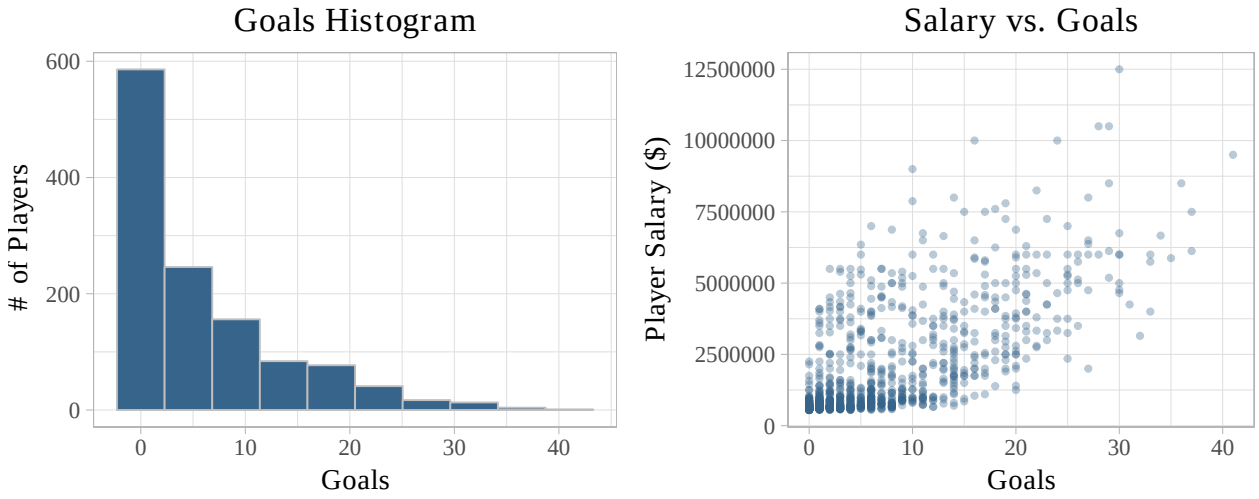


Figure 7: Descriptive Plots of Goals

2.2.8 Position (POS)

Categorical variable indicating the position a particular player commonly takes on the ice. A full explanation of the different positions and their purpose can be found within the following SportsNet article. Forwards (either right-wingers, left-wingers or centers) are typically more offensive, while defensemen are more defensive. As a result, we expect different relationships between variables for each position.

Table 2: Descriptive Statistics of Position

Position	Total Number of Players	Average Signing Salary ($)
C	250	1,894,642.8
C/D	1	620,000.0
C/N	1	750,000.0
C/RW	75	1,618,456.7
D	414	1,758,552.4
D/LW	3	679,166.7
D/RW	1	8,000,000.0
LW	124	1,688,068.7
LW/C	108	1,623,718.8
LW/C/RW	24	2,046,213.5
LW/RW	96	1,520,154.5
RW	128	1,597,289.1

2.2.9 Nationality (NAT)

Categorical variable indicating the nationality of the player. Nationality was selected as a covariate of interest since within the data set, it appeared that there was a considerable difference in salaries of North American players versus those from other regions. This is likely due to two factors. First, hockey is not as popular outside of North America. As a result, only very talented players are able to make the jump to the highest-level of professional hockey in the NHL. Second, players from outside North America are more likely to play in international leagues that may be able to pay them better or just as well, like the Kontinental Hockey League (KHL).

Table 3: Descriptive Statistics of Nationality

Nationality	Number of Players	Average Signing Salary ($)
AUT	7	2,707,143
CAN	649	1,597,875
CHE	15	1,319,444
CZE	31	2,688,413
DEU	11	2,434,091
DNK	7	3,807,143
FIN	28	2,242,530
FRA	4	1,350,000
HRV	1	575,000
ITA	1	3,600,000
LTU	1	700,000
LVA	3	1,116,667
NOR	4	2,328,750
RUS	29	3,045,774
SVK	13	2,429,615
SVN	1	10,000,000
SWE	77	1,974,636
USA	343	1,569,835

2.2.10 Age (AGE)

Age of an individual player, measured in days. It is reasonable to assume that age has at least some influence on the salary of a player, since younger players may lack considerable experience within the NHL, so pay may come more from expectations as opposed to talent. Older players, on the other hand, may be more likely to have injuries or impairments as a consequence from a long-term career in the NHL. Moreover, since they are past their prime, it is often expected that their performance will only go downhill, which may lower their salary.

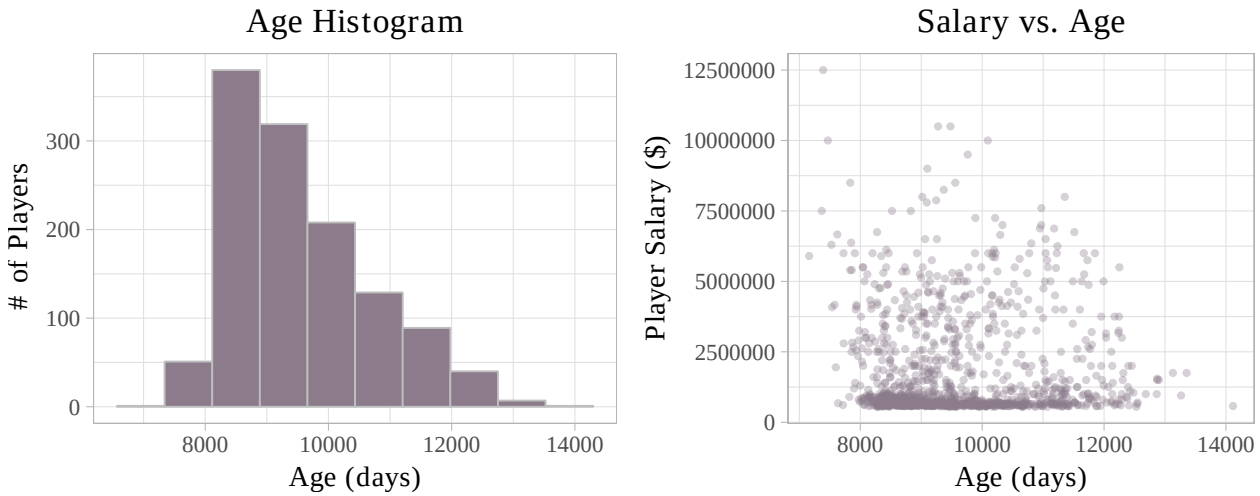


Figure 8: Descriptive Plots of Age

2.2.11 Takeaways (TKWY)

The total number of takeaways achieved by a player, in which takeaways are defined to be a form of turnover in which the player takes the puck from the opposition, rather than gaining possession through an opposition error. Takeaways are of importance since they help teams maintain control of the puck, which may influence the probability of scoring goals on the opposing team, or preventing goals on their own net. Players with a high number of average takeaways are thus desirable for managers, which will influence their salary allocation.

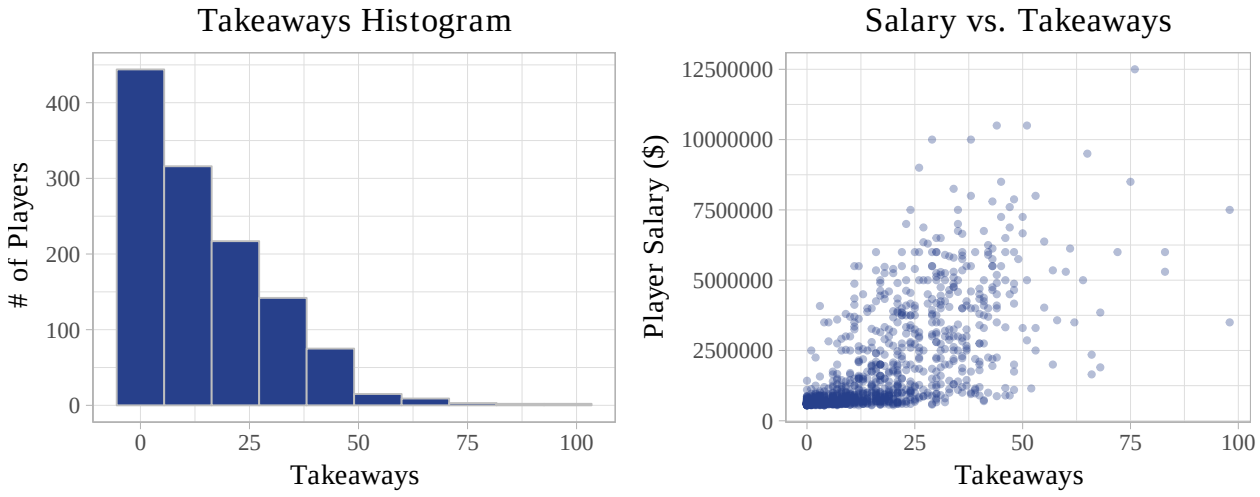


Figure 9: Descriptive Plots of Takeaways

2.2.12 Giveaways (GVWY)

Giveaways achieved by a player. A giveaway is a form of turnover in which a player makes an error resulting in the opposing team gaining position of the puck. Players with large numbers of giveaways are less attractive to teams because this means that they will frequently give up possession of the puck to the opposing team. Although giveaways are increasing as salary increases in the below figure, it is likely that this is because it is confounded by the number of games played by players.

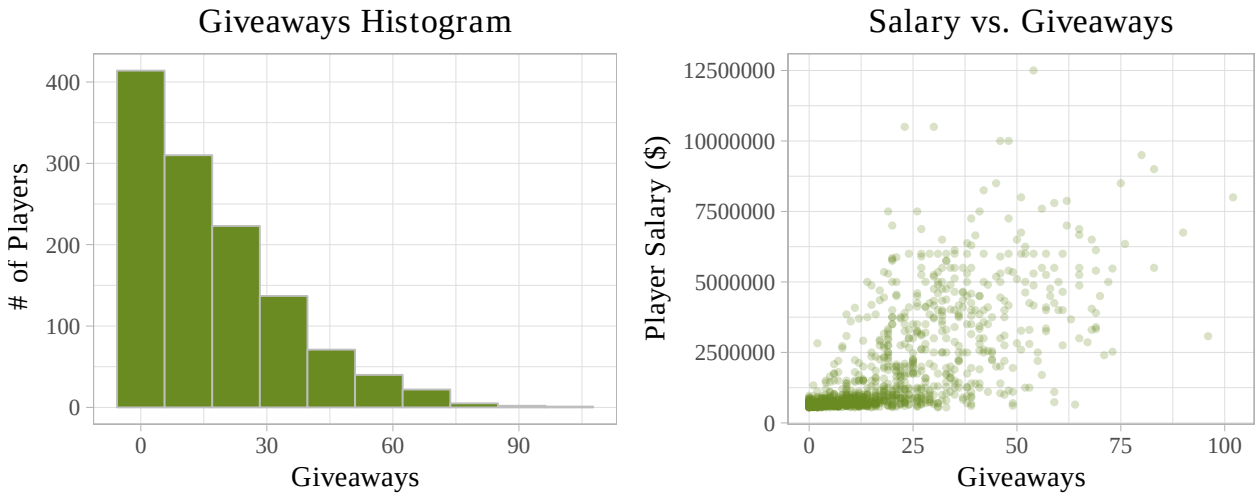


Figure 10: Descriptive Plots of Giveaways

2.2.13 Relative Corsi (REL_CORSI)

Refers to the average relative corsi, which can be positive or negative. This statistic that measures the number shot attempts on the opposing team's net (including any shot resulting in goals, misses, or blocked by the goalie) minus the number of shot attempts toward their own goal while a player is on the ice. Corsi is only calculated under circumstances in which both teams have 5 players on the ice simultaneously. Corsi indirectly provides a measure of the time a team spends in the offensive zone versus defensive zone; a positive score indicates more time in the offensive zone (as a consequence of more shot attempts on the opposing team's net) while a negative score implies the opposite.

According to the NHL, corsi is an advanced statistic intended to 'take advantage of the much larger sample size of shot attempts than goals, and to remove the influence of varying goaltending performance from a measure of skater performance' (source: www.NHL.com/glossary).

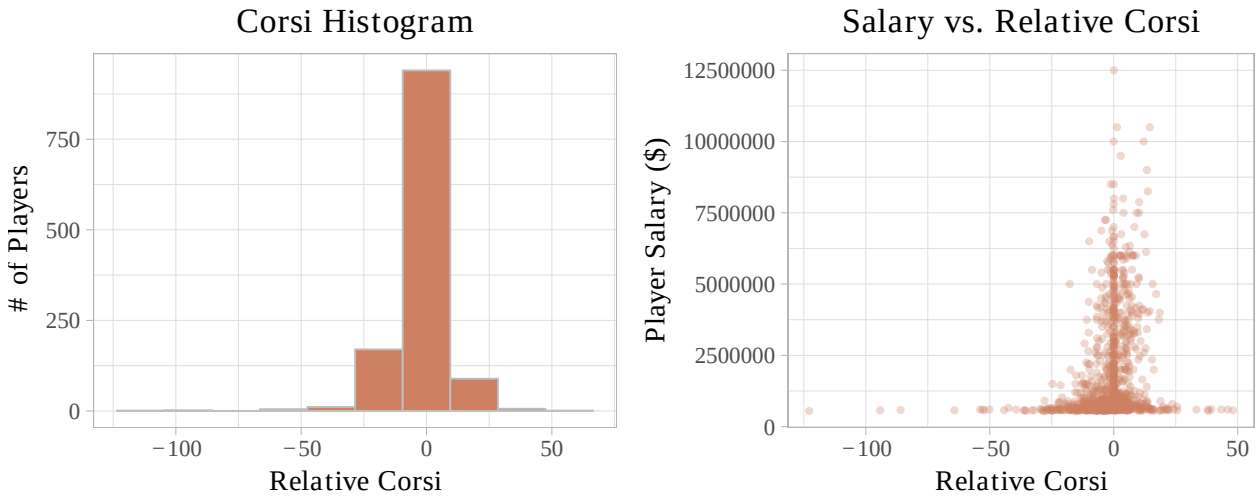


Figure 11: Descriptive Plots of Relative Corsi

2.2.14 Penalty Minutes (PIM)

Penalty minutes accumulated per minute by an individual player. Penalties occur when a player infringes the rules set by the NHL, such as unwarranted physical contact between players, fighting, or unsportsman-like behaviour. Players committing penalties are sent to the penalty box for a specified period of time depending on the severity and frequency of penalties within that game. The player in the penalty box is not replaced, resulting in their team lacking a player for the duration of their penalty time. For this reason, player accumulating large numbers of penalty minutes place their team at a disadvantage since the opposing team will have an extra player (called a *power play*).

Since incurring penalties places teammates at a disadvantage, it is undesirable for players to accumulate a large number of penalty minutes, and may be reflected via lower salary assignments. However, although penalty minutes in general are undesirable, every team has a certain number of 'grittier' players who do more of the 'dirty work', and will often take more penalties. As a result, this statistic can characterize a player with more 'grit', which are are priced differently than offensive players.

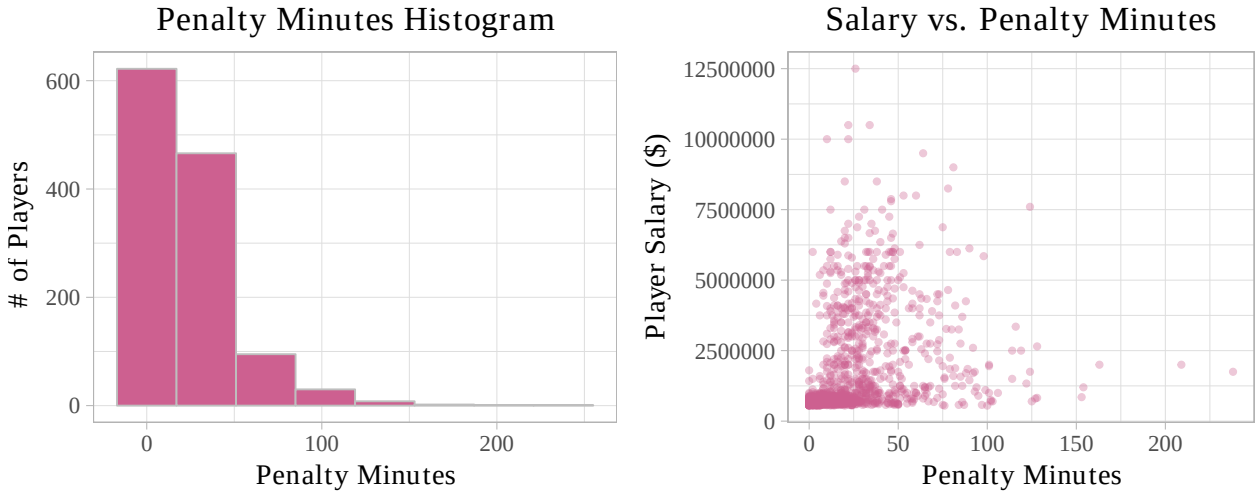


Figure 12: Descriptive Plots of Penalty Minutes

2.2.15 Hits (HITS)

The total number of body checks a player delivered to the opposing team's puck carrier (called a hit). Hits are typically used to gain possession of the puck from the opposing team. Similar to penalty minutes, hits is a measure of a player's 'grit', and grittier players will earn different salaries than offensively talented players.

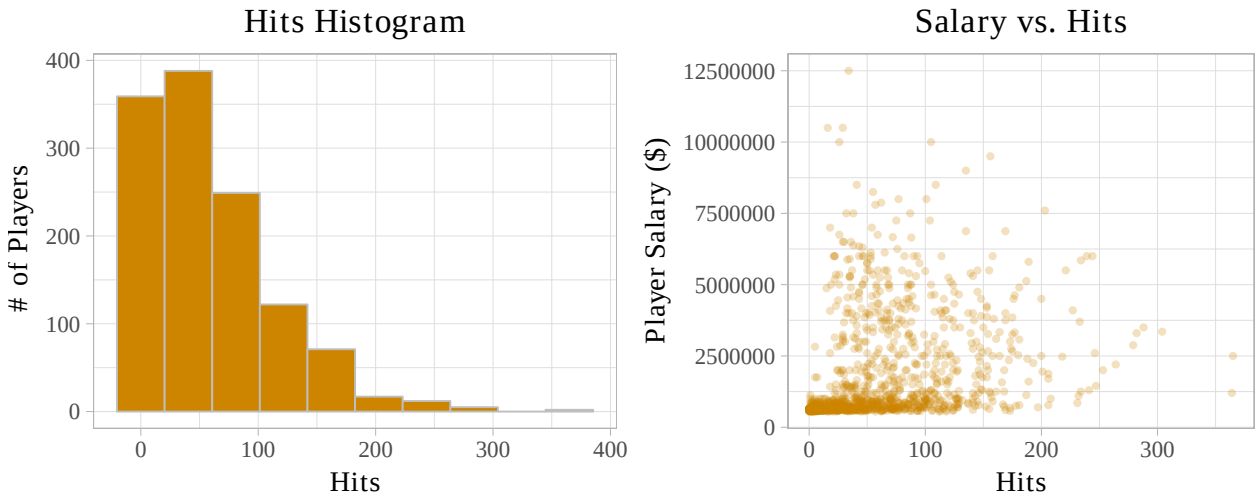


Figure 13: Descriptive Plots of Hits

2.2.16 Plus-Minus (PM)

"Plus-minus is a team's goal differential while a particular player is on the ice, excluding the power play but including empty net situations. All the skaters on the ice receive a plus or minus when an even strength goal or shorthanded goal is scored depending on which team scored; plus-minus is not tracked for goalies. However, plus-minus is heavily influenced by the strength of a player's teammates and goaltending, as well as small sample variances." (source: www.NHL.com/glossary).

A high plus-minus indicates a player is on the ice for a larger amount of goals by their team compared to the opposing team. This may reflect the quality of their skills, though there is potential for bias due to other teammates as mentioned above. It is reasonable to assume players with a high PM score may be more desirable (thus leading to higher salaries), though the extent to which this metric is considered for salary decisions may be smaller than other variables previous mentioned.

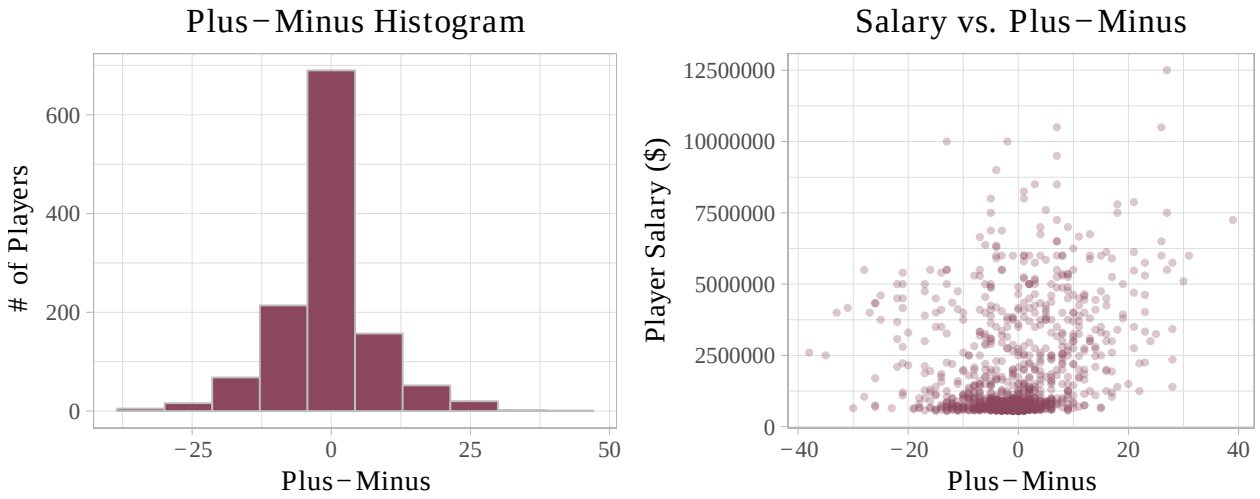


Figure 14: Descriptive Plots of Plus-Minus

3 Preprocessing

This section will discuss the preprocessing steps taken to optimize the data for the model building process. These steps include variable transformation, initial outlier handling, and splitting the data into a training and validation set.

3.1 Log Transformation of the Target Variable

First, we take the log transform of the target variable. The cap hit variable is rather skewed, as shown in Section 2.2.1, so we take the log transform to avoid heteroskedacity. The 'target variable' in this report will refer to the log transformed cap hit as opposed to the cap hit hereafter.

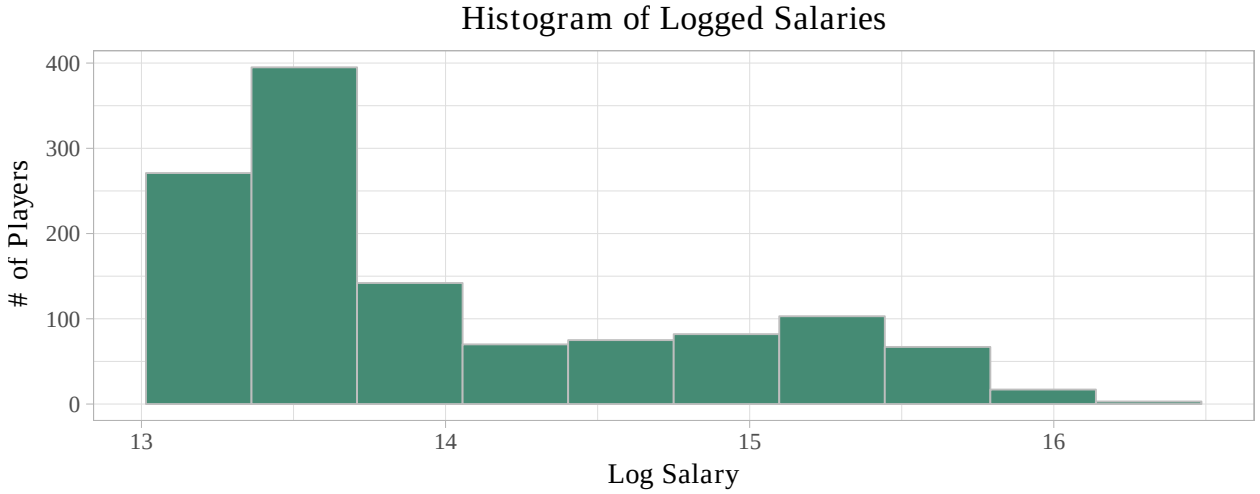


Figure 15: Descriptive Plot of Logged Salaries

3.2 Handling Categorical Variables

Second, we must handle our categorical variables: position and nationality. Nationality has 18 levels in the original dataset, while position has 12 positions and/or combinations of positions for different players. This many categories not only provides excessive granularity to our dataset, but will likely not be helpful in our analysis. As a result, we group our categorical variables. We limit the nationality to Canada, USA, and other. Position will be grouped into forward and defenseman. Then, we create dummy variables for position and nationality, which is required for some models in our analysis.

3.3 Initial Outlier Handling

Third, we do some initial outlier handling. We note that there is no (or very little) measurement error in the data. As a result, there is no reason to eliminate players for having extreme values in any category, because players with such statistics do exist, and they therefore must be accounted for in our model. However, one variable which we will look at for the purpose of our study is games played. Our goal is to predict salaries of NHL players in the next season, so we want to exclude players that play minimal games after being called up from the American Hockey League (AHL) or another minor leagues. According to the NHL website, a player is considered to have played a season if they played 25 games. As a result, we limit our analysis to players who played 25+ games in the season before signing their contract such that we are left with 813 players in the dataset. We acknowledge the possibility that this may remove some players that have major injuries during the season and miss enough games. However, this does not affect any players in our dataset.

Note that in terms of outlier removal, we also tried removing outliers using the multivariate method Cook's distance, as described in Appendix C. This was not included in our final report because the models were comparable to those presented in this paper and these 'outliers' are still NHL players who need their salaries predicted.

3.4 Training and Validation Sets

Fourth, since we will be selecting parameters and variables in the next section based on our dataset, we want to ensure we have a completely separate validation set to ensure that we don't add any bias to our predictions by only looking at cross-validation error within our dataset. As a result, we split our data into training and validation sets using an 80/20 split such that the training and validation sets have 650 and 163 players, respectively. We will henceforth do the analysis *only* on the training set, unless otherwise specified.

3.5 Correlation of Features

In Table 4, we can see that there is clear correlation between some of the variables in the training set. We note that many of these variables seem to be correlated with games played. In particular, variables which increase with a number of games tend to be correlated with games played, which makes sense intuitively. As a result, we want to do a unit conversion on these variables (P, G, TKWY, GVWY, PIM, and HITS). We use a standard conversion that is done in the literature, where we take the variable/60 minutes played (the length of a game). Note that we convert the units in both the training and the validation set.

Table 4: Correlation Between Unaltered Numerical Predictors

Var1	Var2	Correlation
G	P	0.91
TKWY	P	0.75
GVWY	TOI.GP	0.72
TKWY	G	0.69

Var1	Var2	Correlation
P	GP	0.66
TKWY	GP	0.65
GVWY	GP	0.59
G	GP	0.58
HITS	PIM	0.56
GVWY	P	0.56
GVWY	TKWY	0.51

After unit conversion, Table 5 shows the remaining correlations adjusted for time, where VARIABLE.60 refers to the variable per 60 minutes. The variables are intuitively correlated: goals and points both refer to offensively oriented players, hits and penalty minutes refer to a grittier player, and players who are "grittier" tend to play less ice time. We tried PCA between the groups of variables to remove remaining correlation, as described in Appendix C, but this will not be included in the study, as the models had worse performance.

Table 5: Correlation Between Converted Numerical Predictors

Var1	Var2	Correlation
G.60	P.60	0.86
HITS.60	PIM.60	0.62
HITS.60	TOI.GP	-0.51

3.6 Variable Standardization

The final step in our preprocessing is variable standardization so that the predictors were centered around zero and have unit variance. Standardization is an important data preprocessing step for multiple reasons. First, in methods such as GBM or XGBoost where they are optimized using gradient descent, unstandardized data may cause some weights to converge faster than others, which will make it difficult to move towards the optimal solution. Moreover, in the distance function of KNN, the algorithm assumes that features have be centered and have variance in the same order. If a feature is larger in magnitude than others, it may dominate relative to the others and lead to an unexpected result. In contrast, random forest, spline and multiple linear regression methods are unaffected by scaling. The scaling parameters are determined from only the training set and are then used to scale both the training and validation sets to avoid introducing additional information and bias into the training predictors.

4 Models

This section will compare models of different types, including multiple linear regression, smoothing splines, KNN, random forest and XGBoost. Model performance is compared using the root-mean-square-error (RMSE) as defined by:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y_i})^2}$$

Models will be compared by using both a 5-fold cross validation on the training set as well as predictions onto the validation set. By using the two datasets, we ensure that models aren't performing better on the training set than it would on a true out-of-sample player because models were tuned and features were selected based on this data. Moreover, by comparing with the cross-validated RMSE, we ensure that it isn't a 'fluke' that models are performing well on the validation set.

4.1 Multiple Linear Regression

4.1.1 About Multiple Linear Regression

Multiple linear regression provides a reasonable starting point for modelling this data. It has the distinct advantage of being intuitive and easy to understand for individuals without extensive training in statistics, which is valuable in the context of this particular data set. Moreover, it can provide an explanation for salary predictions based on player performance metrics. A major disadvantage to multiple regression is the limitation in its ability to describe the relationship between predictors and response variable. The relationship is constrained to be modelled linearly with respect to the regression coefficients, which may not reflect the true relationship. Additionally, the inclusion of high-leverage points may cause significant changes in the model fit which may be unwanted. However, in this context these effects are relatively limited as a consequence of the outlier removal outlined in the pre-processing portion of the report.

4.1.2 Model Fitting

To fit our final model, we eliminate variables that do not strongly contibute to the quality of the fit. To do this, we perform recursive feature selection starting from our full list of main effects, polynomial terms, and interaction effects, as described in Appendix A. Note that interaction terms are denoted by VARIABLE1VARIABLE2, while polynomial terms are defined as VARIABLED, with D being the degree of the polynomial. We use 5-fold cross validation, and remove the least important variables recursively, where variable importance is defined by the absolute value of the t–statistic. Figure 16 shows the cross-validated RMSE where we cumulatively drop variables, until there is only one variable left. This is represented by the right-most point on the graph, where the variable remaining is the variable that has not been dropped (i.e. is not listed in the x-axis labels).

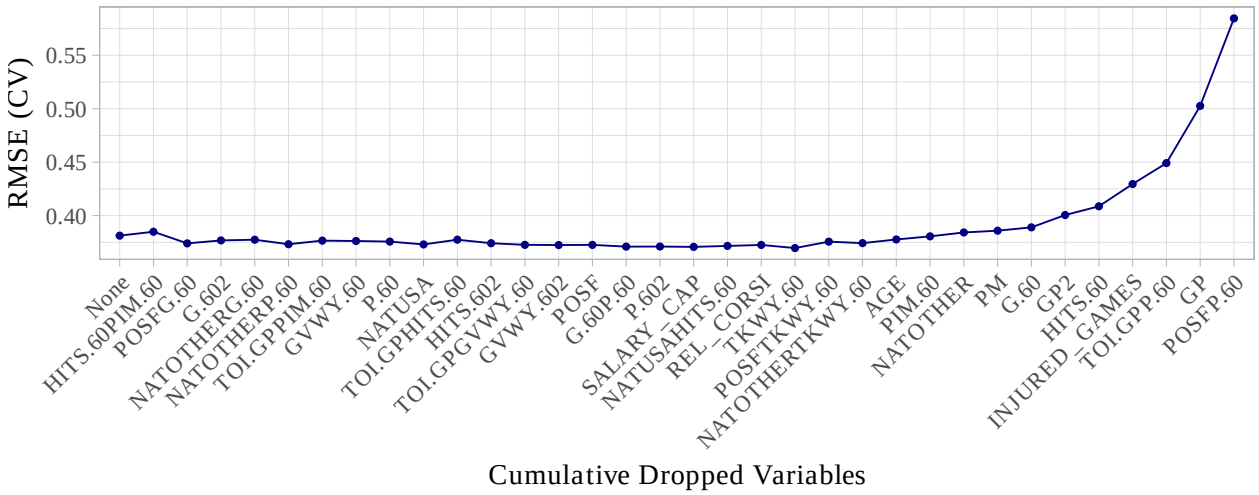


Figure 16: Recursive Feature Elimination for Multiple Regression

We note that the final fitted model will result from some trade-off between a suitably low RMSE value and model complexity, as defined by the number of parameters. The minimum RMSE occurs after dropping TKWY.60 such that the covariates are: POSFTKWY.60, NATOTHERTKWY.60, AGE, PIM.60, NATOTHER, PM, G.60, GP2, HITS.60, INJURED_GAMES, TOI.GPP.60, GP, POSFP.60, TOI.GP.

Examining the Figure 17 below, we see the importance of the remaining variables in the model. In this case, the time on ice per game, the number of games played, and the number of injured games are the most important features. That is, we can determine the most about a player's salary simply based on how much they play, relative to the games they are injured.

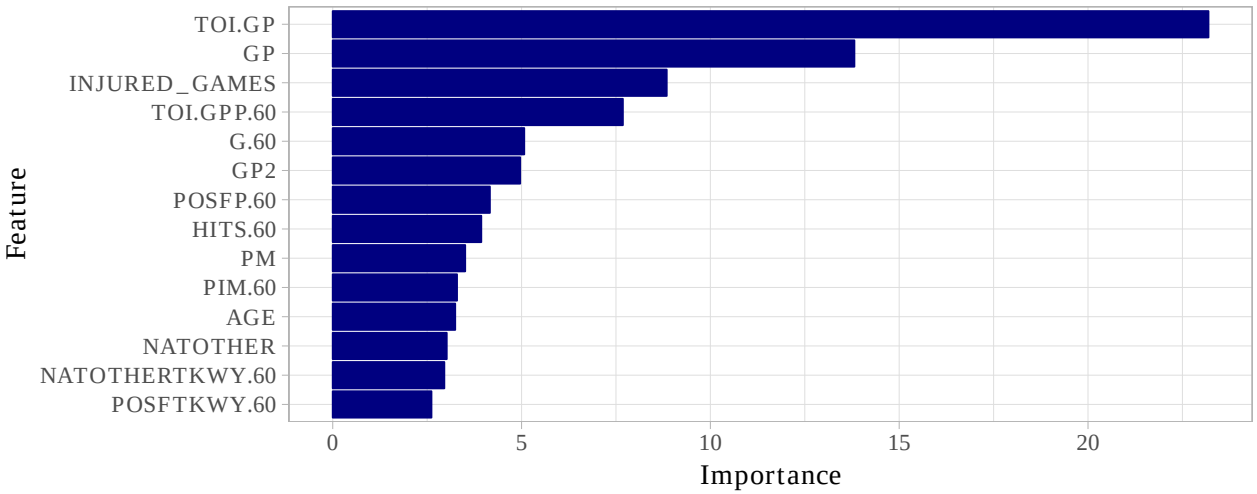


Figure 17: Feature Importance for Multiple Linear Regression

For the best model, we have a cross-validation RMSE on our training set of 0.3697 and a validation RMSE is 0.3925. Thus, our model appears to be performing worse on the validation set, but not considerably so.

Examining the residual plots in Figure 18, we note that there is not a complete normal distribution and there is a reasonably strong decreasing pattern between the residuals and fitted values, and a relatively strong increasing pattern between the residuals versus actual. Thus, there is opportunity to improve the fit of this model, particularly around the relative endpoints of the data.

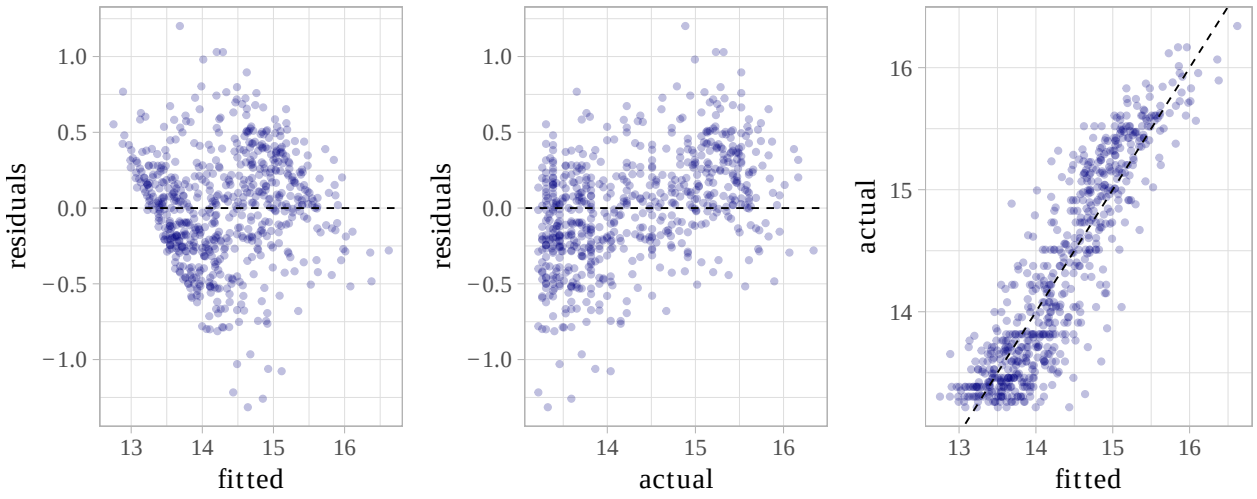


Figure 18: Diagnostic Plots for Multiple Regression

4.2 Generalized Additive Model (GAM) using Smoothing Splines

4.2.1 About GAM using Smoothing Splines

While multiple linear regression models provide a good global fit, splines are more suitable for explaining local behaviour. This is important because we expect players with similar statistics to be good predictors for each other. Smoothing splines provide local fits between knots in the data, where these knots act as boundaries that distinguish parameter estimate regions. The optimal solution to the penalized

least-squared problem for smoothing splines is a natural cubic spline with knots at each unique data point. Smoothing splines provide a similar fit to a LOESS model with the appropriate degrees of freedom, with the exception of the endpoints being linear in a smoothing spline. As a result, only smoothing splines are considered in our report. The complexity of the fit produced by such models can be controlled using the degrees of freedom parameter. The higher the degrees of freedom, the more complex the model, but also the more bias the estimates become. For lower degrees of freedom, the model is smoother and less bias but has a higher variance. This is an example of the typical bias-variance tradeoff and care must be taken to select an appropriate balance by tuning the number of effective degrees of freedom.

4.2.2 Model Fitting

We perform model selection on the GAM with splines by recursively removing the least imortant variable or interaction in order to simplify the model while also choosing the optimal degrees of freedom for the model using a 5-fold cross validated grid search. We define importance for the GAM with splines as the p-value for nonparametric F-tests for continous variables, and parametric F-tests for categorical data where we are testing the importance of each additive model compared to the null hypothesis. In doing so, we will be able to select the best model such that we have accounted for the bias-variance tradeoff. The RMSE calculated for the model with the optimal parameters using 5-fold cross validation will be displayed in Figure 19 for each incremental model in order to illustrate our selection based on variable significances.

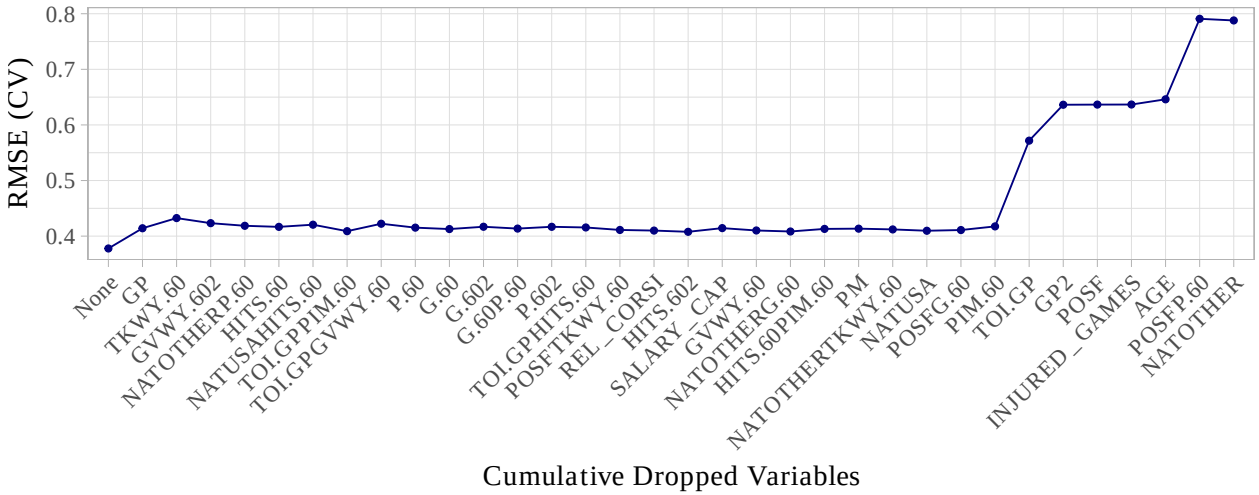


Figure 19: Recursive Feature Elimination for Splines

We note a peculiar trend in the cumulative feature selection relative to other models. It appears the removal of a single covariate results in a considerable increase in the RMSE. However removing additional covariates from this point onward produces a similar trend relative to previous models. Thus, removing any covariates from the model would result in less accurate predictive power, and so our final model remains highly complex since it contains all initial covariates.

Examining the Figure 20, we see the optimal degrees of freedom for the spline as well as the importance of the remaining variables in the model. In this case, the nationality of a player is by far the most important feature to include within the model. Time on ice per game played is also an important feature, while the interaction between time on ice per game and points per 60 minutes is the most significant interaction. Furthermore, we see that selecting the degree of freedom of the spline to be 2 results in the minimal RMSE for these selected parameters.

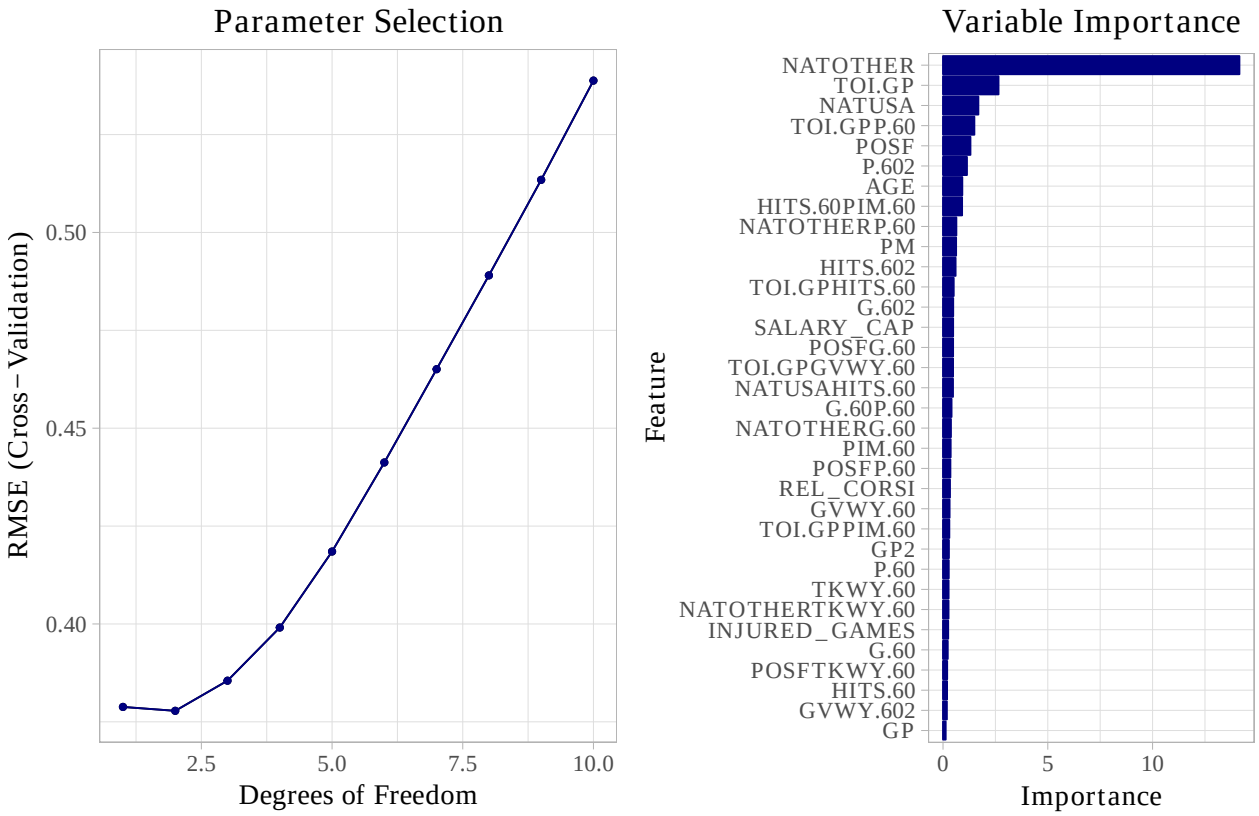


Figure 20: Model Fitting for Splines

For the best model, we have a cross-validation RMSE on our training set of 0.3778 and a validation RMSE is 0.3834. Thus, similar to before, our model appears to be performing only marginally worse on the validation set.

Examining the residual plots below in Figure 21, we note a similar trend to the residual plots produced by the multiple linear regression model. There appears to be linear trends within both the residuals versus fitted, and residuals versus actual plots. This indicates, in particular, that we may have issues with fitting the endpoints of the data once more.

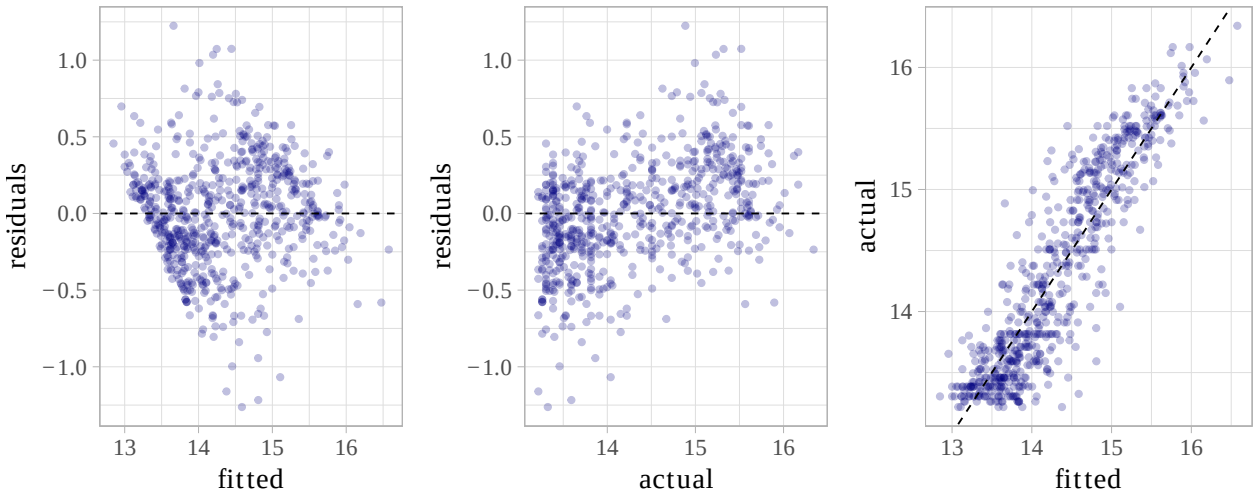


Figure 21: Diagnostic Plots for Splines

4.3 K-Nearest Neighbours (KNN) Regression

4.3.1 About KNN

KNN regression is a simple algorithm that is easy to understand: players' salaries would be predicted based on other players who have "similar" statistics to them, which makes it a sensible choice for this data. KNN does, however, have some disadvantages. In particular, KNN can only predict within the range of the training data. Moreover, KNN can be quite computationally expensive due to the fact that you must find the distance between all neighbours. You must also store the entire dataset for finding distances with neighbours during predictions. For the purpose of this study, the dataset is relatively small so it will not be greatly affected. Finally, KNN is sensitive to both irrelevant features, since all features contribute to the similarity, as well as the number of features, since KNN is subject to the curse of dimensionality. As a result, feature selection will be an important factor for the performance of KNN. KNN is also sensitive to the value its parameters, namely the number of neighbours, k. In particular, when k is low, KNN can have a flexible fit with high variance and low bias, while a high k would provide a smoother fit with low variance and high bias that is more robust to outliers. Thus, we must ensure to finely tune this parameter.

4.3.2 Model Fitting

Given the fact that the model is sensitive to feature selection and there is a bias-variance tradeoff with the number of neighbours, we perform recursive feature selection use 5-fold cross validation. That is, we train models and remove the least "important" variable. We define importance for KNN as the loess r-squared variable importance. For this metric, a loess smoother is fit between the target variable and the predictor, and then the R^2 is calculated for this model against the null model with only the intercept. For each of the models that we train, we perform a 5-fold cross-validated grid search to find the optimal parameters, which in this case is the number of neighbours, k. More details on the grid search will be discussed for the final model for KNN. In Figure 22, we see the succession of RMSE based on cumulatively dropped variables.

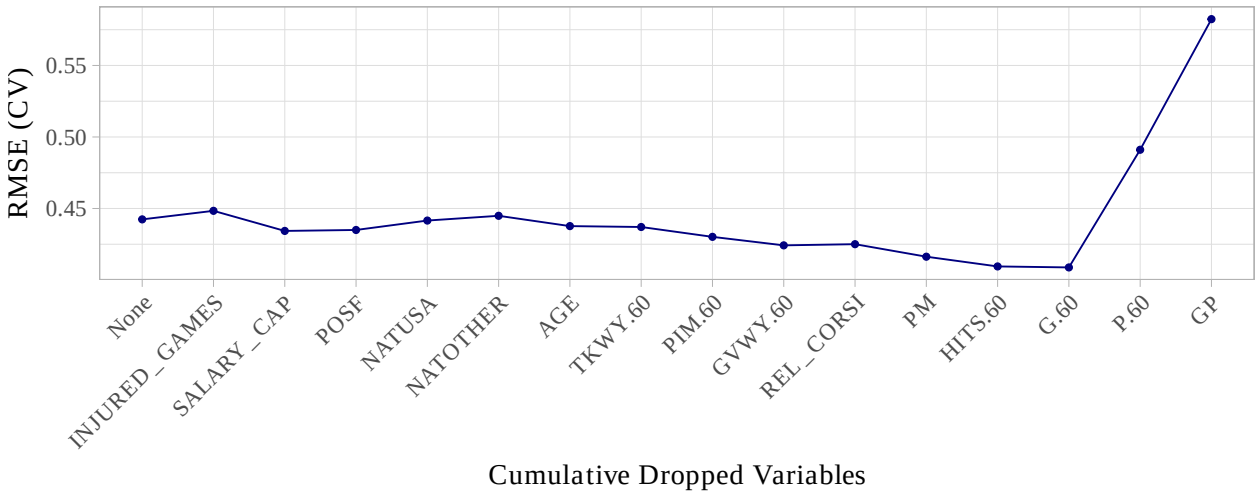


Figure 22: Recursive Feature Elimination for KNN

Thus, we can see in Figure 22 that we achieve the best model when we include the following variables: P.60, GP, TOI.GP. Note that we do not consider the validation error during parameter tuning and variable selection to keep this as a completely out-of-sample set of observations.

We can now examine the best model. First, we look at the optimal parameters, which can be found in Figure 23. We can see from the graph that the optimal model has k = 30 for the grid search composed of k = 5, 10, 15, 20, 25, 30, 35, 40, 45, 50. We also note the importance of the variables. Moreover, we can see that time on ice is the most important variable, followed by games played and points per 60 minutes.

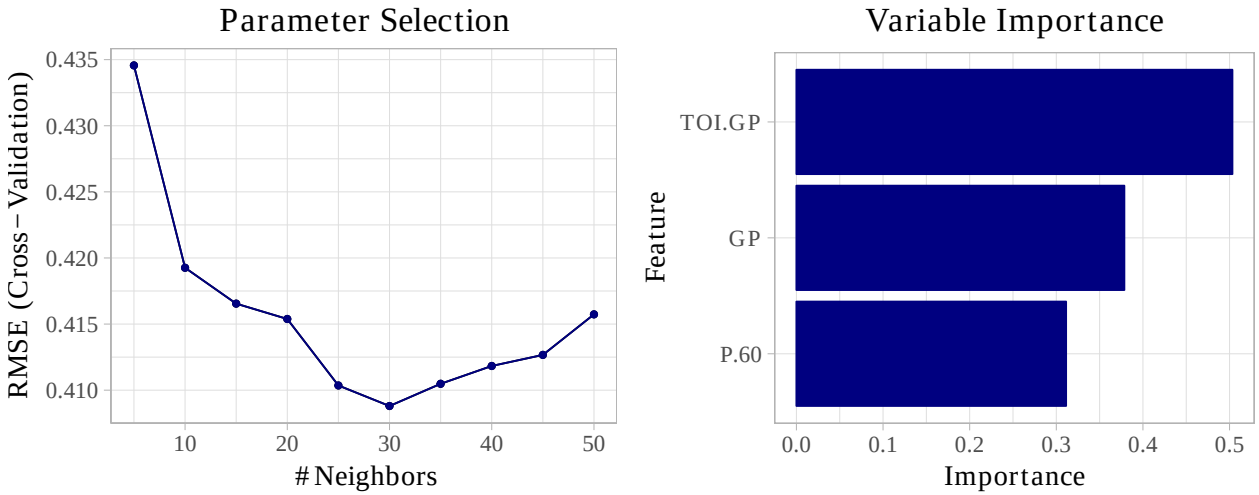


Figure 23: Model Fitting for KNN

For the best model, we have a cross-validation RMSE on our training set of 0.4088 and a validation RMSE is 0.3535. We can see that there is a *clear* discrepancy between the two values. This is indicates that performance on the validation set may not be indicative of the overall performance.

Furthermore, examining the diagnostic plots in Figure 24, we can see that the residuals are not completely normally distributed. In particular, we notice a slight decreasing trend when we look at the residuals vs. fitted. In contrast, we notice an increasing trend when we compare the residuals to actual. That is, we overestimate low salaries and underestimate high salaries. This makes some intuitive sense, because we

know that KNN can only fit within the range of the training data since it will take an average of the nearest neighbours. As a result, there is room to improve upon this model.

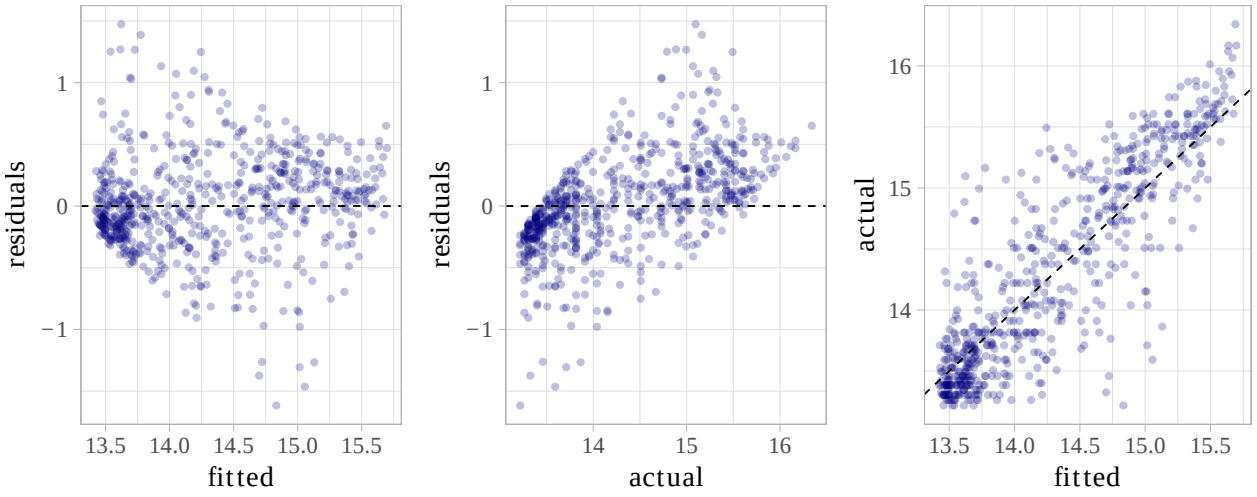


Figure 24: Diagnostic Plots for KNN

4.4 Random Forest

4.4.1 About Random Forest

Random forest is a popular and easy-to-use algorithm that is known for producing accurate predictions. Unlike KNN, the random forest is not as sensitive to irrelevant features because it subsamples the features during bootstrapping and it is not sensitive to the scaling of the data. However, similar to KNN, they can be quite complex. As the forest grows, it the computing time and memory requirements will increase for both training and predictions. Adding this complexity comes in different forms. For the purpose of this report, we consider the number of variables randomly sampled as candidates at each split (mtry), the number of trees (ntree) and the maximum number of nodes in each tree (maxnodes). As with KNN, the random forest is subject to the bias-variance tradeoff as the complexity increases (i.e. as mtry, maxnodes and ntree increase).

4.4.2 Model Fitting

Although the random forest is not overly sensitive to inaccurate features or parameter tuning, it can greatly affect the computational complexity of the model. As a result, we perform the same procedure that we did with KNN. We train the models and remove the least important variable, where we define importance for the random forest as the increase in mean-square-error of the predictions estimated with out-of-bag cross validation as a result of the variable in question having its values permuted (randomly shuffled). We perform a 5-fold cross-validated grid search to find the optimal parameters for each set of features. Figure 22 shows the cross-validated RMSE where we cumulatively drop variables.

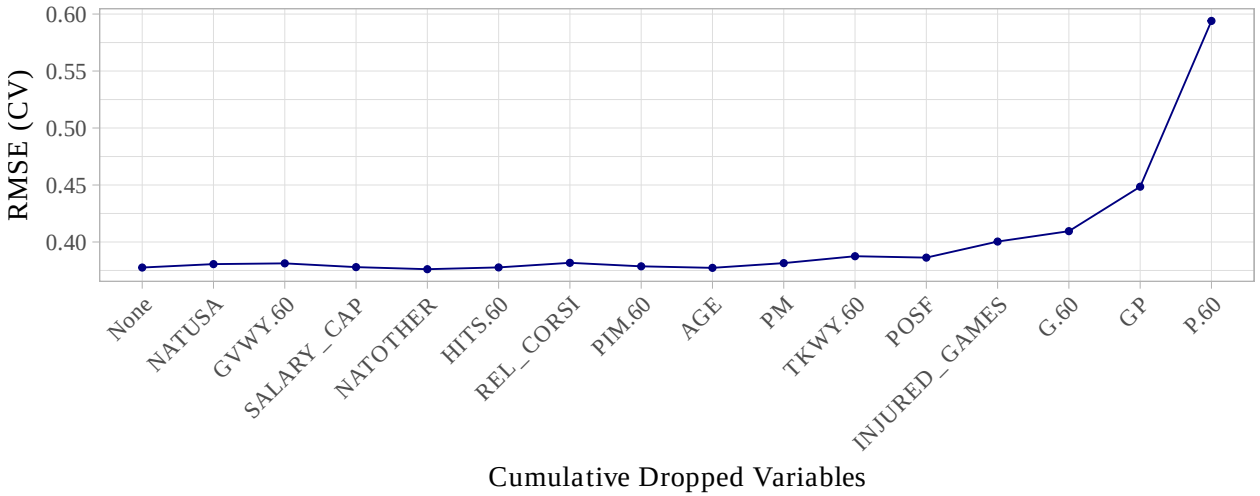


Figure 25: Recursive Feature Elimination for Random Forests

As expected, we can see that the model is much less sensitive to variable removal than KNN, as the RMSE does not decrease when we begin removing variables. In Figure 25, we can see that we achieve the lowest RMSE when we include the following variables: HITS.60, REL_CORSI, PIM.60, AGE, PM, TKWY.60, POSF, INJURED_GAMES, G.60, GP, P.60, TOI.GP. We note that we could remove more variables and achieve a similar RMSE (and also reduce complexity based on the number of variables). However, the best model for those variables had more computational complexity in terms of its parameters. As a result, we keep the best model as determined by the cross-validated RMSE.

In Figure 26, we can see the results of the parameter optimization, where the numbers in the grey sections of the plot refer to the indicate the number of trees, ntree. Here, we can see that the optimal variables had an mtry of 6, 100 trees, and a maximum of 250 nodes per tree.

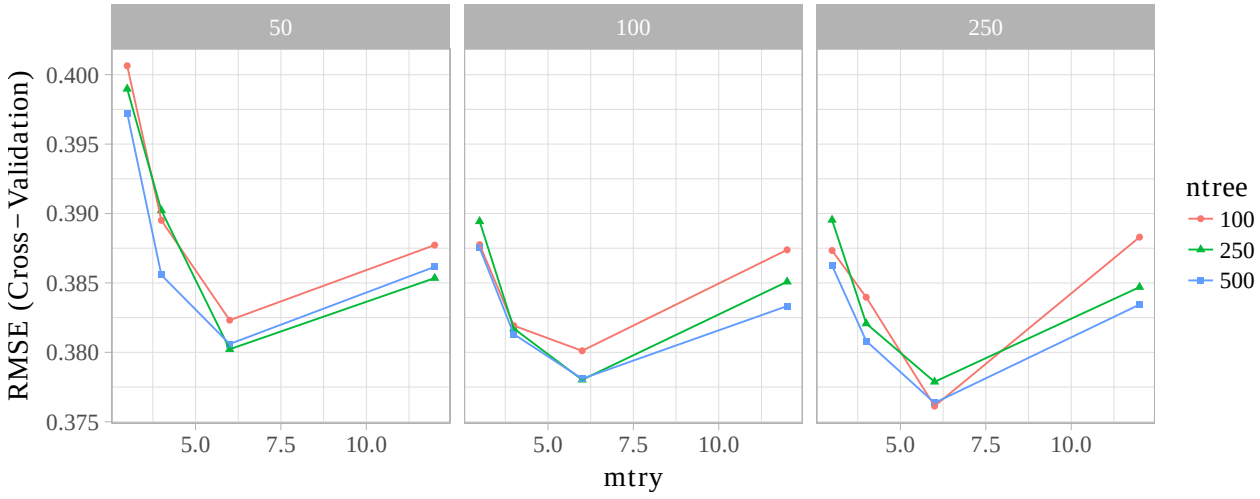


Figure 26: Parameter Optimization for Random Forest

Figure 27 shows the variable importance. We note the three most important variables are the same as those from the final KNN model.

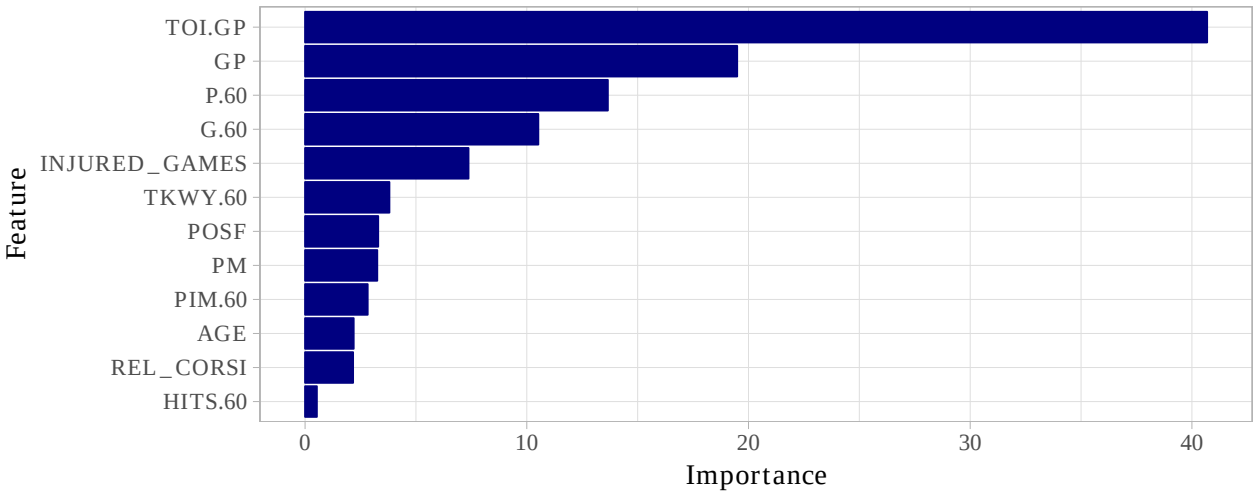


Figure 27: Feature Importance for Random Forest

This model has a cross-validation RMSE on our training set of 0.3761, and a validation RMSE is 0.374. Both of the RMSEs are close, so there is no evidence that the model is outperforming on either set.

Finally, the diagnostic plots in Figure 28 shows that the residuals are fairly normally distributed when looking at the residuals vs. fitted. There is, however, still a slight increasing trend when we look at actual salaries vs. their residuals, as in .

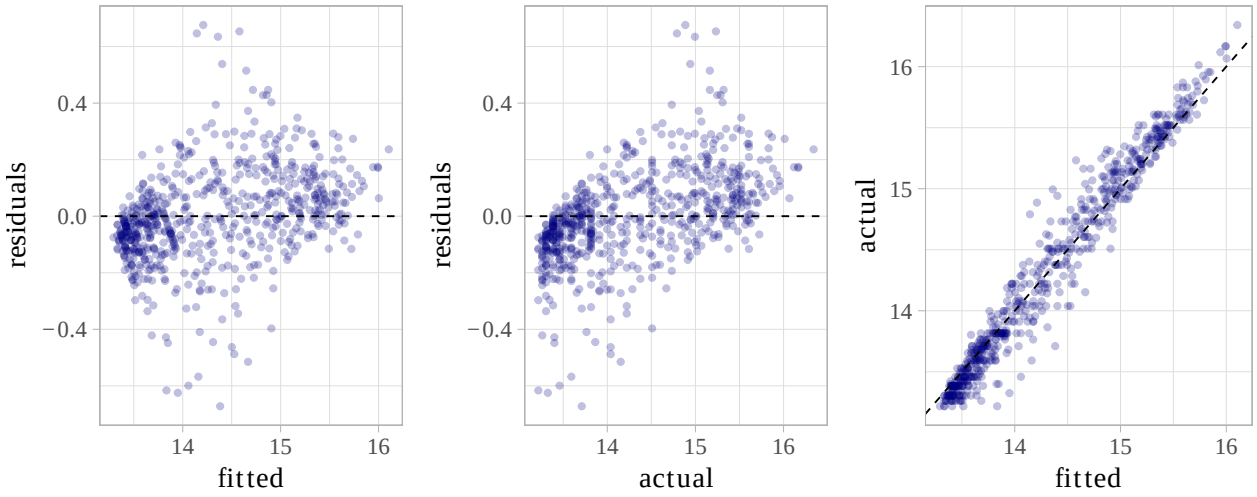


Figure 28: Diagnostic Plots for Random Forest

4.5 Extreme Gradiant Boosting (XGBoost)

4.5.1 About XGBoost

Random forest was an example of bagging trees, while XGBoost is an example of boosting trees. Boosting has the advantage over bagging (which is used for random forests) because not only will it reduce variance, but it may also reduce bias by putting weight on difficult to predict points. However, if a single model is overfitting, then bagging trees may be the better choice. As a result, both models have their pros and cons, and we will evaluate their performance on our dataset.

XGBoost, in particular, is known for two reasons: it's computationally efficient and has excellent modern performance. This is why we have selected this model for our paper. XGBoost has proven to have excellent accuracy by being the algorithm used by number Kaggle competitions. Moreover, it is faster than a similar tree-boosting algiorithm, gradient boosting models (GBM), and has additional regularization constraints to avoid overfitting. As a result, we have decided to only include GBM in Appendix C, since the model has similar results, but is a slower and generally less accurate algorithm.

4.5.2 Model Fitting

Similar to the random forest, we select features and parameters largely to improve the computational complexity of the model. Following the same methodology, we remove variables based on importance, where importance is the same as the random forest, except it computes the measure using the entire training dataset (not the just out-of-bag observations). Similarly, we perform a 5-fold cross-validated grid search to find the optimal parameters for each set of features. The parameters that are tuned are the number of rounds you go through the data (nrounds), the maximum tree depth (max_depth), the learning rate (eta), and the fraction of observations that are subsampled to grow trees (subsample). Figure 29 shows the cross-validated RMSE where we cumulatively drop variables.

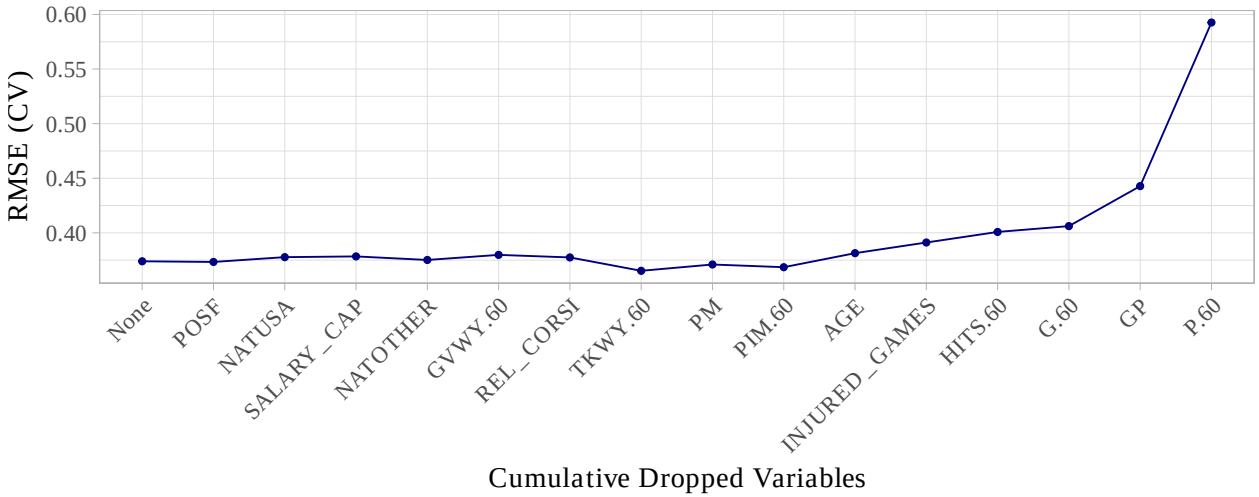


Figure 29: Recursive Feature Elimination for XGBoost

In Figure 29 we can see that the model is not particularly sensitive to having to many features, similar to the random forest. We achieve the lowest RMSE when we include the following variables: PM, PIM.60, AGE, INJURED_GAMES, HITS.60, G.60, GP, P.60, TOI.GP. As in our model selection for the random forest, we could remove additional variables from the "optimal" model such that the cross-validated RMSE stays approximately the same, but there are fewer features. However, for the same reasons, we will keep the best-selected model.

In Figure 30, we can see the results of the parameter optimization, where the numbers in the grey sections on the top of the plot refer to the indicate subsample percentage, and those on the side refer to the learning rate, eta. Here, we can see that the optimal variables are 500 rounds, trees with a maximum depth of 10, a learning rate of 0.05 and observation subsampling of 50%.

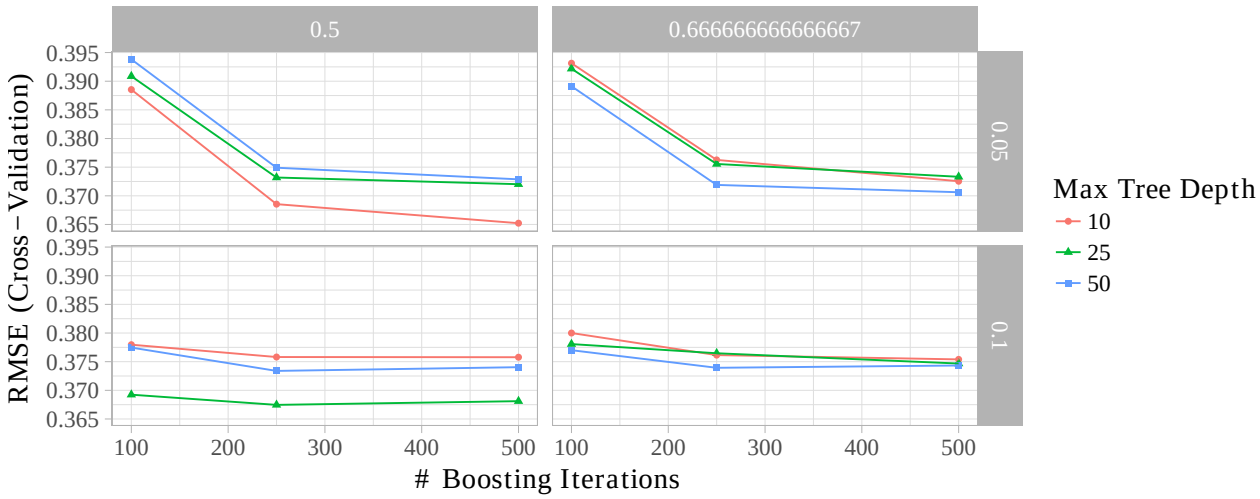


Figure 30: Parameter Optimization for XGBoost

Figure 31 shows the variable importance. We note the three most important variables are yet again the same as the previous two models.

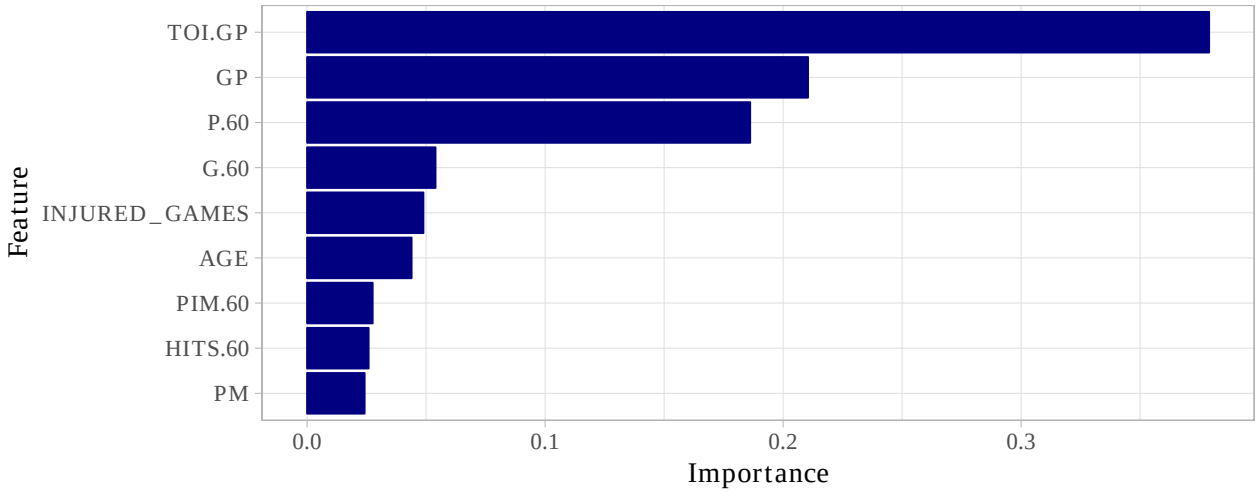


Figure 31: Feature Importance for XGBoost

This model has a cross-validation RMSE on our training set of 0.3652, and a validation RMSE is 0.3701. Both of the RMSEs are close, so there is no evidence that the model is outperforming on either set. These values are very similar to the errors of the random forest.

Lastly, the diagnostic plots in Figure 32 show that the residuals are normally distributed when looking at the residuals vs. fitted. There is, however, still a slight increasing trend when we look at actual salaries vs. their residuals.

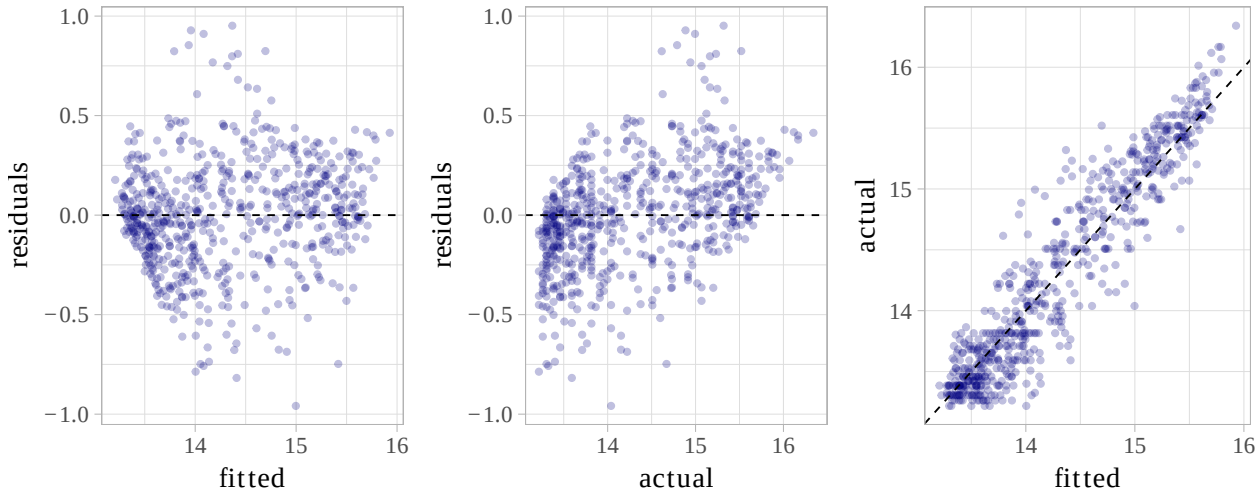


Figure 32: Diagnostic Plots for XGBoost

5 Statistical Conclusions

For determining our best model, we use multiple criteria. First, we look at both the cross-validated and validation RMSEs to compare the accuracy and stability of predictions. Next, we look to measure computational complexity of the models through two measures: time to train the model as well as the memory taken up during training. For this assessment, we have ignored the amount of time for prediction as they were all relatively similar. Finally, we determine whether or not the results are interpretable. That is, whether or not we can explain to a player the reason behind their market value and why they are being offered a specific contract. This is something that managers often have to do when entering negotiations for a new contract with a player whose existing contract is about to expire.

Based on our results in Table 6, we can see that many of the models have similar results in terms of errors. We note that KNN is the worst model in terms of prediction accuracy because of its inconsistency between the cross-validation and validation error, which indicates the instability of this model. Multiple linear regression has a similar issue, although the magnitude of this error was less. This leaves us with three remaining models: the spline, random forest, and XGBoost models. We note that despite being interpretable, the spline has the worst error of the remaining three methods. It also has the longest training time. As a result, this model is not selected. Between the XGBoost and random forest models, XGBoost has smaller errors, while random forest has reasonably similar errors with about half the computation time and memory than that of the XGBoost model. Although we take training time and memory into account, it is not of the utmost importance for this problem unless there is a considerable difference because of the fact that we have a small dataset and predictions are not needed within a short time period for this problem. We note that XGBoost also had the least indication of patterns within the residual plots shown in the sections above. As a result, we select XGBoost as our final model.

Table 6: Model Comparison

Model	CV Error	Val. Error	Train Time*	Train Memory*	Interpretable?
Multiple Linear Regression	0.3697	0.3925	0.751	5.19 MB	Yes
Spline	0.3778	0.3834	8.188	2.15 MB	Yes
KNN	0.4088	0.3535	0.785	0.17 MB	No
Random Forest	0.3761	0.374	1.707	1.8 MB	No
XGBoost	0.3652	0.3701	4.796	3.59 MB	No

* Note: We refer to train time/memory as the time/memory to compute 5-fold cross validation of the final model of each type (i.e. using the optimal parameters and features selected in the previous sections).

6 Conclusions

For our final selected model, XGBoost, we note that the most important variables are time on ice (per game played), games played, and points per 60 minutes. The variables contributed most heavily to the predictions in the XGBoost model. From previous knowledge, we can intuitively explain why these variables are of the highest importance. The best players are those who are most likely to get the most ice time. Additionally, barring any unforeseen injuries, players who are worth the most will be playing in as many games as possible. Coaches want their best players on the ice, and so how many games a player is playing in an accurate indication of how good the player is and how much money they are making. Lastly, in order to win games, teams must score goals. This is highly intuitive then that scoring more goals leads to increased player salaries.

Team officials such as head coachers, general managers, and owners must then consider these different factors when negotiating player salaries and justifying why they have offered their players the contracts that they have. Teams who do not understand the key features that determine a players value risk overpaying their players, as we can see in Table 7. Conversely, when players do not negotiate for the prices that they are worth, we see underpaid players as in Table 8.

Table 7: Top 10 Overpaid New Signings

Name	Signing Date	Actual Salary ($)	Expected Salary ($)	Difference ($)
Patrick Kane	2014-07-09	10,500,000	5,830,027	4,669,973
Jack Eichel	2017-10-03	10,000,000	5,371,872	4,628,128
Anze Kopitar	2016-01-16	10,000,000	5,526,206	4,473,794
Connor McDavid	2017-07-05	12,500,000	8,036,205	4,463,795
Jonathan Toews	2014-07-09	10,500,000	6,757,077	3,742,923
P.K. Subban	2014-08-02	9,000,000	5,688,168	3,311,832
Nick Leddy	2015-02-24	5,500,000	2,243,607	3,256,393
Nikita Nikitin	2014-06-26	4,500,000	1,510,037	2,989,963
Ryan O'Reilly	2015-07-03	7,500,000	4,574,893	2,925,107
Andrej Meszaros	2014-07-01	4,125,000	1,222,020	2,902,980

Table 8: Top 10 Underpaid New Signings

Name	Signing Date	Actual Salary ($)	Expected Salary ($)	Difference ($)
Brenden Dillon	2014-10-05	1,250,000	4,218,176	-2,968,176
Jaden Schwartz	2014-10-05	2,350,000	5,236,919	-2,886,919
Andre Benoit	2014-07-23	800,000	3,510,232	-2,710,232
Christian Ehrhoff	2015-08-23	1,500,000	3,902,023	-2,402,023
Chris Butler	2014-07-16	650,000	2,951,042	-2,301,042
Jamie McBain	2014-11-13	550,000	2,582,496	-2,032,496
Danny DeKeyser	2014-09-20	2,187,500	4,197,001	-2,009,501
Mats Zuccarello	2014-07-22	3,500,000	5,471,672	-1,971,672
Ryan Johansen	2014-10-06	4,000,000	5,928,154	-1,928,154
Mark Stone	2015-06-25	3,500,000	5,373,568	-1,873,568

It is important to also note that, in these tables, we notice a trend. Firstly, the players who are

considered, by our model, to be overpaid tend to be team captains or veteran players. This leads to the conclusion that there are features about these players which are not being considered by our model, such as players in leadership roles being paid more for this reason. There are features that increase player value which can not be accounted for my traditional hockey statistics. Additionally, we notice that some of the underpaid players are those who play a large portion of every season for teams in the American Hockey League (AHL), the level below the NHL. Since players are able to play in the AHL and be called up to play in the NHL as needed by the team, we have players with 'two-way contracts'. Contracts such as these can be difficult to explain since players be good in the AHL and perform well when called to the NHL. However, their contracts are valued lower since they are not expected to play many games in the NHL, and the AHL (where they predominantly play) has much lower salaries.

7 Future Work

Given that all of our models provided similar RMSE, even those that are known for their accuracy, such as XGBoost, we believe that using other statistical or machine learning methods would not likely provide substanital improvements to our final model. However, if more model fine-tuning was desired, there are additional machine learning techniques that could be considered for such problems. Methods such as neural networks or support vector regression are good at capturing non-linear trends. Neural networks may also be useful in the event that we expand our dataset to include a time component where statistics go back multiple years before the contract signing instead of simply using the season before.

One aspect of the models presented in this report that could be more significantly improved upon, however, is the data. Some variables that could have been included are elements like previous cap hit. However, as we mention in Appendix A, this data was not readily available and would require scraping each individual salary. Upon including this variable, we would need additional variables related to the previous salary, such as the contract type, the length of the contract, etc. This was one variable that was heavily relied upon in one study in our literature review. We also came across other variables which were important in their reviews that we did not have access to from our data sources. These variables, such as grit (a measure of the player's toughness calculated by SportsNet) and physical characteristics, were inconsistnet in our data source. This illustrates our largest obstacle to predictive capability in the compliation of the data. Moreover, from our analysis, we note that there is difficult to predict the extreme salaries. Including variables such as a two-way contract indictor to reveal some lower-end signings or variables that can take into account leadership and 'superstar' qualities, such as whether a player was a captain, could help our models predict these exteme salaries better.

There are also potential benefits of introducing alternate strategies in the model feature selection, outside of the methods included in our analysis. For example, 10-fold cross validation may be used instead of 5-fold, which may reduce some anomalies in predictive error on the training and validation set (such as those observed for models in which the validation RMSE was considerably lower) as a consequence of sampling. Other simpler feature selection methods could also be considered, such as backwards or forward selection for multiple linear regression.

Additionally, further investigation of potential interactions between covariates could be considered. For simplicity and brevity, our analysis only considered two-way interactions, though literature may suggest potential higher order terms of interest. Furthermore, there may be underlying relationships between covariates resulting in interactions that we have not considered within Appendix A.

8 Contributions

8.1 Data

- Collection/Cleaning/Preprocessing: Galen

8.2 Report

- Introduction: Marcus
- Literature Review: Alessandra
- Data: Marcus & Galen
- Models:
  - Multiple Linear Regression: Marcus & Alessandra
  - Smoothing Splines: Marcus & Alessandra
  - KNN: Galen
  - Random Forest: Galen
  - XGBoost: Galen
- Statistical Conclusions: Galen
- Conclusions: Galen and Alessandra
- Future Work: Galen, Marcus, and Alessandra
- Other Work in Appendix: Galen, Marcus, and Alessandra

8.3 Presentation

- Slides: Galen, Marcus, and Alessandra

Appendix A - Data

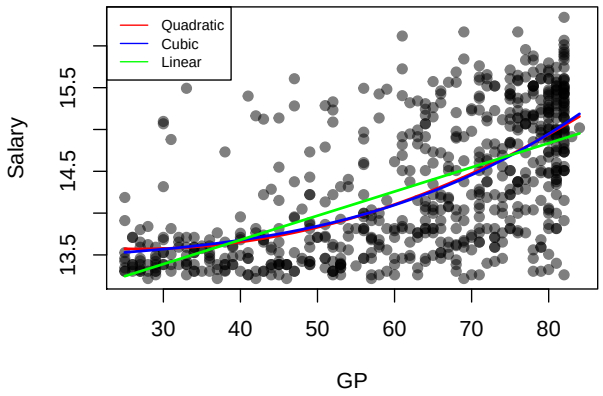
8.4 Excluded Variables

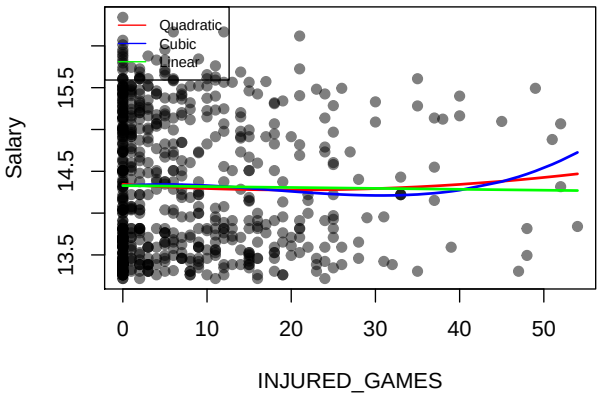
One important variable that was excluded for the purpose of our analysis was previous salary. This has proved to be important in the literature review in Appendix B. However, this data was not available in the same format for each year in the data sources we were using. Different forms of "salary" are reported each year. Since it would be necessary to individually scrape each salary off of another site for this statistic, we have eliminated it from our study. As mentioned in our recommendations for future work, we believe that this variable could help with the analysis.

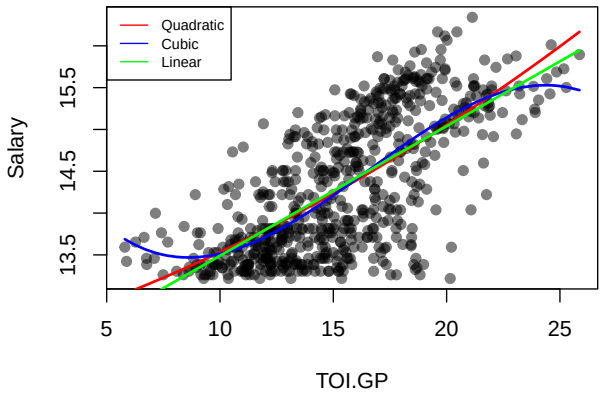
8.5 Polynomial Variable Selection

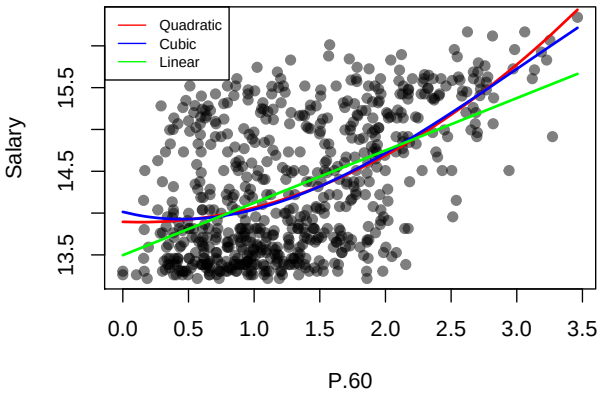
To determine the suitability of including particular power terms in the spline and regression models, we examine different polynomial regression models between the numerical covariates and the response variable, salary. If the inclusion of a particular power term is statistically significant, and the fit of the model on the data is reasonable based on our understanding of the data set, it is sensible that this power term should be included as an initial covariate during modelling. In the case where the cubic model has the cubic term as insignificant, or the model fit is poor, we consider the quadratic model. We apply the same process between quadratic and linear to select polynomial terms as a consequence of both statistical significance and intuition.

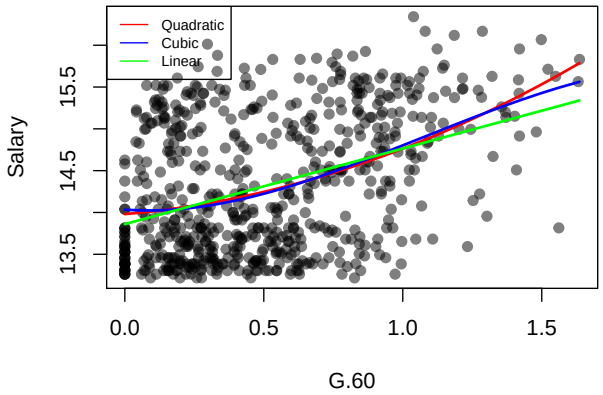
Note that when variable selection is applied to the models, these terms may be excluded as they are deemed insignificant in the context of other variables in the model.

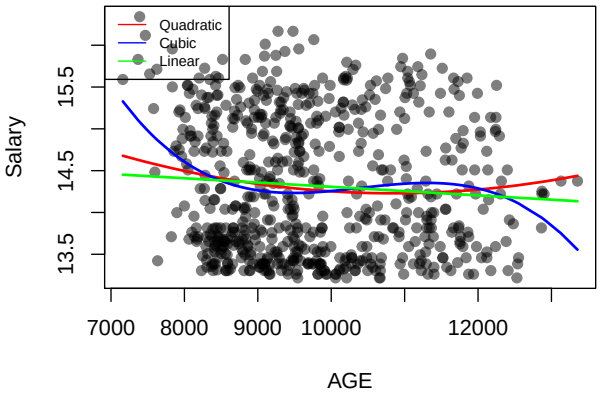

Quadratic
Cubic
Linear
Salary
13.5
14.5
15.5
30
40
50
60
70
80
GP

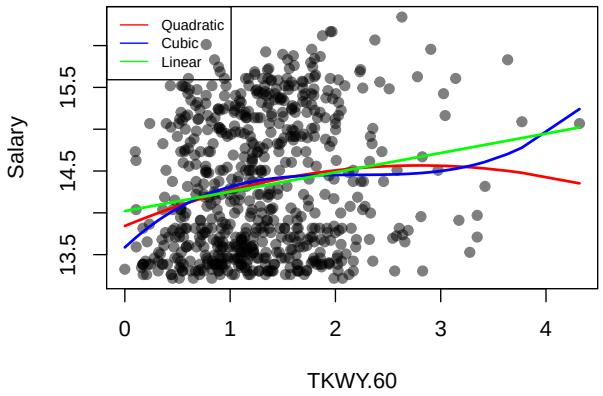

Quadratic
Cubic
Linear
Salary
13.5
14.5
15.5
0
10
20
30
40
50
INJURED_GAMES

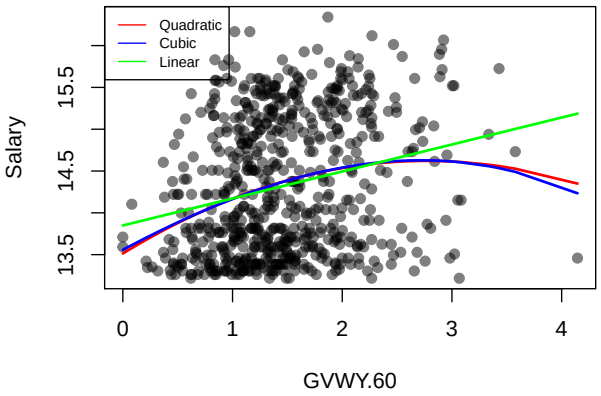

Quadratic
Cubic
Linear
Salary
13.5
14.5
15.5
5
10
15
20
25
TOI.GP


Quadratic
Cubic
Linear
Salary
13.5
14.5
15.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
P.60


Quadratic
Cubic
Linear
Salary
13.5
14.5
15.5
0.0
0.5
1.0
1.5
G.60


Quadratic
Cubic
Linear
Salary
13.5
14.5
15.5
7000
8000
9000
10000
12000
AGE


Quadratic
Cubic
Linear
Salary
13.5
14.5
15.5
0
1
2
3
4
TKWY.60


Quadratic
Cubic
Linear
Salary
13.5
14.5
15.5
0
1
2
3
4
GVWY.60

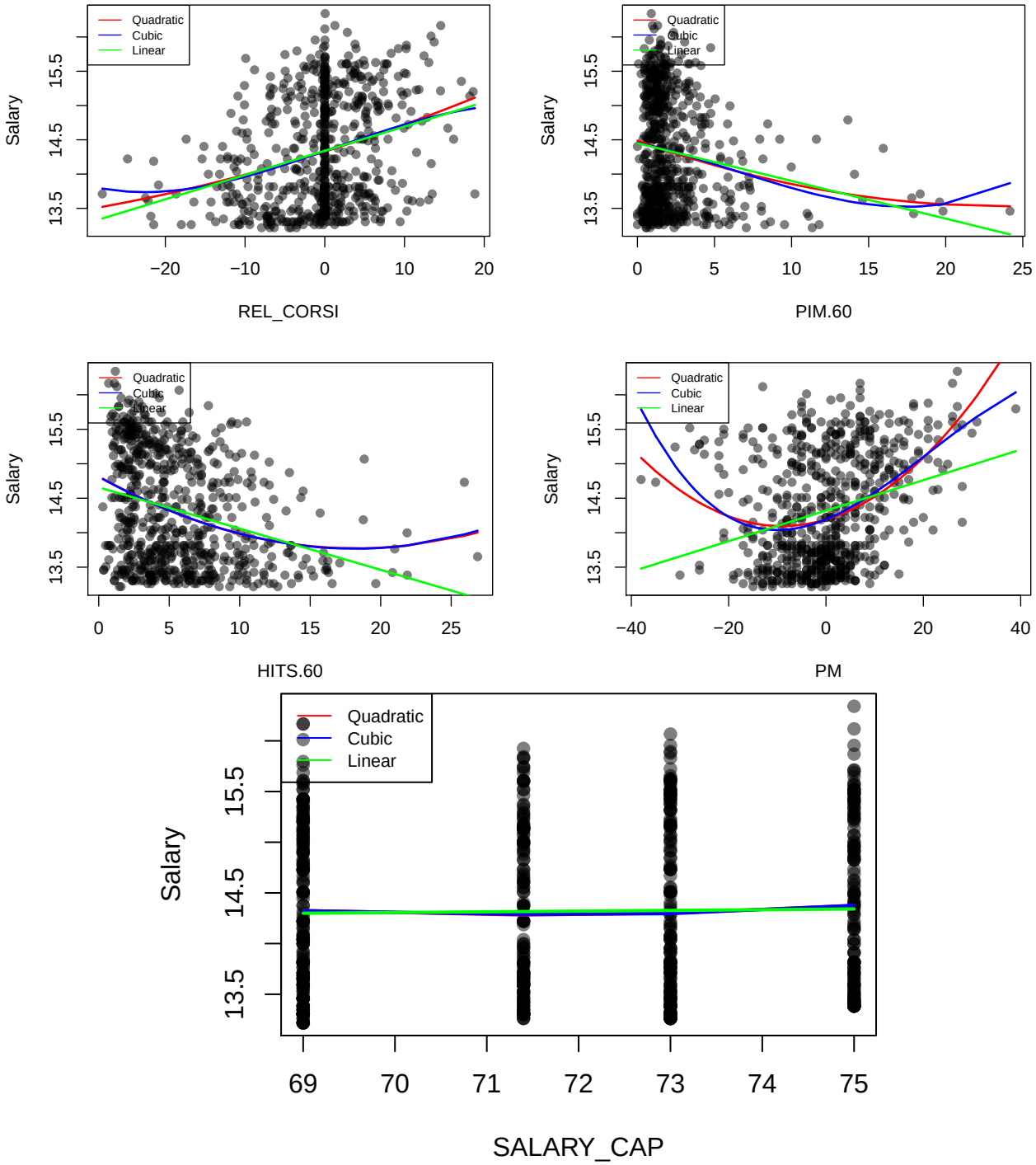


Table 9: Quadratic Model P-Values

Predictor Variable	P-Value Beta 1	P-Value Beta 2
GP	0.02	0.00
INJURED_GAMES	0.38	0.42
TOI.GP	0.05	0.07
P.60	0.69	0.00
G.60	0.26	0.01

Predictor Variable	P-Value Beta 1	P-Value Beta 2
AGE	0.06	0.08
TKWY.60	0.00	0.05
GVWY.60	0.00	0.01
REL_CORSI	0.00	0.50
PIM.60	0.00	0.32
HITS.60	0.00	0.00
PM	0.00	0.00
SALARY_CAP	0.35	0.35

Table 10: Cubic Model P-Values

Predictor Variable	P-Value Beta 1	P-Value Beta 2	P-Value Beta 3
GP	0.89	0.87	0.55
INJURED_GAMES	0.66	0.41	0.31
TOI.GP	0.00	0.00	0.00
P.60	0.31	0.08	0.35
G.60	0.67	0.11	0.29
AGE	0.00	0.00	0.00
TKWY.60	0.00	0.02	0.05
GVWY.60	0.13	0.77	0.81
REL_CORSI	0.00	0.88	0.44
PIM.60	0.35	0.65	0.49
HITS.60	0.02	0.59	0.94
PM	0.00	0.00	0.02
SALARY_CAP	0.91	0.91	0.91

Using the plots above and the corresponding p-values, we can determine which polynomial terms to include:

GP: The cubic term is insignificant, but the quadratic term is significant. Based on the plot, the inclusion of the quadratic fit better fits to the data, and thus including a quadratic GP term is reasonable.

INJURED_GAMES: The higher powers are insignificant and thus excluded.

TOI:GP: The cubic term is significant, however it does not seem intuitive that longer time on ice would lead to a lower salary (as depicted on the right hand side of the plot), so the inclusion of a cubic term does not seem reasonable. Furthermore, the quadratic term in the quadratic model is insignificant and thus also excluded.

P.60: The cubic term is insignificant, but the quadratic term is significant within the quadratic model. Thus we include a second power term for this variable.

G.60: The cubic term is insignificant, but the quadratic term is significant and appears to improve fit, so again, we include this second power term.

AGE: Both the cubic and quadratic terms are significant in the cubic model. The cubic model appears to have extreme behaviour at the endpoints; we would expect some difference in salary with respect to age, however the extremity of the curve towards the end appears to be exaggerated. The quadratic model appears to be more intuitive, however the quadratic term is insignificant. Thus we just include the linear term for age.

TKWY.60: The cubic term is insignificant, as well as the quadratic term in the second model. We use only a linear term.

GVWY.60: The cubic term is insignificant, so we look to the quadratic model in which the quadratic term is significant. Visually, the quadratic fit seems reasonable; players who giveaway the puck more often may be viewed as undesirable, and thus a decrease in salary may be expected.

REL_CORSI: Both power terms are insignificant.

PIM.60: Both power terms are insignificant.

HITS.60: The cubic term is not significant, but the quadratic term is. The quadratic fit is visually reasonable since players with more hits have better control of the puck for their team, which may lead to higher salaries.

PM: Both the cubic and quadratic models contain significant terms, however visually the models seem counter-intuitive. Players with low PM scores means the opposing team scores more often than their team while they are on the ice, which may be an indication of poor quality players. Thus, the linear model seems most reasonable since low PM scores lead to low salaries.

SALARY_CAP: Both power terms are insignificant and thus excluded.

8.6 Interaction Analysis

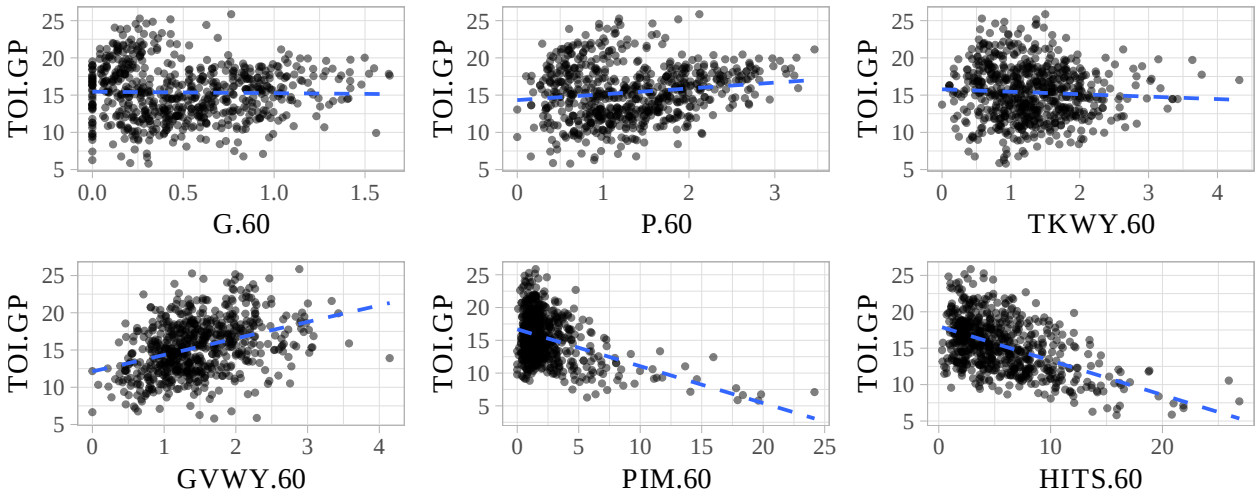
Within particular models, two-way interaction terms were considered as covariates to better predict the target salary of individual players. The selection of two-way interaction terms was based on intuition and knowledge of hockey, and these relationships are described below. Higher order interactions were excluded as there were no intuitive three way interactions between any of the initial predictor variables; introducing these terms risks increasing model complexity without a considerable improvement in fit.

8.6.1 Time on Ice per Game and Player Metrics

Players with longer time on ice per game have more opportunities to make an impact on the game via acquiring goals, points, takeaways, accumulating penalty minutes, hitting other players or giveaways of the puck. Intuitively, it makes sense to explore the relationship between these variables and time on ice per game to determine if there is a significant relationship that motivates the inclusion of two-way interaction terms within models.

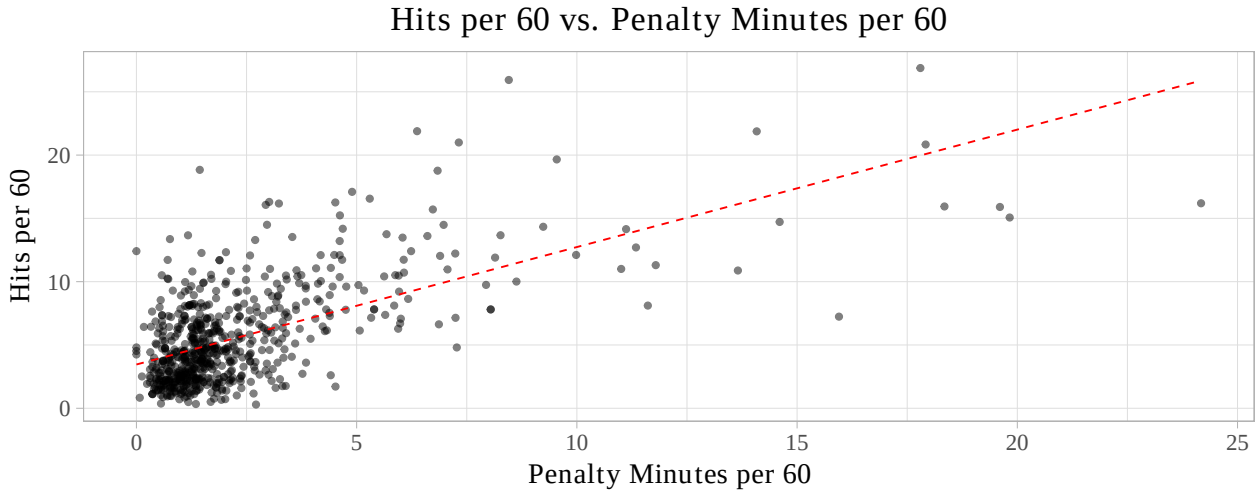
	Correlation with Time on Ice	Regression Estimate for Beta 1	P-value
Points (per 60)	0.1468373	0.78	0.00
Goals (per 60)	-0.0194358	-0.19	0.62
Takeaways (per 60)	-0.0566953	-0.33	0.15
Giveaways (per 60)	0.3687466	2.22	0.00
Penalty Minutes (per 60)	-0.4115554	-0.56	0.00
Hits (per 60)	-0.5128546	-0.47	0.00

With the exception of goals and takeaways, we note fairly large magnitudes for the correlation between time on ice with the player metrics. Furthermore, most of the β_1 values, corresponding to the regression slope for a linear model containing time on ice regressed on the player metrics individually, are significant. Thus, it seems reasonable to include interactions terms between time on ice and points, giveaways, penalty minutes, and hits within the model. In all cases, these interactions were subject to variable selection in the models, meaning their significance was further considered.



8.6.2 Relationship between Hits and Penalty Minutes

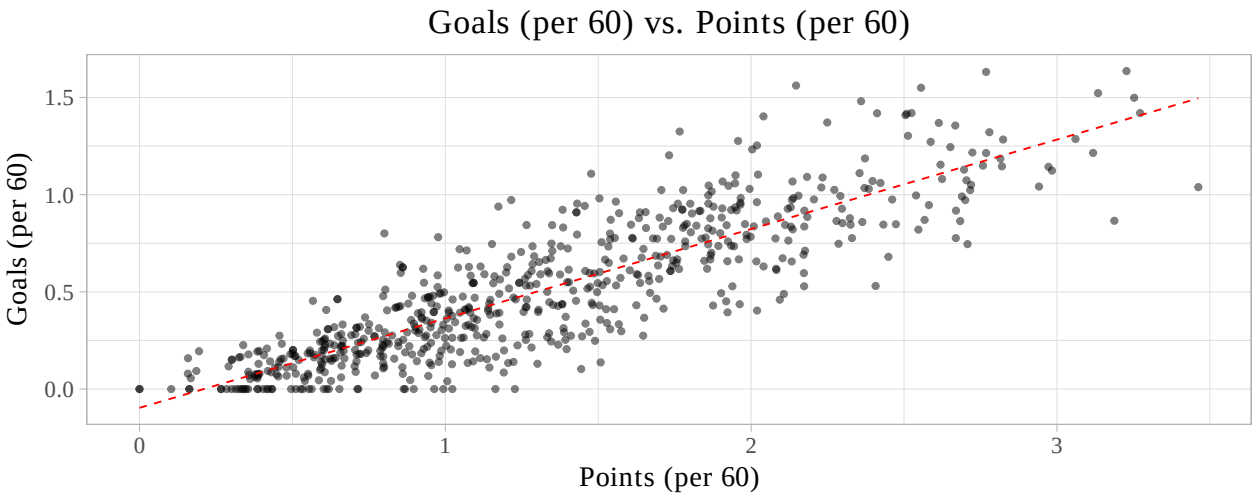
In circumstances in which one player achieves a hit (as defined above), this hit may have been achieved as a consequence of foul behaviour, resulting in penalty minutes being assigned to that player. Thus players with a high number of hits may also have a high number of penalty minutes assigned to them, which could result in these covariates interacting. The correlation between the numbers of hits and penalty minutes assigned to a player within the data is 0.62 which suggests this may be a relationship worth exploring.



The slope of the regression line is 0.4190715, with a corresponding p-value of 0, indicating that this is a additional two-way interaction to consider. Visually there also appears to be some form of relationship as well.

8.6.3 Relationship between Points and Goals

Similar to hits and penalty minutes, we can assume that there would be a relationship between points and goals, particularly for offensive players. The correlation between the numbers of points and goals (per 60) assigned to a player within the data is 0.86 which suggests this may be a relationship worth exploring.



The slope of the regression line is 0.4599014, with a corresponding p-value of 0, indicating that this is a additional two-way interaction to consider. Visually there also appears to be some form of relationship as well.

8.6.4 Player Position and Player Metrics

For the purpose of analyzing potential interaction between player position and metrics, we consider position as a binary variable (with levels of 'Forward' and 'Defense') as carried out in the pre-processing of the data.

Position	Avg. Hits	Avg. Points	Avg. Goals	Avg. Takeaways	Avg. Penalty Min	Avg. Time on Ice
Defense	58.39141	10.77566	2.525060	10.90453	23.53699	16.61459
Forward	58.46278	18.14516	7.794045	17.17866	23.44045	12.50922

Above we see relatively comparable average values for hits and penalty minutes with respect to either position. However, there is a reasonably large difference between the other metrics by player position. Thus, there appears to be some dependence on particular player metrics and the position they typically play on the ice. As a consequence, these are two-way interactions which may be beneficial to include in models prior to variable selection.

8.6.5 Player Nationality and Player Metrics

We consider the nationalities as decided in the pre-processing step:

Nationality	Avg. Hits	Avg. Points	Avg. Goals	Avg. Takeaways	Avg. Penalty Min	Avg. Time on Ice
Canada	60.08	14.47	5.62	14.22	26.00	13.50
Other	61.07	22.52	8.41	21.12	22.24	15.31
America	53.55	13.12	5.06	12.43	19.54	13.74

We note that appears to be some variance in the average player metrics with regards to nationality. It appears that the players outside of North America appear to achieve a higher average number of points, goals, and takeaways relative to North American players. Additionally, the number of average hits for American players appears to be considerably lower than the other categories. Consequently, it appears there

is some degree of interaction between player nationality and the variates discussed above. This motivated the inclusion of these two-way interactions within the data modeling portion of the report.

Appendix B - Literature Review

A typical literature review, in the context of our problem, is not necessarily appropriate. Because the dataset we are analyzing has been created by us through the amalgamation of data from multiple sources to gain a wide variety of possible explanatory variates, analysis has never before been conducted on the same data. Others who have previously looked into the problem of predicting player salaries used much more limited data in terms of the number of covariates. They also used data that was procured from different sources than ours, meaning that the results are not perfectly comparable. With this in mind, we can still review the techniques and results from similar research problems.

Firstly, there was a Kaggle competition that attempted to predict salaries for the 2016/2017 season. Data for this competition came from one of the sources that we used in our own joint dataset. This competition ended with posted results from only one individual, Cam Nugent. His approach was to use random forest and XGBoost, combine the two methods, and create an optimal joint prediction model. He used the naive estimate of median salary of the league for every player (regardless of unique covariate values) as a baseline for how well his model improvements were predicting. He also compared his results against the default XGBoost and random forest models. In the end, his random forest only model predicted to an average error of $1,578,497 and his XGBoost only model resulted in an average error of $1,574,073. After bagging these two models, he was able to decrease his error by around $25,000 on average.

Next, there are three papers which implemented varying methods of statistical learning with varying levels of predictive power. First, there was a thesis written by Kevin Peck at Western Michigan University. In his analysis, he used a very naive analysis. This is evident because the author initially only begins approaching the data by including variables that he deems, in his own opinion, to be important. This results in the inclusion of only less than 10 covariates. Then, using basic multiple linear regression, he excludes predictors that were not deemed significant based on a 0.05 significance level, leaving only 4 covariates and does not investigate any possible interaction terms. Finally, he uses natural logarithmic transformations as outlier handling. He does not attempt to apply his final model as prediction however, and without this test set, we are unable to truly infer how accurate his model was, though it likely was not as accurate as other models based on its simplicity in term of set-up and statistical techniques.

The next most sophisticated paper about explaining NHL salaries was completed by Vincent Eastman. His model utilized Quantile Regression: a method in which the ith quantile of the conditional distribution of salary (given covariates) is a linear function of the covariates. This means that, mathematically:

$$y_i = \beta_0(\theta) + \beta_1(\theta)x_i + \epsilon_i \text{ and } Q^{\theta}(y_i|x_i) = \beta_0(\theta) + \beta_1(\theta)x_i$$

for a response value y_i, quantile function Q and covariate x_i where $Q(\epsilon_i|x_i) = 0$. This means that, for each quantile of response variable (salary in this case), the model can have different parameter values and, potentially, different covariates that affect it. This method was then used in conjunction with a log-linear model. Another key difference between this analysis from other methods is that the author included height and weight as covariates in the model. These predictors are used in an attempt to characterize the value of defensive talent, which is a very valuable skill in hockey, but cannot be measured by traditional statistics such as goals or shots. The data also came from the pre-lockout period in the NHL, which was the period before the 2004/2005 season. We chose to use more recent data and included everything following the lockout season (post 2004/2005). Ultimately, the approach yielded varying degrees of effectiveness as it was divided up by position (forward or defense) and quantile of salary. This paper, again, did not attempt to predict salary for new observations that had not been used to generate the model, and thus is not directly comparable to our modelling efforts.

Lastly, an analysis by Emmanuel Perry implemented a K-Nearest Neighbours approach to predicting the player's AAV (average annual value - a form of salary that includes any bonuses, signing or otherwise).

The data used in this analysis came from a database "General Fanager" which is no longer available. Their analysis settled on 16 neighbours as the optimal number. The model iteratively assigned weights to each covariate that determined how much they impacted the prediction using the Nelder-Mead algorithm, made predictions and then updated the weights. The results of this method were better than any of the other papers. In the test set, it was observed that 85% of residuals fell below $1 million and almost 50% were before $250,000. The average error on out-of-sample predictions was $638,763 while the average in-sample error was around $500,000.

8.7 Sources

Nugent, Cam. https://www.kaggle.com/camnugent/nhl-player-salary-prediction-xgboost-rf-and-svm

Perry, Emmanuel. "Hockey and Euclid: Predicting AAV With K-Nearest Neighbours." (2016)

Peck, Kevin. "Salary Determination in the National Hockey League: Restricted, Unrestricted, Forwards, and Defensemen." (2012).

Vincent, Claude, and Byron Eastman. "Determinants of pay in the NHL: a quantile regression approach." Journal of Sports Economics 10.3 (2009): 256-277.

Appendix C - Modelling Details

Cook's Distance

One method of outlier handling that we tried was using the multivariate approach of removing variables with a large Cook's distance. That is, those with a Cook's distance more than four times the mean. This approach did not improve our results, so as discussed in the body of the report, we kept all data points.

PCA of Correlated Variables

In an attempt to remove the correlation of the variables, we tried using PCA on the sets of correlated variables (P.60, G.60) and (HITS.60, PIM.60), which correspond to offensive players and "grity" players, respectively. After getting the principal components, we selected the most important variable which had the vast majority of the variance, and replaced the variables in the model by the corresponding principal component. However, as mentioned in the model, this reduced the accuracy of the model. This is likely because points and goals can tell a different story. For example a player may be a great playmaker and have a lot of points, but not a great goal scorer, so their salary may be priced accordingly. As a result, both variables are obviously important for the predictions.

Gradient Boosting Model (GBM)

We initially fit a GBM to our data. However, XGBoost performed with similar accuracy and had faster performance, so it was kept in the report. Using the same methodology as the XGBoost, we found the best model. The model has a cross-validation RMSE on our training set of 0.3695, and a validation RMSE is 0.3747. Both of the RMSEs are close, so there is no evidence that the model is outperforming on either set. These values are very similar to the errors of the random forest.