Data Mining Final Project: Hockey Mine

Cameron Lloyd
Ohio State University
lloyd.331@osu.edu

1. Abstract

This project uses supervised learning to predict the outcome of a National Hockey League head-to-head matchup between two teams. I attempt to solve the classification problem by first gathering a dataset of the all the games played in the 2015-2016 NHL season, using as many game-time statistics that may influence the outcome of any given match. My approach combines using both traditional hockey statistics, such as goals scored and winning percentages, as well as some more advanced, newer statistics such as Corsi-for and PDO. These features are then explored in an attempt to learn more about how each one may affect the outcome of a given hockey match. Using this dataset, I then construct several different classification models and evaluate each one to find which one will produce the most accurate result. My results indicate that both the Gradient Boosted and RandomForest Classification models produce the most accurate predictions of a given NHL matchup's result. It was also found that the most informative features of an NHL matchup were the team's current winning streak as well as the location of the game.

2. Introduction

2.1 Problem Statement

In recent years, various sports have become increasingly dependent on using advanced analytics to evaluate the performance of a team. If you've ever seen the movie Moneyball, or have ever read the book, you'll know that baseball has been a pioneer to the movement of using advanced statistics to build a better team. Even more recently, the Cleveland Browns have shifted towards this movement with the hiring of Paul DePodesta in desperation of turning their team around into a winning one. Hockey, however, is fairly new to integrating advanced analytics into their performance assessment. If one could use the library of NHL statistics available to predict the final season outcome of an NHL team, it could be possible to use these same statistics to evaluate and improve any given team.

In this project, I attempt to break down this problem into the classic subproblem of whether or not it is possible to successfully predict the outcome of a games in a previous NHL season. Using supervised learning models, this project predicts the outcomes of all games in the final third of the 2015-2016 NHL regular season schedule.

2.2 Related Work

This project was inspired by a love for hockey, but further research suggests that attempts at solving this problem have previously been made. The most notable attempts at this found were "Forecasting Success in the National Hockey League using In-Game Statistics and Textual Data" by Joshua Weissbock at the University of Ottawa in 2014 [1]. Another attempt [2] has been made by Weissbock with a few

colleagues, but this project chose to reference Weissbock's solo attempt since it provided much more detail.

In Weissbock's effort he used Weka to build a Neural Network, a Naive Bayes, a Support Vector Machine and a Decision Tree model all with 10-fold cross-validation on a 66/33 training to test split. Using all game data before and after every game for the 2013-2014 NHL season, his final results for this report suggested that the Neural Network provided him with the most optimal results with an accuracy of 59.38%. His results also suggested that the three most informative metrics in his dataset where the location (Home or Away), the Goals Against amount and the Goal Differential.

Although Weissbock's results we're fairly insignificant, his work provided some insight in my report; most notably the "luck" statistic, PDO, and what models to use. I was previously unaware of the "luck" statistic before reading his report. He also provided a few models and their results so I could get a baseline on which models to, and not-to, use.

2.3 Outline

In Section 3 I will highlight some of the technologies during this process. In Section 4 the process of how the NHL matchup information obtained, as well as a short description of a few features used. In section 5 I will describe the approaches used for preprocessing and exploratory data analysis. Section 6 will consist of both the classification model evaluation as well as the final classification models used. Finally, Section 6 will contain the results and a brief discussion of my work.

3. Technology Used

For the remainder of this report please note that all data scraping, preprocessing, exploration and modeling was done through R. Please see the README included in this directory for a short description of where the code base is located and what it represents, as well as some libraries that will be required for execution.

The README also includes a list of the libraries containing some of the functions used in my scripts. Noteable tools used include 'caret', 'tree', 'randomForest' and 'GBM' for data model creation. The web-scraping was performed using R's package 'rvest'. Graphing was done using R's 'ggplot' libraries and a config file was used to hold key terms by using the 'config' package.

The GitHub repository https://github.com/cameronlloyd/HockeyMine was used to store the code base.

4. Data Retrieval

The full list of features can be found in Appendix A, but I will try to note as many relevant ones as I can for each step.

4.1 Features and Data Scraping

All data that was used in this project was scraped from http://www.hockey-reference.com/ [3]. The scraping process itself was a bit rigorous due to the fact that the advanced hockey analytics community is still growing and not all data can be found on the same page. For each record, or event, in this dataset, the attributes were all either accumulated or averaged, depending on the feature, to represent the summary of the given team at that point in time. This is done since there is currently no way to view a team's record at any given time. Thankfully, most of the features concerned by this project were able to be found through some method of scraping this website.

The first step in this process was to retrieve the full schedule for each team. For each team, the script would find the page listing their full schedule and essentially copy all relevant information to be stored in a separate CSV file to be used in a later step. Of course, much more information was grabbed in this step, but the following list contains some of the most noteable features used in a later step.

- GP: The amount of games played before the current event.
- Date: The date that the current event is being played on.
- Location: Either 'Away' or 'Home', depending on where the current event takes place.
- Opponent: The team opposing the team that this schedule is being scraped for during the current event.

I would also like to note that at the time of writing this, the script that performed this task is currently missing from the repository. The script itself took 5 hours to create and was lost during a system crash. Due to time constraints, I haven't been able to recreate it at this time.

The second step of this process was to find more matchup relevant features that were not available in the schedule's summary. For each record in each team's schedule, the script would use the events Date, Location and Opponent to create a unique game ID to reroute to the page that includes more information about the given matchup. This is possible due to each matchup in Hockey-Reference having a unique link. In this step more advanced statistics were obtained, such as the corsi-for percentage for all close, 5v5 and even strength situations which are briefly described below.

- Corsi-For: Shot Attempts = Shots + Missed Shots + Blocked Shots [4].
- Corsi-Against: The Corsi-For amount for the opposing team.
- Corsi-For %: Corsi-For / (Corsi-For + Corsi-Against) [4].
- Close Situation: When either team is tied or has a one-goal lead.
- 5v5 Situation: All situations where either team is off the ice for a penalty.
- Even Situation: Includes 5v5 situations, but also includes situations when both teams have a play of the ice due to a penalty.
- Corsi-For % (Close): The Corsi-For % during close game situations.
- Corsi-For % (5v5): The Corsi-For % during 5v5 situations.
- Corsi-For % (Even): The Corsi-For % during even strength situations.

The third step of this process was retrieve 2 more statistics: the average age of plays on the team and the PDO. The PDO, which is referred to as the "luck" statistic, was created to represent how "lucky" a team has been. PDO is the summation of a team's Shot Percentage and Save Percentage. PDO is not an acronym but rather a name after the Internet username of the person who created it. It has been found to be a useful statistic due to Hockey appearing to have stochastic processes play a much larger role than

other sports [1,2]. Since this is a harder statistic to find, the PDO and the average age was found on each team's summary page and is thus not accumulative, rather the season's average.

The final step of this process was to merge all matchups in each team's schedule into one matchup dataset ('Matchups.csv'). This was mostly accomplished by joining all matchups in each schedule on the unique game IDs created earlier. The matchup dataset was then sorted in descending order on the date. The most important part in this step was to verify that there were no duplicate records. This matchup dataset was used, and will be mentioned, throughout the rest of this report.

The complete dataset created contains 1230 records with 69 features. 64 of these features are specific to the statistics of each team; with both teams each having 32 team specific records. Thus, it appears that there are duplicates of 32 of these features. This is because each record contains information for each matchup between two teams. The other non-team specific features, referred to as matchup specific features, are the row ID, the Date, the Home Team, the Away Team and the Result of the event. The result of the event will be either a 1 or a 0, with it being a 1 if the home team was declared the winner (either in regulation, overtime or shootout) of the current event, and a 0 if it was the away team who won.

5. Data Preprocessing and Exploration

The first step to processing the matchup dataset was to first remove all row IDs, Dates and Team Names. These features were only left on the dataset for easy readability and filtering, and since these features have no part in deciding the outcome of the game, they were removed.

The next two paragraphs are referring to both team's, team-specific records. Each team's record in the dataset contains a feature called Streak. On hockey-reference, this was represented with an 'L' followed by a number representing a losing streak, or a 'W' followed by a number representing a winning streak. To convert these to a numeric value, the number following was multiplied by either a 1 or a -1 depending on the type of streak. A positive value for Streak represents a team that has one 1 or more of the previous games, with a negative value representing a losing streak. A 0 will hold it's place during the first game of the season.

This set also contains the record that the team has had during the last 10 games. These are denoted by the features L10Wins, L10Losses and L10OL. All values in L10Wins will be the amount of games won in the last 10 games by the team, L10Losses is the amount of games lost by the team and L10OL is the amount of games lost in overtime by the team. To provide these features with meaningful values, for each record in the dataset, if either time has played less than 10 games that record was dropped. This is also likely to help improve the fitting of the model. At this stage, the dataset contains 1,089 records with 65 features and at this point no other features will be removed. At this point, all processed data at this point is stored in a separate dataset called 'FilteredMatchups.csv'.

In order to gauge how impactful the differences between each team's, team-specific attributes are, I then created a difference matrix. This matrix is stored in the file 'DifferencesMatchup.csv'. This matrix was

created by simply subtracting the away team's features from the home team's features where their feature name's matched. From this matrix, I calculated summaries of each attribute. The summaries are stored in the file 'DiffSummaries.csv'. From these summaries, 9 attributes seemed to have a decent amount of spread and were thus plotted and observed. The plots can be seen below in Figure 1. Since this dataset is so highly dimensional, I am only plotting the 9 mentioned above.

Figure 1: Observing The Spread Between Various Attributes.
From Left to Right, Top to Bottom These Include:
{ROW, L10Losses, SHG, PKG, ShotsAgainst, ShotsFor, SV, AvGoalieCnt and HomeP}

From these plots it can be observed that this dataset can be further filtered to remove outliers, noticed by

the points further from the borders of the whiskers. To mitigate this I identified the maximum and minimum threshold value for each feature. For the maximum value of each attribute I used the attributes first quartile value and added it to the product of the attributes Interquartile Range multiplied by 1.5. Similarly, the minimum value for each attribute was found by using the third quartile value of the attribute and subtracting from it the same product mentioned previously. For any record in the differences matrix that contained an attribute value that fell out of this range, its row ID was recorded so that it's representative row could be removed from the FilteredMatchups dataset. As a result of this step, 180 more records were removed from the dataset leaving a final number of records at 909. The resulting set is stored separately in 'FilteredMatchups2.csv'. Since the amount of samples in this dataset is fairly small, no more modifications were done to the dataset for fear of overfitting.

6. Model Evaluation

The FilteredMatchups2 dataset now being used consists of observations from 909 NHL games during the 2015-2016 regular season. I then split the dataset using 66% percent of it for training and the remaining for testing. This would leaves me with 559 observations to train on, and with 310 observations to test with. Since the feature values in the dataset depend on the previous events, I wanted to leave the datasets ordered as much as I could. This meant that the first 559 observations were used for training with the remaining for testing. I then divide the training set into 7 folds, and for each model being considered, I use cross-validation to evaluate its performance. In cross-validation, I evaluate models using the true positive rate, rather than accuracy rate because I want to prioritize avoiding false negatives.

6.1 Model Selection

I evaluated using the following model types:

- Decision Tree
- Random Forest
- AdaBoost (Using Trees)
- Logistic Regression

Aside from the Decision Tree, none of these models were attempted in the previous reports mentioned early. This is important to me since although my methodology is a bit different in a few ways, I didn't want to build models using only what's been tried before. The goal is to learn something from this.

6.1.2 Models and Parameter Settings

In this section, I discuss the strategy in using Decision Trees, Random Forests, AdaBoost, and Logistic Regression.

6.1.2.1 Decision Tree

Evaluation of the decision tree was done using a 7 fold cross-validation split. The initial split of the tree was done on the Home Streak feature, suggesting it carries a higher weight than the other attributes. This

is also interesting to note since it agrees with the prior research. Secondary splits included the away team's winning percentage when playing as the away team as well as the home team's last 10 game overtime loss record. This suggests that a home team losing a previous game in overtime has a heavy influence of the outcome of the following game. The average true positive rate for each fold produced from this model was found to be 87.13%

6.1.2.2 Random Forest

The random forest randomly selects a certain number of predictors to consider at each split. We use cross validation to determine the 'optimal' number of predictors to consider at each split. Note that the number of trees is chosen so that the amount is not too disproportionate to the dataset. The 7 fold cross validation results for various settings of the number of predictors randomly considered at each split are displayed in table 1.

Table 1
Number of Trees: 100

# Predictors Considered At Each Split (bags)	7-fold TPR
5	90.92%
10	94.36%
15	95.36%
20	95.14%
27	94.01%

There isn't much of a difference in the True Positive rate across the different parameter settings. The difference might just be due to the random nature the random forest, and may not have anything to do with the selection of p. But p=15 does give the highest 7-fold TPR.

6.1.2.3 Boosting

Using tree based boosting, I choose a learning rate equal to .01, and cross validate over a grid of values for the maximum depth of each weak learner allowed [1,2,4,6], and the number of weak learners used [300,500,700]. The smaller the shrinking parameter the better the predictive performance, but the worse the computational performance. Since I'm not concerned with the computational performance, a smaller value for the learning rate is used. With 300 models, and a maximum tree depth of 1 levels, I achieve the an extraordinarily high true positive rate of 100%.

Further examining the model I find that the average accuracy for all 7 folds at this depth is 88.20%. This leads me to believe that this model produces more optimistic results.

6.1.2.4 Logistic Regression

Now I evaluate a Logistic Regression model using cross-validation using all attributes. The average true positive rate of all folds is found to be 89.62% .

Perhaps the most interesting trait to note from this model is the coefficient values generated by it. Table 2 below shows the results

Table 2

Feature	Coefficient Estimate (R^2)
Home, Streak	6.594
Home, L10 Wins	-3.254
Home, L10 Losses	-3.508
Home, Average Corsi-for (Close)	-1.710
Home, Average Time on Ice	-1.838
Away, Streak	-3.485
Away, Away Win Percentage	-3.538

This suggests that whenever the current game is played at home, the streak that the home team on is highly influential of the result. Conversely, if the away team has a high win percentage against away opponents, the away team is more likely to win.

6.1.2.5 Summary of Results

Table 3 shows 7-fold Cross Validation True Positive Rates at the 'best' setting for each model form

Table 3: 7-Fold Cross Validation TPRs

Model Type	7-Fold TPR
Decision Tree	87.13%
Random Forest (p=15, Number of Models = 100)	95.56%
Boosting (MaxDepth=1 & Number of Models = 300)	100%

Logistic Regression	89.62%

Although the Boosting provided a TPR, I'm fairly skeptical this wasn't caused by human error. Thus I will continue to build my final model with both the RandomForest with p=15 and the number of models = 100, as well as the Boosted Tree using interaction depth=1 and the number of models =300.

6.2 Prediction

Each model is ran 6 times using an increasing cutoff threshold each run. The thresholds used are 0.5, 0.6, 0.7, 0.8, 0.9, 0.95. The best true positive rate with the threshold cutoff value used is reported.

6.2.1 Random Forest Results

I'll begin by fitting a random forest with p=15 predictors on the entire training set, and then use the resulting model to predict the results of the remaining test set. To predict the model I used the same 6 cutoff thresholds mentioned earlier above. The results can be seen in Table 4 and Figure 2 below

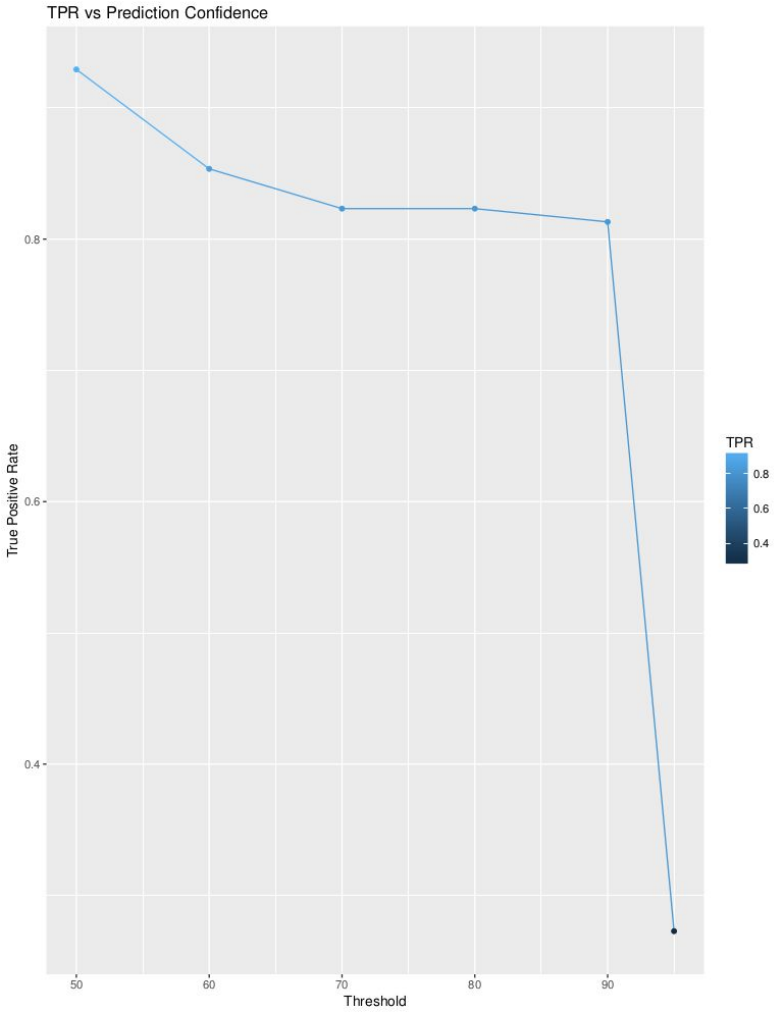

Figure 2: TPR Prediction Confidence Using Different Thresholds

Table 4: RandomForest Model Evaluation At Different Thresholds

Threshold	TPR
50%	92.93%
60%	85.35%
70%	82.32%
80%	82.32%
90%	81.31%
95%	27.27%

This suggests that the RandomForest model created produces a 92.93% true positive rate with 50% confidence. What's most interesting to note about these results are that the RandomForest model is fairly confident up until 90% confidence.

6.2.2 Boosting Results

Similar to above, I used the same 6 cutoff thresholds to evaluate the confidence in the predictions using the boosted tree. I continue using a shrinkage factor of 0.01, a bag faction of 0.5, interaction depth 1 and using 300 models. The results can be seen in Table 5 and Figure 3 below

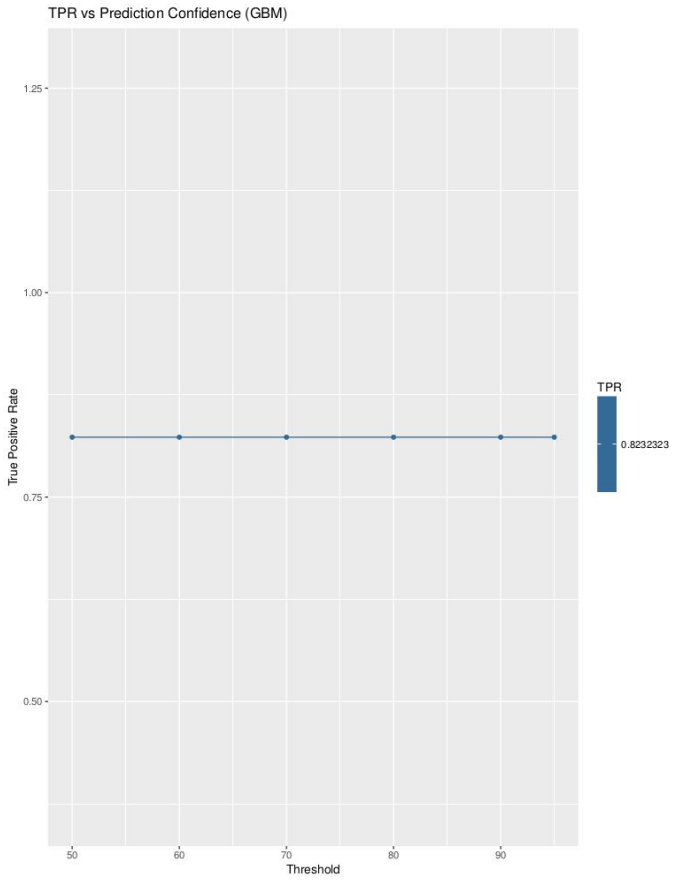

Figure 3: TPR Prediction Confidence Using Different Thresholds (Boosting)

Table 5: Boosting Model Evaluation At Different Thresholds

Threshold	TPR
50%	82.32%
60%	82.32%
70%	82.32%
80%	82.32%
90%	82.32%
95%	82.32%

Compared to the RandomForest model, the Boosting model produces a 82.32% true positive rate with "complete" confidence. The downside being that the true positive rate is much less with 50% confidence than the RandomForest model.

7. Discussion

7.1 Analysis of Results

From these results, it's suggested that the RandomForest has the most upside, with just a slight amount of risk provided the confidence required isn't greater than 90%. Conversely, the Boosting model has slightly less reward, but the risk factor is completely eliminated. Since hockey is game consisting of a great amount of luck, the Boosted model may be the best classifier to use when simulating a larger dataset.

For both of the final models, the approximate true positive rate given would be 82.32%. While the Boosted model produces a TPR of 82.32% no matter the confidence, the RandomForest produces a TPR of 82.32% for both confidence intervals of 70 and 80%. In either case, the average TPR produced is much greater than the models used in previous experiments.

7.2 Limitations and Future Work

To produce a greater accuracy than the models in this report, one would need to use more features, or at least more descriptive features. Some features I would have liked to included in this model are how the team performs during a given month and time of day and the teams Fenwick Percentage which wasn't able to be found through Hockey-Reference. I would have also liked to have found some sort of 'Arena Factor'. Ideally, this would provide a different weight to teams when they play at home depending on their arena. Similarly, it would provide a metric for how much more difficult it would be to win in a "hostile" environment. Since hockey is still expanding in terms of advanced analytics there are limitations to the amount of statistics readily available.

Another issue that I'm well too familiar with after this project is the issue of being able to get all statistics at once. Hockey-Reference is helpful in providing a place to retrieve all statistics in one place, but it still has a ways to go in terms of the metrics it provides.

An additional improvement that could be made would also be to provide more useful graphs. Some of this is due to me being new to R and being unfamiliar with the graphing libraries, but most of these issues could be improved by creating better statistics.

8. Appendix B: Feature Vector

- GP: Games Played - The amount of games played so far in season.
- GF: Goals For - The amount of goals scored by the team.
- GA: Goals Against - The amount of goals scored by opposing teams
- Wins: The amount of wins the team has had so far in season.
- Losses: The amount of losses the team has had so far in season.
- OL: Overtime Losses - The amount of losses the team has had in overtime so far in season.
- Streak: The amount games the team has either lost(-) or won(+) in a row.
- ROW: Regulation and Overtime Wins - The amount of wins the team has had in regulation or in overtime this season (i.e. not shootouts).
- L10Wins/Losses/OL: The results of the team for the previous 10 games.
- PPM: Power Play Minutes - Time in minutes the team has had the power play.
- PPG: Power Play Goals - The amount of goals the team has scored on the power play.
- PPO: Power Play Opportunities - The amount of opportunities for a goal the team has had on a power play.
    - Power Play: The opposing team has committed a penalty. Thus, for a period of time (usually 2 min) the fouling team has one less player on the ice.
- SHG: Short Handed Goals - The amount of goals scored while the team is a man down.
- PKM: Penalty Kill Minutes - Time in minutes the team has had the Penalty Kill.
- PKO: Penalty Kill Opportunities - The amount of opportunities for a goal the team has had on a Penalty Kill.
    - Penalty Kill: The team has committed a penalty. Thus, for a period of time (usually 2 min) the team has one less player on the ice.
- Shots For: The total amount of shots the team has had.
- Shots Against: The total amount of shots the team has faced.
- SV: Save Percentage - The ratio shots saved over shots faced for a given goalie.
- AvCFClose: The Corsi-For % during close game situations.
- AvCF5v5: The Corsi-For % during 5v5 game situations.
- AvCFEven: The Corsi-For % during even strength game situations.
- AvGoalieCnt: Average Goalie Count - The average amount of goalies the team has played per game.
- AvShiftCnt: Average Shift Count - The average amount of shifts per skater.

- ATOI: Average Time On Ice - The average amount of time the collective amount of skaters has been in play.
- AvAge: Average Age - The average age of players on roster.
- PDO: Statistic determining “luck”.
- WinP: Win Percentage - Ratio of wins over total amount of games.
- HomeP: Home Percentage - Ratio of wins over total amount of games at home.
- AwayP: Away Percentage - Ratio of wins over total amount of games away.
- Results: 1 if game result in the home team winning, 0 otherwise.

8. References

[1] Weissbock, Joshua. *Forecasting Success in the National Hockey League Using In-Game Statistics and Textual Data*. Thesis. University of Ottawa, 2014. 25:35., n.d. Web. 10 Oct. 2016. <https://www.ruor.uottawa.ca/bitstream/10393/31553/3/Weissbock_Joshua_2014_thesis.pdf>

[2] Weissbock, Joshua; Viktor, Herna; Inkpen, Diana. *Use of Performance Metrics to Forecast Success in the National Hockey League*. University of Ottawa. 1:10, n.d. Web. 10 Oct. 2016. <https://dtai.cs.kuleuven.be/events/MLSA13/papers/mlsa13_submission_2.pdf>

[3] "NHL Stats, History, Scores, & Records | Hockey-Reference.com." *Hockey-Reference.com*. N.p., n.d. Web. 10 Oct. 2016.

[4] HockeyAnalysis.com. "Stats.HockeyAnalysis.com." *Glossary of Advanced Hockey Statistics*. N.p., n.d. Web. 15 Dec. 2016.