Forecasting Success in the National Hockey League using In-Game Statistics and Textual Data

by

Joshua Weissbock

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the M.Sc. degree in
Computer Science

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

In this thesis, we look at a number of methods to forecast success (winners and losers), both of single games and playoff series (best-of-seven games) in the sport of ice hockey, more specifically within the National Hockey League (NHL). Our findings indicate that there exists a theoretical upper bound, which seems to hold true for all sports, that makes prediction difficult.

In the first part of this thesis, we look at predicting success of individual games to learn which of the two teams will win or lose. We use a number of traditional statistics (published on the league's website and used by the media) and performance metrics (used by Internet hockey analysts; they are shown to have a much higher correlation with success over the long term). Despite the demonstrated long term success of performance metrics, it was the traditional statistics that had the most value to automatic game prediction, allowing our model to achieve 59.8% accuracy.

We found it interesting that regardless of which features we used in our model, we were not able to increase the accuracy much higher than 60%. We compared the observed win% of teams in the NHL to many simulated leagues and found that there appears to be a theoretical upper bound of approximately 62% for single game prediction in the NHL.

As one game is difficult to predict, with a maximum of accuracy of 62%, then predicting a longer series of games must be easier. We looked at predicting the winner of the best-of-seven series between two teams using over 30 features, both traditional and advanced statistics, and found that we were able to increase our prediction accuracy to almost 75%.

We then re-explored predicting single games with the use of pre-game textual reports written by hockey experts from `http://www.NHL.com` using Bag-of-Word features and sentiment analysis. We combined these features with the numerical data in a multi-layer meta-classifiers and were able to increase the accuracy close to the upper bound.

Acknowledgements

I want to thank all of those who have played a role in helping me complete my two years at the University of Ottawa. First and foremost is my supervisor Dr. Diana Inkpen who has helped me from day one. Thanks to her dedication and effort I have been able to successfully enter the University of Ottawa and complete my Masters of Computer Science; as well she has given me many research opportunities, to be able to develop as an academic researcher, and allowed me to take on a project such as this one.

Secondly many thanks goes to my fiancée who has thankfully been able to put up with the amount of hockey I've watched over the last two years for "research" and for the plenty of hours she has put in to help me review my writing. She will be glad for me to be finished school so I can spend less time watching hockey.

I cannot forget my parents and family who have been supporting me with my schooling since I was five. I am sure, no matter what I take on next, they will be there with me cheering me on the entire way.

My final thanks go to all of my supervisors at work who have been very flexible over the last two years to allow me to attend school while working full time. If it was not for them I would have not been able to complete my requirements. Many thanks go to Lieutenant-Commander Kris Langland, Major John Yorke, Major J.P. Paris and everyone else who has supported me in my Chain of Command.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introduction

Hockey is a team sport that is popular in Canada and Northern Europe and is similar to Football (Soccer) on ice. A game is played between two teams, of varying roster sizes, with five players and a goal keeper ("the goalie") on the ice at a time. The objective of the game is for the five players to put the puck (a round rubber disk, 25 mm thick, 76 mm in diameter) in the opponent's net using their hockey stick which is often made of wood or composite material. The goalie's job is to prevent goals being scored, by stopping the puck from entering the net with his body or his hockey stick.

The game is regularly played over 60 minutes, commonly broken into three periods of 20 minutes; time does not run constantly, as the clock stops after the referee has blown the whistle after an infraction of the game's rules. When a team is penalized for breaching a rule of the game, the common penalty is for that person to be excluded from play for a pre-determined amount of time (usually two minutes). During this time the penalized team has one less player on the ice giving the other team a "man-advantage," making it easier for the team who had received the penalty to concede a goal. This is common referred to as a "Power Play" when you legally have one (or more) players on the ice than the other team, or as a "Penalty Kill" when it is your team who has less players.

The most popular and competitive league in the world is the National Hockey League (NHL) which is based in North America. There are 30 teams from which 7 are in Canada and the remaining 23 are in the United States. While their organization into divisions has recently changed, during the time-frame of this research (2012-2013 NHL season) the 30

teams were divided geographically into two conferences of 15 ("Eastern" and "Western") and each conference was further subdivided into three divisions of five teams (North-West, Pacific, Central, Atlantic, North-East and South-East). Teams within the same division play each other 6 times a year, within the same conference, but not the same division. Teams play each other 4 times a and 1-2 times a year if they are in opposite conferences (determined by random chance each year based on the leagues scheduling algorithm).

Changes to the organization of the league changed at the start of the 2013-2014 season by re-organizing the divisions. The new organization does not appear to have an impact on our results.

At the end of the 82 game regular-season schedule (occurring from October through April), the top eight teams in both conferences move on to the post-season championship tournament and play for the "Stanley Cup". Teams are seeded so that top-ranked teams play the bottom ranked teams and all pairs of teams play a best-of-seven series. The first team to win four games moves to the next round for subsequent re-seeding and playing another best-of-seven series, while the losing team is eliminated from the tournament. This continues until only one team remains and that team is crowned the champion of the season.

1.2 Motivation

While hockey is a fascinating and intricate sport to watch and play, it is only in the infancy in its analytics. Compared to sports such as basketball and baseball, the amount of statistical knowledge known about the game is only at the beginning. This is possibly because of three issues:

1. Firstly, hockey is not globally popular compared to sports such as basketball, baseball and football.

2. Secondly, the difficulty of analyzing the sport, as it is a fluid game where there are few defined events, and events such as "goals" do not occur frequently due to their small sample size which often is washed over by the noise of random chance.

3. Thirdly, most hockey leagues have rules established create parity amongst teams. In the NHL, teams are limited on how much money they can spend on player's

salaries in a year. This causes teams to be much more even in talent than you would see in other sports such as football.

It is in this difficulty from which we derive our motivation; our aim is to automatically learn from the noisy data and make sense of the games using data mining techniques. Machine learning is a branch of artificial intelligence that applies statistical methods with algorithmic modeling in order to teach a computer (machine) tasks such as "recognition, diagnosis, planning, robot control, prediction etc" (Nilsson, 1996). With these techniques, we can analyze large amounts of data for patterns which allow us to discover if it is possible to learn from the few events that happen in a game of hockey and predict the final outcome. If it is not easily possible, we want to know why this is.

1.3 Goals

This thesis has several goals. Within the context of the sport of ice hockey, we aim to use data mining techniques and machine learning algorithms in order to automatically learn and predict the outcome of a single game. We look to see if it is possible to predict single games from the text of pre-game reports as well as the common in-game statistics. Because of the difficulty that we encounter when predicting a single game, we want to identify the theoretical upper bound in predictions for sports and then we predict the outcome of best-of-seven series between two teams. As there are more games, the random chance starts to normalize and regress to the mean, making predictions easier, and possibly increasing the final overall accuracy.

1.4 Hypothesis

Our hypothesis of this thesis claims:

> Despite hockey's constant flow and its low number of events a game, it is possible to predict the outcome of hockey games automatically, using data mining techniques applied to numerical and textual data, more accurately than the baseline. Though this is difficult because there exists an upper bound in hockey due to the random chance that plays a large role in each game.

1.5 Intended Contributions

This thesis brings the following contributions:

1. While machine learning has been used in others sports, to our best knowledge; we are the first to apply machine learning to anything related to hockey. Previous authors have explored hockey with statistical models to analyze the sport, but none have been in forecasting future events.

2. We explore a theoretical upper bound in hockey predictions using the standard distribution of observed win-percentages compared to a theoretical skill/luck split. This subject has received some attention on Internet blogs in reference to American Football by comparing a single season observed win/loss records to theoretical skill/luck split records.

3. We present a method of predicting sports using textual data. By using the pre-game reports written by experts, we expect to learn from the biases in the writing and to automatically learn which team is likely to perform better. Evaluations suggest that the method performs nearly as well as the numeric data within hockey prediction and combined together can increase the overall accuracy.

4. Finally we present a meta-classifier. To our knowledge meta-classifiers have not been used in sports predictions. We use numerical data, the pre-game textual data and sentiment analysis in their own individual models and the feed the outputs into a second layer meta-classifier. We explore multiple variations on how to determine the final prediction of a game including a cascade-classifier, voting and highest confidence. The results are positive and suggest a higher accuracy is possible, despite the difficulty due to the minimal gap between the baseline and the theoretical upper bound.

1.6 Outline

The remainder of this thesis is outlined as follows. In Chapter 2 we cover related works by reviewing previous machine learning and statistical modeling of sports, both in hockey and other major sports. In Chapter 3, we review the first model that we use to forecast outcomes in single hockey games. We use both traditional statistics and performance metrics to model a game, learn from the data and determine which team will win. In

Chapter 4, we look at the difficult of predicting a single game in hockey and in sports by looking at the random chance involved. We use a Monte Carlo method to analyze observed win-percentages to determine a theoretical upper-bound in sports prediction. In Chapter 5, we look at predicting best-of-seven, post-season championship tournament series. As one game is difficult to predict, the random chance and noise should start to regress to the norm over seven games, making prediction easier. In Chapter 6, we expand upon this model by using textual data from pre-game reports from expert writers, as well as sentiment analysis. We create three separate models from each type of data (numerical, sentiment and textual) and form a meta-classifier by using the output from each model in a second layer. Finally, in Chapter 7, we wrap up these ideas and conclude with a brief discussion, review our work and contributions and look at future work.

Chapter 2

Related Work

2.1 Background on Machine Learning and Text Classification

In Alan Turing's 1950 seminal paper, "Computing Machinery and Intelligence" he stated that the question, "Can machines think?" should be changed to, "can machine do what we do?" (Turing, 1950). In 1959, Arthur Samuel further explored this idea and defined the field of machine learning as "the field of study that gives computers the ability to learn without being explicitly programmed" (Simon, 2013). This was further refined in Tom Mitchell's 1997 book to: "a computer program is said to learn from experience E with respect to some class of task T and performance measure P, if its performance at task T, as measured by P, improves with experience E" (Mitchell, 1997). That is to say that machine learning is the field of artificial intelligence that deals with the learning from data. Machine learning can be broken into many subfields such as supervised and unsupervised learning.

Data mining is a related field. Similar to how gold mining is the function of searching for gold from rocks or sand, data mining is the idea of "extracting or 'mining' knowledge from large amounts of data." More appropriately it could be expressed as the idea of knowledge mining from data (Nilsson, 1996). Data mining involves the intersection of a number of computer science and mathematical fields including artificial intelligence, machine learning, statistics and database systems (ACM, 2006). In this work, we use data mining with machine learning algorithms and statistics in order to forecast the winners of individual hockey games. We also use text classification in order to improve our results.

Text classification is the automated categorization of natural language texts with thematic categories into a set of predefined classes (Sebastiani, 2002). Text classification has been around since the 1960s using a "manually defined set of rules, encoded expert knowledge on how to classify documents under the given categories" (Sebastiani, 2002). Since the 1990s, this method has lost popularity and turned towards a more machine learning-centric approach; currently, text classification borders between machine learning and information retrieval (or context-based document management) (Büttcher et al., 2010).

We explore and survey several machine learning algorithms in this thesis for our various experiments. The most common algorithms (and their variants) that we use are: Neural Networks, Naive Bayes, Support Vector Machines and decision-tree algorithms.

Neural Networks are "networks of non-linear elements, interconnected through adjustable weights." During the learning phase, the network adjusts these weights, on one or many hidden layers, so as to be able to correctly predict the output. They have been used in a wide variety of tasks such as handwritten character recognition and training a computer to pronounce English text (Nilsson, 1996). The Neural Network (NN) that we use is WEKA's (Witten and Frank, 2005) implementation Multilayer Perceptron which is good for noisy data. As well, it has the ability to classify patterns on which it was not trained but it requires long training times. As neural networks have been shown to work well with noisy data, such as the ones within sports, we includes these types of algorithms within our work. A visual depiction of a Neural Network can be seen at figure 2.1.

Naive Bayes comes from a class of Bayesian Classifiers which are statistical classifiers. They can predict class probabilities based on prior probabilities and the use of conditional independence. They have proven to do as well as decision trees and neural networks in specific classification tasks with high speed and accuracy. Naive Bayes is based on Bayes Theorem to estimate prior and posterior probabilities of the item under examination and the various classes in order to come up with final probabilities (Mitchell, 1997). We explore NaiveBayes, ComplementNaiveBayes, NaiveBayesMultinomial and NaiveBayeSimple, all different WEKA implementations. Bayesian Classifiers were included in our approach so we could determine our final predictions with a probabilistic approach. A visual depiction of the use of Naive Bayes to calculate prior probabilities can be seen in figure 2.2.

Support Vector Machines (SVMs) are the third type of Machine Learning algorithms that we explore. SVMs are one of the newer classification techniques which finds a linear

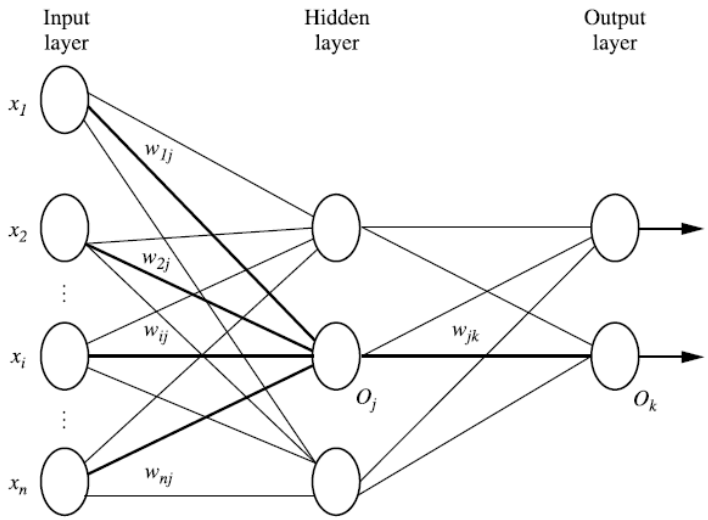

Figure 2.1: Visual Depiction of a Neural Network (Nilsson, 1996)

separator between a dataset. When the data is not linearly separable, it uses a nonlinear mapping to transform the data to a higher dimension. Their training times can be quite slow, but they are highly accurate because of their ability to find separations in nonlinear separation boundaries (Nilsson, 1996). Within WEKA there are two main types of SVMs that we explore: SMO (Platt et al., 1998) and LibSVM, two different implementations. We chose to include Support Vector Machine algorithms within our classification tasks as they have shown to perform well in the related works literature in previous tasks. A visual representation of an SVM finding the linear boundary between two classes from a set of data can see seen at figure 2.3.

Our final types of algorithm are rule-based algorithms which can conceptually be thought of as a series of IF-ELSE statements. One type of rule-based classification algorithms are decision trees; these tests are organized in a tree structure in which the internal nodes are tests on patterns within the data. For this work, we use the J48 algorithm which is the WEKA implementation of C4.5; in a C4.5 decision tree, each node of the tree chooses the attribute to split the data based on which attribute has the higher normalized information gain (based on Shannon's entropy from information theory) (Quinlan, 1993). We also look at algorithms such as JRip, Random Forests and Random Trees. We choose to explore these types of classification algorithms as their output is human-readable. A visual representation of a decision tree can be seen at

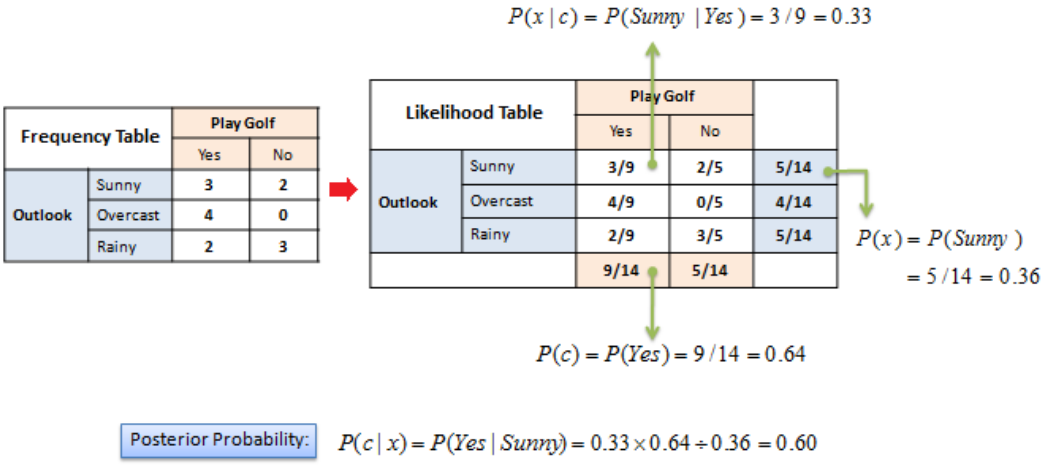

Figure 2.2: Example of Naive Bayes Prior Probabilities Calculations (Sayad, 2012)

figure 2.4.

One of the methods we explored in our work is a cascade meta-classifier which to the best of our knowledge has not been used in the context of sports predictions. A cascade-classifier is a multi-level machine learning algorithm which recursively applies machine learning algorithms on the output from one or many machine learning algorithms. In our case, we use a two-layer cascade classifier where our second level determines the final prediction based on input from three machine learning classifiers. Cascading classifiers were first used by Viola and Jones (2001) in the use of face detection in images. The idea of a cascade classifier is that the model makes its final decision based on the information collected from output of other given classifiers. It is a type of ensemble learning which combines the results from several machine learning classifiers.

Another method we use to analyze hockey, specifically the role that random chance plays in a single game, is the Monte Carlo method where we are able to simulate many schedules with random results and see how the results converge to the norm. The Monte Carlo method was first described by Metropolis and Ulam (1949) who explain it as "essentially, a statistical approach to the study of differential equations, or more generally, of integro-differential equations that occur in various branches of the natural sciences." Simply, it is a method of simulating the unknown using random values, repeated many times, and using the obtained probabilities distributions to estimate your unknowns.

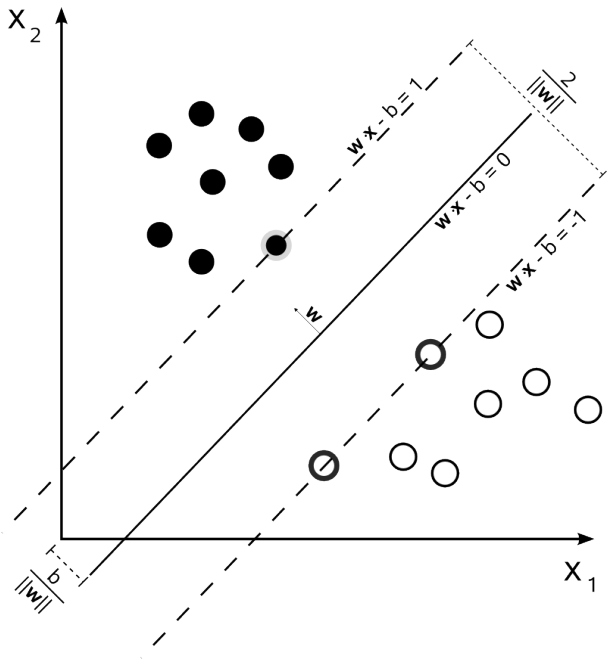

Figure 2.3: Visual Depiction of an SVM (Commons, 2009)

2.1.1 Textual Representation

There are a few different ways we can represent the textual documents as input features to our classification algorithms. We can model each document as a vector v in the t dimensional space R^t (Han et al., 2006). Using a Bag-of-Words approach, each term in the document can be represented with one of three possible values: binary, if the word is present or not; the term frequency, the number of times the term t is in the document d which is $freq(d, t)$; and by term frequency/inverse document frequency which multiples the term-frequency by the inverse document frequency by the inverse document frequency (IDF); $IDF = \log \frac{N}{DF}$ where N is the total number of documents and DF is the number of documents that contain the term t. Terms that appear in only a few documents tend to be more specific. (Han et al., 2006).

With a TF-IDF approach we compare the association of a term t in our document d to the scaling factor of a term t which looks how important a term is within a document compared to the collection of documents (Han et al., 2006). Together these are combined as shown in equation 2.1 with one variant of $TF(d, t)$ defined in equation 2.3 and one variant of IDF defined in equation 2.2

Figure 2.4: Visual Depiction of a Decision Tree (Sayad, 2012)

$$TFIDF(d,t) = TF(d,t) \times IDF(t) \tag{2.1}$$

$$IDF(t) = \log \frac{1+|d|}{|d_t|} \tag{2.2}$$

$$TF(d,t) = \begin{cases} 0 & \text{if } freq(d,t) = 0 \\ 1 + \log(1 + \log(freq(d,t))) & \text{otherwise} \end{cases}$$

Alternate methods to the bag-of-words/tf-idf approach of modeling the textual documents do exist and include word2vec, eigenword dictionaries and hashing. We chose to use this approach because of its success in the past as well as its ability to simplify word representation even though it disregards grammar and word order.

2.2 Predictions in Sports Analytics

2.2.1 Animal Racing

One of the earliest pieces of works in sports prediction using Machine Learning and Data Mining in sports came from Chen et al. (1994) who predicted the winners of greyhound racing. The authors used ten features on recent performances from each dog including the average time of their seven most recent races, best time, and win-percentage. The

authors modelled the data both with an ID3 decision tree and a neural network and compared their results to three experts. To analyze their results they bet $2 on each predicted winner and compared the payouts to the winnings of the experts. While their analysis only used a sample of 100 dogs to bet, the biggest issue with their work is that analysis by betting profit is not a good metric of forecasting performance. Betting odds are balanced by the bet-makers and which side people are placing bets on so that the bookmakers earn a profit; they do not necessarily accurately reflect the most likely winner.

2.2.2 American Football

Moving to American Football, data mining and machine learning have had plenty of uses in analyzing the sport. The biggest difference between the academic work relating to these two sports is that the machine learning approaches for American Football have been fairly similar. Most authors (Blaikie et al., 2011; David et al., 2011; Kahn, 2003) and (Purucker, 1996) have analyzed the National Football League (NFL) while others have looked deeper into the National College Athletics Association (NCAA) Football (Pardee, 1999) and Wilson (1995). Having only two leagues makes the results comparison much simpler. All of the methods that analyzed the NFL have strictly used Neural Networks without surveying of other methods such as Naïve Bayes, Support Vector Machines or Regression. David et al. (2011) took the most unique direction by using committees of Neural Networks to train on different random partitions of the dataset to find the best permutations.

The features used by the authors are fairly similar mainly focusing on the in-game statistics. Blaikie et al. (2011) looked at efficiency statistics such as yards per pass play, yards per rush play, points, fumbles per play and interceptions per play while David et al. (2011) added in additional features such as scoring, passing, fumbles and interceptions. Kahn (2003) looked at the differential between the two teams stats as well as adding in-game features for the home and away team and Purucker (1996) focused on the team's yards gained, rushing yards gained, turnover margins and time of possession.

Blaikie et al. (2011); David et al. (2011); Kahn (2003) and Purucker (1996) all used data within the same season. This can potentially lead to small sample size issues which can be seen in some analysis such as the work performed by Purucker (1996) who obtained varying levels of accuracy, between 69% and 81%, when testing on a single week of data. Blaikie et al. (2011) used the largest sample size in that he used all the data for the

seasons from 2003 to 2010 while testing on only the most recent season.

This small sample size can also be seen in the results of prediction from the experts. Blaikie et al. (2011) did not report accuracies for his model but rather the mean square error and stated that it was better than the Vegas bookmakers, a similar conclusion was reported by David et al. (2011). Kahn (2003) reported an accuracy of 75% in the 2003 week 14 and week 15 test data and Purucker (1996) reported accuracies of 62% and 78.5% in the 1995 week 14 and 15 test data. Pardee (1999) reports that experts in American Football are able to accurately predict between 62.5% and 75%.

The authors who used machine learning to analyze the NCAA were mainly focusing on determining which two teams are the best choices for the national championship game. NCAA football has a unique issue where the two best teams are selected by a combination of polls from sports writers, computer rankings, strength of schedule, and total losses: this can lead to bias issues and the picking of weaker teams. Both Pardee (1999) and Wilson (1995) used neural networks to analyze the NCAA to pick which team is the strongest. Delen et al. (2012) used Neural Networks in combination with Decision Trees, Support Vector Machines and Regression to predict the outcome of single bowl games. Using 8 years of data and 10-fold cross-validation, they were able to achieve an accuracy of 85% and found that the two most valuable features were: non-conference team winning percentage and the average margin of victory.

2.2.3 Basketball

Basketball follows a similar path as American Football in terms of data mining and machine learning. There has been some work in predicting single games using fairly similar features and algorithms. Unlike American Football, Basketball work has surveyed a vast array of different algorithms. Cao (2012) has applied Simple Logistic Regression, Support Vector Machines, Neural Networks and Naïve Bayes while others have focused on only one type of algorithms such as Loeffelholz et al. (2009) looking at Neural Networks, Miljković et al. (2010) dissecting the use of Naïve Bayes in this field and Yang and Lu (2012) using Support Vector Machines.

Features used are fairly similar to other sports with the in-game statistics that experts consider most relevant to winning. Some authors focused on league statistics such as the number of games in the last five days, rest days, performance over the season, total win percentage, home/road games win percentage, rest day win percentage and versus opponent win percentage (Cao, 2012) while Miljković et al. (2010) used many of the

standard basketball statistics such as field goals made, field goals attempted, 3 points, free throws, rebounds, blocked shots, fouls and more. The authors have used different features, Cao (2012) used the average of the last 10 game home and away statistics, Loeffelholz et al. (2009) used the season to season statistics, splitting home and away while Yang and Lu (2012) only looked at the teams in the post-season.

The accuracies were fairly similar despite the variety of methods taken to predict a game. Cao (2012) reported an accuracy of 69.67% while Miljković et al. (2010) cites that experts were able to predict at 68.87% and he was able to improve upon that with an accuracy of 74.33%. Yang and Lu (2012) claim 86.75% accuracy with their linear kernel SVM. They also made the interesting observation on the large role of randomness within the game that makes it very difficult to predict. When they predicted the winners of the fifteen post-season best-of-seven series, they were only able to accurately predict at 55%. They cited that because of the randomness, the best team could be knocked out of the tournament in the first round.

Predicting the National Basketball Association's post-season championship was a similar task as performed by Wei (2011). Using Naive Bayes and learning from the home and away statistics during the 2005/06 to 2010/11 season, the authors were able to predict at varying levels of accuracies for each year, for the 15 series. The median accuracy for the five post-seasons were 73.3% and the authors predicted that the Chicago Bulls had the highest probability of winning that year's post-season championship over the San Antonio Spurs. In reality, Chicago ended up losing in the Semi-Finals while the San Antonio Spurs did not advance past the first stage, which can show how better teams do not always wins.

2.2.4 Hockey

To the best of our knowledge, there is no research in Machine Learning and predictions within hockey. There have been uses of algorithmic and statistical models to analyze other sports including single game prediction, long term predictions and non-game related predictions.

Brimberg and Hurley (2009) questioned if referee's behaviour is markovian, that is, they remember previous penalties and take these memories into consideration when making future decision. The authors found that if a referee has given one team the first two penalties, they are highly unlikely to give a third one suggesting that referees even out the number of penalties given. Other authors have looked at various areas within hockey

using data mining and Machine Learning to analyze rates at which teams score and yield goals (Buttrey et al., 2011), and data-mined hockey data to analyze individual player contributions to their team (Hipp and Mazlack, 2011). Statistical models in hockey have been created to analyze how a team reacts when the opposing team scores first (Jones, 2011; Swartz et al., 2011), expectation models were developed for goals based on traditional NHL data (Macdonald, 2012) and decision trees were created to explore the attacker-defender interaction in sports to predict the outcome (Morgan et al., 2013).

2.2.5 Soccer

While many other sports have been analyzed in predicting single games using data mining and machine learning. Soccer is a very popular sport for this application. To make predictions within the sport of soccer, Aslan and Inceoglu (2007) used a neural network for their algorithm of choice. They use four features as input, one for both teams' home and away rating (how well the teams perform when playing at their main arena compared to other arenas). They create two models using two permutations of these for the Italian Serie A 2001/2002 series and were able to obtain accuracies of 51.29% and 53.25%. Both are lower than the baseline of the historical home win% of 54% but the authors show it is an improvement over using Elo, a chess ranking rating system, which obtains an accuracy of 47.71%.

Joseph et al. (2006) used Bayesian Nets to predict the outcome (win, loss, or draw) of individual soccer games for a specific team between 1995 and 1997. They used very specific player features such as if certain identified players are in the line up and what position they are playing. The model was able to predict at 59.21% accuracy and was able to outperform other Machine Learning algorithms: MC4 (decision tree learner), Naïve Bayes, Data Driven Bayesian and K-Nearest Neighbours. A big issue with the presented model is that it was trained for a specific team and is not adaptable to other teams even within the same league and time frame.

In early soccer data mining work performed by Nunes and Sousa (2006), the authors attempted to both look for interesting patterns within the 2005/2006 Portuguese Championship football league as well as predicted the outcomes of the individual matches using the C4.5 decision tree. Features used were based on cumulative goal data such as goals scored and scored against. The apriori algorithm did not find anything unexpected within the data. The decision tree was able to predict at 59.81% accuracy, approximately a 5% increase on the baseline with the home team historically winning 54% of matches.

Since then there has been plenty of other work from researchers in soccer predictions including (Buursma, 2010; Constantinou et al., 2012; Constantinou, 2012; Hoekstra et al., 2012; Hucaljuk and Rakipovic, 2011; Tucker, 2011). A large number of different algorithms have been surveyed by these authors. Buursma (2010) surveyed the a vast array of algorithms, including NaiveBayes, Logitboost, RandomForests, Linear & Logistic Regression, and largest-class classifier to predict outcomes of games in the EPL and the Dutch League. Algorithms focused on others were Constantinou et al. (2012) and Constantinou (2012) using Bayesian Networks on the English Premier League (EPL). Hoekstra et al. (2012) (hoekstra) uses k-NearestNeighbours and an evolutionary ensembler classifier for the Dutch Eredivisie, while Hucaljuk and Rakipovic (2011) surveys all categories of classification algorithms and Tucker (2011) (tucker) used Neural Networks (the Multilayer Perceptron implementation).

Features used by soccer predictors have generally been similar across most works. A common feature in soccer is what is known as "form" and is a single variable representation of how the team has performed in the last n games. Buursma (2010) compared a teams strength of schedule and form from the previous 20 games and the entire season. Other similar features that have been used include the strength of the team, based on performance over the season (Constantinou et al., 2012; Constantinou, 2012), the distance between the stadiums and the number of cup matches the teams have played in the previous week (Hoekstra et al., 2012), the team's current ranks in the league, how the two teams performed against each other in their last meeting, the number of injuries on the team, and the cumulative goal statistics (i.e., goals scored and scored against) (Hucaljuk and Rakipovic, 2011) and Tucker (2011) used form over the previous 9 games.

Despite the wide range of algorithms and features that have been used, the reported accuracies seem to fall within a similar band between 50 to 60%. Buursma (2010) was able to predict at 55% with their Bayesian network, while the evolutionary ensembler of Hoekstra et al. (2012) predicted between 57% and 59%. Hucaljuk and Rakipovic (2011) reported a best accuracy at 68%, though that may be high given that the dataset suffers from small sample size issues as they test and train on a total of 96 matches which is the entire 2011 season. The Neural Network of Tucker (2011) was able to achieve 47% accuracy and it appears that despite the diversity of the methods, the results for soccer are being reported around the same accuracies, with slight variations depending on the league and of the year their data.

What appears to be an issue from the soccer literature is that plenty of works, such as (Buursma, 2010), (Constantinou et al., 2012) and (Constantinou, 2012) are comparing

their models to if they are profitable when using the results for predictions. Constantinou et al. (2012) suggests that whether their model is profitable is their main method for evaluating their predictions. Similar to the issues that were discussed with Chen et al. (1994), this is not a reliable method to compare prediction models against betting odds, as betting odds do not accurately reflect the true likelihood of winning, but rather a balance between what people are betting on so that the bookmakers can earn a profit.

2.3 Long-Term Predictions

Research has been performed in using machine learning and data mining techniques for long-term predictions in sports. Huang and Chang (2010) and Huang and Chen (2011) presented a multi-stage Neural Network to predict each of the stages within the 2006 FIFA World Cup. At each stage of the tournament, they retrained their model on all previous data and tested on the next stage. In the FIFA World Cup tournament, it is a single knock-out game where draws are not possible and this allowed the authors to achieve an accuracy of 76.9%.

Sardar and Salehi (2012) used neural networks to predict the FIFA ranking of the Iranian National Football team. They collected features such as the month to month change in Iran's FIFA ranking over the previous 120 months as well as their month's results of games including the summary of in-game statistics such as goals scored and scored against. They were able to create a Neural Network which produced predictions that had an R^2 correlation of 0.999972 with the real world results.

2.4 Statistical Modeling in Sports

2.4.1 American Football

In American Football statistical models have been used to evaluate how well prediction models performed compared to experts, to the betting market and to other popular methods. Boulier and Stekler (2003) compared power scores, the relative ability of teams based on objective criteria, with probit regression versus the opinions of experts, naïve models and the betting market and found that the betting market is the best predictor followed by the probit predictions model (where the dependent feature can only be a binary value). Herda et al. (2009) evaluated the success of an NCAA football team's recruiting and whether it leads to more wins in the future. Using corresponding

predictive indices (R^2) and Pearson product moment correlation coefficients, the authors found that their model can predict 45% of the variance of wins. The worst accuracies of predictions come from the opinions of experts which are comparable to bootstrapping and are inferior to naive forecasts. Song et al. (2009) looked at consensus within binary game prediction forecasts using Cohen's kappa coefficient and found that forecast systems are better than the betting line and that the higher the consensus amongst the models, the higher the confidence of predictions.

2.4.2 Baseball

In baseball, statistical models have been used to forecast single games and to make long term predictions. Barry and Hartigan (1993) used a choice model with markov chain sampling combined with features such as the strengths of the teams and home advantage. The authors used this to predict outcomes of single games in order to make long term predictions for division champions in the 1991 Major League Baseball Season. Also predicting single games was Miller (2011) who surveys a number of regression models with teams offensive productions, pitching, defensive abilities, past game outcomes and previous win percentages.

2.4.3 Basketball

Basketball research is much more focused on the collegiate portion of the sport, specifically on making predictions within the 64 team single-game elimination-tournament to determine the national champion. Carlin (1996) used prior probabilities and regression techniques combined with data on team's relative strengths and point spreads to produce the probabilities of each team emerging as the regional champion in the tournament. Schwertman et al. (1996) used both linear and logistic regression models combined with seed positions ("ranking") in the NCAA tournament to predict the probability of each of the 16 seeds, for all 4 regional tournaments, to emerge as the regional champion and to eventually become the national champion. Plenty of works have looked at the value of the different seedings within the tournament. Smith and Schwertman (1999) continued this trend by using regression models for tournament games and demonstrated a strong relationship between seed position and actual margin of victory and further demonstrated that the seed itself can be used as a reliable prediction method alone.

2.4.4 Soccer

While algorithmic modelling has been used for predictions of games, traditional statistical based models have played a large role in analysis of nearly every sport. While there are too many statistical modelling works to list them, we present a few from each of the major sports. In European Football Min et al. (2008) used a rule-based reasoner in co-operation with a Bayesian network component and features that included current game scores, morale, fatigue, and skill to predict the results of matches. Van Calster et al. (2008) analyzed scoreless draws in matches using Bayesian networks, least squares Support Vector Machines, and hybrid Monte Carlo multi-layer perceptrons. Using features which include in-game statistics such as goals per game they found that better teams have less scoreless draws. In the English Premier League Vlastakis et al. (2008) used Poisson count regression and compared it to nonparametric Support Vector Machines in combination with betting odds to predict match outcomes.

2.4.5 Tennis

While tennis has little known research in terms of algorithmic modelling plenty of work has been invested in statistical models of the sport. McHale and Morton (2011) used logistic regression ("logit") models combined with players official rankings and official ranking points of the two competing players to forecast the outcome of the match and were able to produce superior predictions compared to betting returns. McHale and Morton (2011) used the logit model to calculate the probabilities of both players winning the match both before and during the match itself at match and point level. Newton and Aslam (2009) used a much more elaborate system to predict winners of matches and simulated entire tournaments. The authors modelled the probability of a player winning a point on a serve and receiving a serve as a Gaussian distributed random variable. Combining this with standard deviations of a player's consistency from match to match and on different playing surface, the authors used a stochastic Markov chain model and obtained the probability density function for a player to win a match.

2.5 Non-Statistical Predictions/Analysis in Sports

2.5.1 American Football

Sinha et al. (2013) took the unique view of mining tweets during the NFL season and used data mining and machine learning to find patterns within the tweets to be able to learn which team will win. They also looked at predicting based on betting outcomes to beat the point spread. They trained and tested on the same season of data, but their test sample size was fairly small, looking at single weeks at a time. Their findings indicate that they can match or exceed the performance of traditional methods of predicting with in-game statistics.

2.5.2 Baseball

Baseball has not yet used data mining for prediction of games but they have used machine learning algorithms to predict which players will be voted into the Hall of Fame (Freiman, 2010; Mills and Salaga, 2011; Young et al., 2008), future player performance (Lyle, 2007) and end of year award winners (Smith et al., 2007). Freiman (2010) used random forests to be able to identify 75% of Baseball Writers Associations of America Hall of Fame selections. Mills and Salaga (2011) used traditional in-game statistics such as batting averages, home runs, runs batted in, hits, etc. with tree ensembles and obtained an error rate of 0.91%. In a similar project. When predicting players who will be voted into the Hall of Fame, features such as the career offensive and defensive statistics and total number of end of year awards have been used, in combination with neural networks, to achieve an 98% accuracy (Young et al., 2008). This high accuracy may be due to the low number of players who are selected to enter the Hall of Fame.

To complement Hall of Fame selection votes, Smith et al. (2007) examined the use of Bayesian classifiers to predict which starting pitchers will win the Cy Young Award, the award given to the MLB's best pitcher. Using in-game statistics from pitchers between 1967 and 2006, they were able to predict with 80% accuracy. Smith et al. (2007) used ensemble learning to predict how players performance will change based on the previous 30 years of in-game statistics.

2.5.3 Basketball

Halberstadt and Levine (1999) and Heit et al. (1994) ran similar experiments with novice

predictors for the NBA and NCAA. Heit et al. (1994) found that novices who receive some training on the relative strengths of teams, and provide feedback on results of matches the predictors initially guessed, can be trained to guess as accurately as the experts. Halberstadt and Levine (1999) found that by making people list their reasons for predictions, they perform worse than their counterparts who were told to not analyze their reasons. This might be because the reasoners either access information that is not appropriate or they have difficulty comprehending and computing the information they bring forward.

Basketball analysts have employed data mining to find interesting patterns within the game of basketball. Ivankovic et al. (2010) used neural networks as well as various in-game events and players' statistics to analyze the game. They found the most important elements that lead to winning are two-point shots under the hoop and defensive rebounds. Poualeu et al. (2011) explored the use of data mining on the events in the game, such as shots made and missed shots, in combination with the location of the players and the ball to build up regulars patterns that can be defined by team, player and year. The authors used these to predict various events within a game. Markoski et al. (2011) determines the best location for a referee to be located to gain the best vantage point of the game by using neural networks in combination with features on the location and the movement of the ball. Santos et al. (2011) combined neural networks with genetic programming to analyze team performance in the Spanish basketball league using the in-game team statistics.

2.5.4 Soccer

Other researchers have looked at the ability of people to make predictions, from a psychology and cognitive science perspective, from making predictions in the 2006 FIFA World Cup (Andersson et al., 2009), 2002 FIFA World Cup (Andersson et al., 2005) and matches within the top Dutch League (Dijksterhuis et al., 2009). Unsurprisingly the authors found that experts are able to make more accurate predictions and with more confidence than novices, but experts are not able to outperform the baseline of using ranking systems (Andersson et al., 2009, 2005). They have also found, with Unconscious-thought theory, that experts can make better predictions unconsciously than if they are forced to think about their decisions (Dijksterhuis et al., 2009).

2.5.5 Tennis

Scheibehenne and Bröder (2007) and Serwe and Frings (2006) looked at the value of lays persons and amateur tennis players making predictions for the 2003 and 2005 Wimbledon Tennis Grand Slam. They found that by selecting winners of matches with the use of recognition heuristics they were able to correctly predict 70% of all matches. This may be because the better players are typically well known and end up playing in more matches in the tournament. Scheibehenne and Bröder (2007) found that when the predictors did not recognize either player, their selection accuracy dropped, becoming closer to a coin flip. The overall accuracy was better than selection based on rankings and the seedings by experts, but did not beat selecting by betting odds.

2.6 Problems To Address

What seems common amongst other predictions in sports and data mining work is that the authors have mainly focused on using machine learning algorithms from a black box perspective, which is simply applying their algorithms on the data with minimal tuning. The approach that many authors use is to treat prediction of sports as simple classification problem and try to solve this by surveying a large number of algorithms. While this is a good place to start further work and different approaches needs to be implemented.

The most commonly used features are the in-game statistics that experts consider to be most representative of the sport. Outside of traditional in-game statistics the most commonly used feature has been "form" to describe how a team has recently performed. Other features should be considered and explored than the most obvious in-game data. There has been little work with text to predict sports and to the best of our knowledge there are no unique methods used for prediction.

We also note that sports predictions within the same sport seem to have similar levels of predictions and there appears to be a ledge that the authors are not able to achieve an accuracy higher than. This difficulty needs to be addressed or to be explained as what the cause is.

Further, while the low sample size of some of the training data is as a result of training on a single season, where possible a larger set of data needs to be included. Single season may produce odd results which could be the cause of seemingly low prediction results, there may be a case of concept drift suggesting that a single season of training data is

the best solution.

Finally some of the previous works use less than ideal methods to evaluate the performance of their classifier such as if it is able to produce a profit. Using betting results, and if or how much of a profit they earn, is not likely to be the most appropriate method of evaluating a classifier. Odds are often based on what people are betting on so "the house" can earn a profit.

2.7 Hockey Performance Metrics

In this thesis, we look at a sport that, to our knowledge, has a limited analytics background. We start by making predictions of single game wins and losses using those traditional in-game statistics that the experts consider to be most valuable. We also explore the value of using the "advanced statistics" in the use of predicting a single game, as they are found to be better correlated with wins over the long term.

Two of the advanced features we look at are puck possession and PDO.

Dermody (2013) performed a well detailed statistical study on PDO. PDO is not an acronym but is rather the internet username of the user who created it. PDO is the summation of a team's on-ice shooting-percentage (sh%) and save-percentage (sv%). Over time the PDO of a team will regress to 100%. While Dermody dives in much greater detail on why this happens, a brief way to explain it is that PDO converges towards 100% because the relationship between Sh% and Sv% is a zero-sum game. As teams score and take shots at the opposing team, this has an effect on their own sh% while inversely effecting the opposing team's Sv%. Over a season there has yet to be a team that has sustained a shooting or save percentage extremely different than the league average. Because of the relationship between these two features, over the season they have consistently demonstrated that they regress to 100%. In the the short term we can use PDO to see if a team has been "lucky" by scoring more often than the norm (a high sh%) while stopping more pucks than the average (a high sv%). The opposite can be true as well, with an "unlucky" team not being able to score themselves while allowing the other teams to score against them frequently. As the author demonstrated, by the end of the season (82 games) all teams will have a PDO of 100% +/- 2%.

Internet analysts are able to make long term forecasts on a team by combining their puck possession data with their PDO. Teams with strong possession skills that have been deemed "unlucky" will likely regress and start to win more frequently. The reverse has also held true where teams with weak possession abilities but high PDO should expect

to regress to the norm and not perform as well. This was demonstrated with two well-known case studies in the NHL: the 2011/2012 Minnesota Wild and 2013/2014 Toronto Maple Leafs. Both teams had very low possession values, a very high PDO, and were ranked first place in the league two months into the season. Analysts predicted both teams would regress and lose places in the standings. By the end of their respective seasons neither team qualified for the post-season championship tournament.

Ferrari (2008) made one of the first big advancements in hockey analytics by realizing that the summation of attempted shots for (represented as a percentage) correlates to time in the offensive zone which over the long term leads to winning. Teams who have the puck more often are able to score more often, and prevent goals from being scored against them. There are a few variations of possession called Corsi and Fenwick, respectively, which are calculated differently but represent the same idea. Nikota (2013) demonstrated that using a stopwatch to calculate the time a team spends in the offensive zone is nearly identical to their Fenwick possession rating. These possession statistics can be further derived to player level statistics such as Corsi Relative (how well the team's possession is with the player on versus off the ice).

We further explore the idea of a "glass ceiling" where hockey and other sports appear unable to achieve an accuracy higher than a fixed value. We dig into this issue by analyzing the portion of games where random chance determines the outcome using a Monte Carlo simulation. We further analyze making predictions over a longer series of games with the best-of-seven post-season series to determine the champion of the year. We then further refocus on predicting a single game using a novel application of meta-classifiers using text from pre-game reports written by experts combined with the traditional in-game statistics that we found worked best in chapter 3 and with sentiment analysis (how "positive" or "negative" the text is).

Chapter 3

Single Game Prediction

3.1 Introduction

In this chapter, we analyze the ability to apply Machine Learning algorithms on hockey data in order to forecast the winner of each game. For this chapter, we used machine learning techniques applied on twelve in-game statistics for features (the input vector (Nilsson, 1996)): traditional statistics (9) and performance metrics ("advanced statistics") (3) which have been demonstrated to be correlated with success in the long run. We analyze their effectiveness for prediction. This chapter was presented at the European Conference on Machine Learning: Machine Learning and Sports Analytics Workshop (Weissbock et al., 2013) and is built upon in future chapters.

3.2 Data

We collected data before and after every NHL game that took place between February 16th, 2013 and April 28th, 2013 for a total of 517 games of 720 games (71.8%) of the shortened 2012-2013 NHL season due to the labour lockout (normally there are 1230 games per year). While a larger sample size of data may improve our accuracy, our work on testing and training over multiple seasons, as further explored in Chapter 4, suggests this lowers the overall accuracy. As we were using many performance metrics for this experiment it is not possible to go back and collect data for some features, as there are no historical records. Fan websites and third party databases only provide the latest values of these metrics. A Python script was used to automate the daily collection of statistics, but regular verification was required by humans to make sure the data was

collected appropriately.

A list of all of the features used in this chapter can be seen in table 3.1. Only the features and values that are known before the game commences are used in this experiment. Additional features collected after the games allowed us to calculate further statistics such as the cumulative goals for and against and PDO, which were used as features in future games.

Statistics in hockey are often divided into two different groups by hockey analysts: traditional statistics and performance metrics. Traditional statistics are those statistics which have been well established within the NHL. These statistics are published and tracked by the league and are often cited in the main stream media reports of the game. Traditional statistics, such as Goals For, Goals Against, etc, are often based on goals or simple counting, which can lead to a small sample-size problem. The major concern with these statistics is there is a low correlation with wins and points in the long term standings of the games, and in some cases, a negative correlation (Charron, 2013).

The other issue is there is a demonstrated bias in the league's statistical trackers at each arena. What might be considered a hit at Madison Square Garden in New York City might not be the same as at the Staples Center in Los Angeles. When comparing team's stats between their home arenas (teams play 48 games in a single location) versus the other 29 arenas, there is almost no correlation between the two for traditional statistics.

Hockey analysts within online communities have attempted to deal with problems by researching and analyzing the games to find statistics and approaches that don't have these problems. The downfall with these statistics is they are often published by third party websites; casual hockey fans do not learn these statistics as quickly. Also the main stream media has been slow to use and broadcast them, and they can require much more advanced mathematics to calculate.

Advanced statistics are usually shot based; this gives a much larger sample size and helps eliminate the bias in the league trackers. As a result, they have a much higher correlation, compared to traditional statistics, with wins and points over the length of the season (Murphy, 2013) as seen in table 3.2. Many derivatives and variations of these advanced statistics exist at the individual and team level. The most common advanced statistic is a measure of possession, known as Fenwick or Corsi (depending on how the number is calculated), which uses the number of shots on goal, shots attempted, and blocked shots. Representing it as a percentage between the two teams, it has been shown to correlate to time in the offensive zone (Ferrari, 2008).

In this experiment we used a total of 12 features to model each team before a game.

Goals For	(Traditional) The cumulative goals scored by a team in the season up to the current game.
Goals Against	(Traditional) The cumulative goals scored against a team in the season up to the current game.
Goal Differential	(Traditional) The differential between the cumulative goals scored for and against a team in the season, up to the current game.
Power Play Success Rate	(Traditional) The percentage of power play attempts where a goal was scored by the team.
Penalty Kill Success Rate	(Traditional) The percentage of penalty kill attempts where your team concedes a goal.
Shot Percentage	(Traditional) The ratio of goals scored to shots on goal by team or player. League average is 8-9%.
Save Percentage	(Traditional) The ratio of goals scored against to shots on goal against faced by a goaltender or a team. League average is around 91.9%.
Winning Streak	(Traditional) The number of consecutive games won by a team.
Conference Standings	(Traditional) The ranking of a team in their conference based numerically in descending order on the number of points earned (a function of wins and overtime and shootout losses).
Fenwick Close %	(Advanced) A proxy statistic for time in the offensive zone for teams in score close situations.
PDO	(Advanced) A measure of random chance by the addition of Shot Percentage and Save Percentage. Over time regresses to 100%.
5v5 Goals For/Against	(Advanced) The ratio of goals scored for and against a team during situations where both teams have 5 players on the ice.

Table 3.1: Features used in the single-game prediction model.

Table 3.2: Advanced vs Traditional Statistics

	Type	vs Wins R^2	vs Points R^2
5v5 Goals F/A	Advanced	0.605	0.655
Goals Against / Game	Advanced	0.472	0.51
STI	Advanced	0.372	0.39
Goals/Game	Advanced	0.352	0.36
Sv%	Traditional	0.227	0.263
PP%	Traditional	0.221	0.231
Shots Against / Game	Traditional	0.198	0.191
Shots / Game	Traditional	0.17	0.203
Shot %	Traditional	0.16	0.145
PK%	Traditional	0.152	0.16
Faceoff%	Traditional	0.097	0.109

As listed in table 3.1, the traditional statistics we used are: Goals For (GF), Goals Against (GA), Goal Differential (GlDiff), Power Play Success Rate (PP%), Penalty Kill Success Rate (PK%), Shot Percentage (Sh%), Save Percentage (Sv%), Winning Streak, and Conference Standing. Statistics collected after the game included which team won and lost each game, the score, and shots for and against each team; these statistics are readily available from `http://www.NHL.com` and `http://www.TSN.ca`. This gives us the ability to calculate further statistics such as estimated-possession over a smaller subset of games rather than the season total and averages. An example of the data for 6 games can be seen in table 3.3.

The advanced statistics that were collected are Fenwick Close %, PDO and 5-on-5 For & Against. Fenwick in "Close" situations refers to the first and second period, when the teams are within 1 goal, and the third period when the score is tied. This is to remove "Score Effects" from our statistics where a team who is losing plays much more risky strategies to catch up. These statistics are not available from the NHL league's website, so we must collect them from third party websites such as `http://www.behindthenet.ca`. These statistics were validated daily to ensure that they were collected properly, as there is no historical record.

PDO is a statistic of random chance ("luck"), a subject that is covered further in chapter 4. PDO is not an acronym but rather named after the Internet username of

the person who created it (obscure naming conventions in advanced statistics are a known issue within the analytics community). PDO is the summation of a team's Shot Percentage and Save Percentage. Over a season, PDO will regress to $100\% \pm 2\%$ for all teams (Dermody, 2013) as players have not demonstrated the ability to maintain a high shot percentage for long periods. If a team has a PDO higher than 100%, then they are deemed "lucky"; conversely, if they are below 100%, they are deemed to have been "unlucky". We can use this to forecast if a team's success in the short term will continue, slow down or become better in the long term.

Hockey appears to have stochastic processes play a much larger role than other sports. Due to random chance, a single goal can cause a weaker team to beat a stronger team. Pucks can bounce off of players, sticks, the goal posts, the boards, etc., which can have a negative or positive impact for a team. While over a season random chance evens out for all teams, in the shorter term we can use PDO to see if teams have been performing better or worse than their true ability.

Team	Location	Fenwick Close %	Gf	GA	GlDiff	PP%	PK%	Sh%	Sv%	PDO	Win Streak	Conf. Standing	5-5F/A	Label
Toronto	Away	44.92	108	100	8	18.7	85	892	919	1027	2	6	1.05	Win
Ottawa	Home	49.85	89	72	17	29.8	89.4	929	939	1010	3	5	1.12	Loss
Minnesota	Home	48.47	119	126	-7	17.6	80.6	921	911	990	-1	8	0.88	Win
Colorado	Away	46.78	115	149	-34	15.1	80.6	926	909	983	1	15	0.83	Loss
Chicago	Home	55.91	154	99	55	16.9	87	906	928	1022	2	1	1.57	Loss
St. Louis	Away	53.89	126	114	12	19.7	84.5	921	910	989	2	4	1.01	Win
Vancouver	Home	51.8	125	114	11	15.4	84.2	914	928	1014	-1	3	1.09	Loss
Edmonton	Away	44.18	118	132	-14	20	83.7	921	921	1000	1	12	0.84	Win
Phoenix	Home	50.25	120	128	-8	15	80	924	927	1003	-1	10	1.05	Win
Anaheim	Away	48.13	137	113	24	21.4	81.2	904	930	1026	3	2	1.3	Loss
San Jose	Home	53.03	122	113	9	20.1	84.9	934	928	994	-1	6	1.01	Loss
Los Angeles	Away	56.98	130	116	14	20	83.2	921	912	991	-2	5	1.08	Win

Table 3.3: Example data for 6 games.

3.3 Method

For this experiment, we use WEKA (Witten and Frank, 2005), a tool that provides many machine learning algorithms and was employed for all classification tasks. Pre-processing of the data was done during all stages of collection as discussed in the previous section.

For each of the 517 games, we had two data vectors of features (a total of 1034 vectors), as seen in table 3.3, one for each team. The winning team, in vector V_1 receives the label "Win" and the losing team, with the vector V_2 receives the label "Loss". The differential of these two vectors were then fed into the algorithms. Since we created two vectors for each game, there is a 50% chance of guessing he outcome each game correctly. Additionally, we modelled the experiment in a single training example for each game (e.g., $V_1 + V_2 + label$ with the label either "HomeWin" or "Away Win"), these results did not differ from the presented results.

Using the above mentioned representation we ran two experiments. First, we looked at how effective traditional, advanced and mixed (both combined) statistics are for forecasting success in the NHL. Secondly, we further analyzed the "luck" feature to see if it can further improve the accuracy of our predictions.

We consider this to be a binary classification problem, where algorithms will learn from the data and make a decision from one of the two possible classes: win or loss. We surveyed a number of algorithms in WEKA and used 10-fold cross-validation to test, where we divide the training set into 10 equal-sized subsets, and for each subset, we test on one after training on the union of the other 9. We repeat this a total of 10 times and then our accuracy is the average of the 10 test units (Nilsson, 1996).

In our first experiment we surveyed four different types of algorithms using their default WEKA parameters. The algorithms are: WEKA's implementation of a Neural Network (NN) (Multilayer Perceptron) which is good for noisy data; Naïve Bayes (NB) for a probabilistic approach; SMO, WEKA's Support Vector Machine implementation as it has shown to do well in previous classification tasks; and, J48, WEKA's implementation of the C4.5 decision tree as it produces human-readable output.

In our second experiment, instead of using the season average of PDO (the feature representing how "lucky" a team has been), we used an n-game rolling average for PDO, that is the average of a team's PDO for the last n games (for values of 1, 3, 5, 10, 25 and all). As PDO will regress over a large number of games; this gives us the opportunity to see if how "lucky" a team has been is a good predictor of how well they will do. The results for this can be seen in table 3.6.

3.4 Results

The accuracies of these algorithms based on the three datasets for the first experiment can be seen in table 3.4.

	Traditional Statistics	Advanced Statistics	Combined ("Mixed") Dataset
Baseline	50.00%	50.00%	50.00%
SMO	58.61%	54.55%	58.61%
NB	57.25%	54.93%	56.77%
J48	55.42%	50.29%	55.51%
NN	57.06%	52.42%	59.38%

Table 3.4: Accuracies of the first experiment for 10-fold cross-validation.

The highest accuracy achieved was from the Neural Networks algorithm at 59.3%. No further tuning of any of the algorithms was able to produce an accuracy higher than this. When using a meta-classifier that used majority voting from the Neural Network, SMO and Naïve Bayes algorithms this accuracy was further increased to 59.8%. Additional ensembler learners, such as stacking and voting with 5 classifiers, were explored but they did not yield statistically different results.

As a side note, we explored the use of gambling odds in this experiment to learn "from the crowd". These features did not affect our overall accuracy, in a negative or positive way. Based on our further explanations, as seen in chapter 4, we hypothesize that the manually produced gambling odds are not able to improve machine learning within sports.

We explored the statistical significance of these results. When comparing traditional features to the baseline all algorithms outperformed the baseline (50% for this experiment) and the improvement was statistically significant. When using only advanced statistics as features, none of the algorithms were statistically different than the baseline. When combining the two datasets in the mixed features, all classifiers except for J48 were better than the baseline. We also found no statistically-significant difference between the Neural Networks and the SMO algorithms in the results.

For further analysis, we split the data into 66% for testing (341 games) and 33% (171 games) for training. This allows us to determine the final prediction for each test case. This method of testing returned an accuracy of 57.83%. We used the outputted prediction to look at each pair of predictions for each game, as each game had two data

vectors, for additional tuning. Since it is not possible to have a game where the outcome is the same for both teams (i.e., Win/Win or Loss/Loss) we looked at all test cases where the predictions where the same. In this situation, for the feature vector with the lowest confidence (probability of being correct), the label was inverted. By doing this, the overall accuracy, for the 66%/33% test/train data-set, increased to 59.0%.

We analyzed the results of this model with the WEKA Consistency Subset Evaluation (CfsSubsetEval) feature selection to see which features contribute most to the overall learning. CfsSubsetEval "evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them" (Witten and Frank, 2005). There are many attribute evaluators to chose from within Weka and elsewhere such as single-attribute evaluators, attribute subset evaluators and more. We chose CfsSubsetEval as we wanted to identify the features that are highly correlated while maintaining a low inter-correlation. From this, the three most valued features are: location (Home or Away), Goals Against, and Goal Differential. This is surprising to us given that advanced statistics have been shown to be more successful in the long term (Charron, 2013; Murphy, 2013). When recreating the same model in this chapter using only those three identified features, the results of its prediction were not statistically different ($p = 0.59$) than when using all 12 features in the 66%/33% train/test model.

The ROC curve for this model can be seen at figure 3.1. An ROC curve stands for: Receiver Operating Characteristics. An ROC curve allows us to see the trade off the 10-fold cross-validation model is making between accurately predicting "yes" for a class versus how often it predicts "no" cases as a "yes" (Han et al., 2006). The area under this curve is one measure of accuracy for a model. A perfect accuracy will yield a ROC Curve of 1.0 while 0.5 is a model on the other end of the spectrum (Han et al., 2006). For this model, our ROC Curve is 0.6037 and like the overall accuracy suggests, it is an improvement from the baseline, but its overall accuracy is not very high.

To further analyze this model, we can look at its precision, recall and F-Score of the weighted average of both the Win and Loss class. These can be seen in table 3.5. Precision is the percentage of retrieved instances that are relevant (i.e., "correct" responses) (Han et al., 2006). Recall is "the percentage of documents that are relevant to the query and were, in fact, retrieved" (Han et al., 2006). F-Score is "defined as the harmonic mean of recall and precision" which "discourages a system that scarifies one measures for another too drastically" (Han et al., 2006). Alternate methods to these accuracy measures exist such as absolute error, squared error, mean/relative absolute/squared error. These four

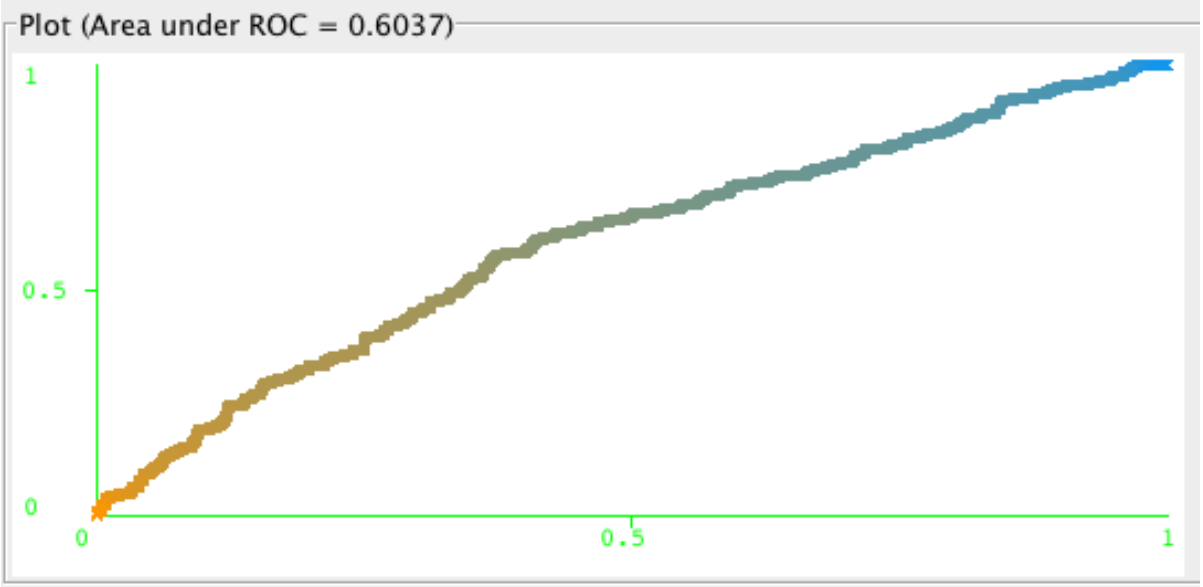

Figure 3.1: ROC Curve of tuned Neural Network on the mixed dataset

methods were chosen as they have shown to do well in classification tasks such as ours. We can see in this table that the results from this model are fairly balanced between the precision and the recall.

When performing error analysis on the predictions of this model, we look at both the ROC Curve and the Precision/Recall/F-Score values. While these values all suggest that we are able to improve from the baseline, we have yet to achieve a high accuracy. While random chance could be playing a role within the sport, there are a few issues that are causing the lower accuracy. One issue is that we are only looking at a subset of data from

	Precision	Recall	F-Score	ROC Curve
Baseline	0.500	0.698	0.581	0.500
SMO	0.586	0.586	0.586	0.586
NB	0.567	0.571	0.569	0.588
J48	0.558	0.534	0.545	0.537
NN	0.583	0.594	0.588	0.604

Table 3.5: Breakdown of each classifier on mixed data for the first experiment using 10-fold cross-validation.

one season. The lower accuracy could be due to the variance within the small sample size or because the data that we have does not properly represent true NHL values.

Additionally the features that we have selected may not be the best features to be able to predict a single game. The advanced statistics have been shown to do well in the long term, so it was surprising they did not work for predicting a single game. The traditional values were focused on goal based data which is not as predictive as shot-based data which might lead to better results.

	PDO1	PDO3	PDO5	PDO10	PDO25	PDOall
Baseline	50.00%	50.00%	50.00%	50.00%	50.00%	50.00%
SMO	58.61%	58.61%	58.61%	58.61%	58.61%	58.61%
NB	56.38%	56.96%	56.38%	56.58%	56.58%	56.77%
J48	54.93%	55.71%	55.42%	55.90%	55.61%	55.51%
NN	57.64%	56.67%	58.03%	58.03%	57.74%	58.41%

Table 3.6: Accuracies of the second experiment using 10-fold cross-validation.

The results from the rolling game PDO average can be seen in table 3.6. While all algorithms are statistically better than the baseline there does not appear to be any difference between the variants of PDO. This suggests that performance metrics such as PDO are not good indicators of predicting a single game.

3.5 Conclusion

In these experiments, we analyzed the ability to forecast success in a single game in the National Hockey League. We compared the ability to predict the outcome of a game by using traditional statistics, advanced statistics and a combination of both. After tuning, the best results came from a meta-classifier with an accuracy of 59.8% using Neural Networks (although the difference between NN and SMO was very small). Our work confirms the intuition of earlier sports researchers that Neural Networks are a good choice in predicting a single game.

We used the CfsSubsetEval function of WEKA to evaluate the features to see which is contributing the most to our classifier. The best features appear to be the traditional statistics: location, goals against and goal differential. This goes against previous work which has demonstrated the advanced statistics success in the long run. Additional works shows that predicting a single game with just the three features produces a model that

is not statistically different in its predictions from the 66%/33% train/test model with all features.

We analyzed the effect of random chance on future outcomes, as teams regress to the norm over the season. Our results found that PDO and other performance metrics did not improve our classification rate.

We believe that our model is correct; however, we think that there are many areas to further examine. There are many other features we wanted to use but there is no historical record so we could not retrieval historical values; they might improve the overall classification rate and future work would need to actively focus on collecting these values while available. These include the number of days of rest between games, time-zone shifts, travel time and distance, change in altitude and climate, weather at the arena of the game, injuries, different variants of possession statistics (e.g. Score-Adjusted Fenwick), statistics based on the starting goalie, recent roster changes due to injuries and trades. Additionally, as these experiments were based on the data from a single season, where teams only played 48 games (rather than 82, there was not enough time for teams to fully regress to their norm). Repeating the results on a full season might return different results, though our intuition and experience with this field suggests this is unlikely.

Instead of different features, we would like to explore why it has been difficult to predict a single game at an accuracy of 60% or higher. As well, if predicting a single game is difficult, then predicting over the long term might be easier with advanced statistics as the random noise in a single game is likely to regress. We shall explore these two ideas in future chapters. First we shall examine luck and upper bound in predictions in the next chapter.

Chapter 4

Upper Bound

4.1 Introduction

Upper bound in predicting outcomes applies to all areas of predictions within machine learning and statistics. In this chapter we explore the difficulty we found in chapter 3 in increasing the accuracy of our single game prediction model. Regardless of the features we explored, and the methods we used we were not able to bring our accuracy higher than 60%. We explore if there is an upper bound in hockey predictions and then analyze other hockey leagues to look at the variance within the different leagues. We find evidence that suggests the existence of a possible upper bound which seems to be related to the amount of parity (the difference in talent between the best and the worst teams) within a league, as well as the number of games played in a season. This chapter was presented at the 2014 Canadian Conference on Artificial Intelligence (Weissbock and Inkpen, 2014).

4.2 Background

In chapter 3, we looked at predicting a single game using both traditional and advanced statistics. While traditional statistics have been around longer and are tracked by the NHL, advanced statistics are shown to have a higher correlation with wins and points over the long term. Despite this, our work found that the most valuable features were: location (Home/Away), goals against and goal differential. This is important, as we use these features to build models in other hockey leagues to help explore luck and variance within the standings.

Random Chance ("luck") has been explored by others within the hockey analytics

community. Tango (2006) compared the true talent level within sports leagues using the classical test theory. This theory was original introduced by Novick (1966) with the argument that it can be used to improve the reliability of psychological tests. The classic Test Theory can be described mathematically as the observed output is equal to the true score (or talent) plus the expected error (as seen in equation 4.1).

$$Obsereved(score) = True(score) + Error \tag{4.1}$$

Tango's hypothesis is that you can subtract an estimated variance using a binomial distribution, from the observed variance and you will have the actual variance of talent in the standings. With this method, he found that random chance explains 37.64% of the variance within the standings in the NHL.

Burke (2007) explored the possibility of a theoretical upper bound in the National Football League by comparing observed win/loss records of teams in a single season to theoretical leagues, until he found one most similar. He determined that the NFL is statistically identical to a league where 52.5% of the games are determined by skill, while the remaining 47.5% are determined by random chance. He concluded that the NFL has an upper bound of prediction of 76%, which appears to be the limit reached within NFL prediction studies.

"Luck" is considered to be a major factor in an NHL team's performance as hockey analysts often look at a team's PDO value (as briefly discussed in chapter 3). PDO is the summation of the team's Save Percentage and Shooting Percentage. Teams who have a PDO that is larger than 100% are considered to have been "lucky," while teams who are less than "100%" are considered unlucky. This is because one or more of these two variables which make up PDO have become unsustainably high (or low). PDO has plenty of statistical analysis as performed by Dermody (2013) and over a full season teams will regress their PDO to within 2% of 100%, as seen at figure 4.1.

4.3 Method

To analyze random chance within hockey we broke the analysis into two parts. In the first half, we used the Monte Carlo method to determine the upper bound for the National Hockey League. In the second half, we created a number of predictive models for other hockey leagues using the three important features we found in chapter 3: location, cumulative Goals For and cumulative Goals Against. We then compare these classifiers

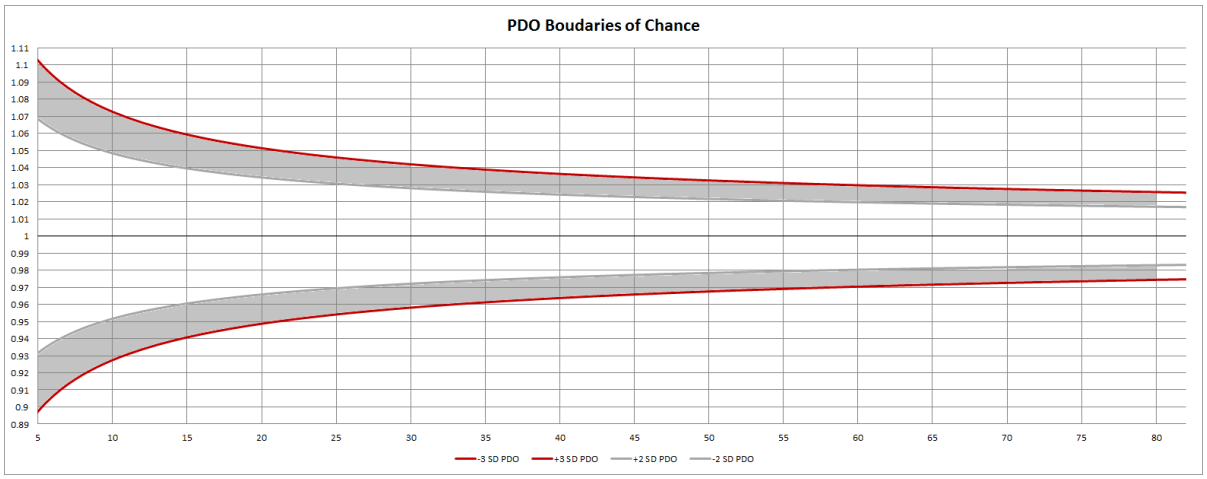

Figure 4.1: PDO Boundaries of chance from Dermody (2013)

to their theoretical upper bounds and use their home team win% as a baseline.

4.3.1 Monte Carlo Method

To calculate the theoretical upper bound of prediction for sports, we used a Monte Carlo method to simulate a hockey season and compared the results to the 2011-2012 NHL schedule.

The Monte Carlo method is a randomized algorithm that relies on repeatedly generating random numbers in many iterations which will converge the results to a fairly accurate estimated output (Metropolis, 1987). The Monte Carlo method is one of many stochastic statistical simulations and was selected due to the simplicity in its use as we intend to create a simplistic model to determine the upper bound. In this instance, we initialized all 30 teams with random strength weights on every iteration (a random number from 1 to 30 with no two teams receiving the same value) and simulated the entire schedule for the year, randomizing the outcomes of games based on various rules, and repeated this process 10,000 times. We chose this number of iterations as it balanced out running time for the simulation and the number of iterations commonly used by members in the hockey analytics field. At the end of the process, we took the average of each standard deviation of win percentage (the ratio of games won to game played by a single team) and compared it to the observed standard distribution.

Each time we ran the Monte Carlo, we varied the amount of "skill" (between 1 and 100)

and Random Chance ("luck" = 100 - "skill") required to win a single game. "Skill" refers to all of the actions on the ice that players have control over and can lead to winning the game, while "luck" is all of the items the players cannot control such as when the referees determine if there has been a rule infraction that is in question. While this method simplifies the factors on the ice greatly, we used this method to keep our method for determining the upperbound straightforward and uncomplicated. Further work should look at verifying this upperbound using alternative methods such as a Markov Chain Monte Carlo. The amount of pre-determined skill and luck established the rules to randomly determine who would win a game. The algorithm to determine the game winner is based on a randomly generated number between 1 and 100. If this number is less than or equal to the amount of luck%, then the winner is determined by a coin flip; if this number is greater, then the team with the higher strength weight wins. This is further described in algorithm 1. We surveyed the ratio of skill to luck until we found the simulated season most similar to the observed data. This method is similar to Burke (2007) but rather uses the standard deviation of observed win percentage over many seasons.

if *rand(1,100)* $\leq$ *luck* **then**
| *winner = coin flip;*
else
| *winner = stronger team;*
end

Algorithm 1: Algorithm to determine the game winner in Monte Carlo

For our experiment, we calculated the observed standard deviation of win percentages by looking at all teams win percentage between the 2005-2006 season and the 2011-2012 season (a total of 7 full seasons). These dates were chosen as the 2005-2006 season was the first NHL season to introduce the salary cap, where each team had a hard limit of money to spend on players salaries, which established parity amongst the league. The 2011-2012 NHL season was the last full NHL season due to a labour lockout during the 2012-2013 NHL season causing the season to be shortened from 82 to 48 games. As teams can both win (and lose) in regulation, overtime or the shootout, we considered any win as a win and any loss as a loss.

This gave us a total of 210 observed team-seasons. From this data, the standard deviation of observed win percentages is 0.09.

By varying the amount of luck and skill required to win a game in our simulated season, we can simulate various seasons until we find the ratio that is most similar to our observed data. This will help us establish our theoretical upper bound. We know that if x% of games are being won because of skill and 1-x% of games are being determined by a coin flip, then a perfect machine learning classifier will be able to predict x% of games correctly (due to the better team winning) and the other 1-x% of games would be guessed correctly with a probability of 50%. This can be expressed as in equation 4.2:

$$\begin{aligned} Limit &= Skill + Luck/2 \\ &= x + \frac{1-x}{2} \end{aligned} \tag{4.2}$$

It should be noted that since we do not have a large number of observed team-season win percentages, we cannot be certain how the tails of the distribution are formed. As we compare the standard deviation of simulated games to the observed, we assume it follows a binomial distribution as it is a simple method that logically follows how wins and losses are determined; however, as observed by Emptage (2013) it is possible that the data is distributed with a gamma, beta or other distribution.

4.3.2 League Prediction Classifiers

In the second part of this analysis, we wanted to see if this "glass ceiling" in predictions holds for other hockey leagues. To do this, we re-ran the Monte Carlo simulation on a number of different hockey leagues, with the same method as in the previous section. Then we created game classifiers using the three features we found from chapter 3, location, cumulative goals against and cumulative goal differential, as they were deemed most valuable to prediction; also due to the lack of data for these leagues it was not possible to recreate the performance metrics.

In order to determine the validity behind the possibility of a theoretical upper bound we looked at a large number of hockey leagues to cover all ages of players, talent levels and geographical areas. We analyzed: the National Hockey League (NHL), which is considered to be the top league in the world for hockey players; the American Hockey League (AHL), a minor league which is used for younger player development before they are ready to play in the NHL; the ECHL (formally known as the East Coast Hockey League), a professional league a tier below the AHL; the Western Hockey League (WHL), the Ontario Hockey League (OHL) and the Quebec Major Junior Hockey League (QMJHL):

Table 4.1: NHL Equivalency Goals

League	Quality
KHL	0.91
Czech League	0.61
Swedish Elite League	0.59
Finland SM-Liga	0.54
AHL	0.45
Swiss League	0.40
German League	0.37
NCAA	0.33
WHL	0.30
OHL	0.30
QMJHL	0.28

three Canadian hockey leagues for players aged 16 to 20 spread out geographically across Canada. These three leagues are the main league that provides talent to the NHL; the British Columbia Hockey League (BCHL), another Canadian hockey league for players aged 16-20 which is considered a tier below the OHL/WHL/QMJHL and often develops players for the National Collegiate Athletics Association in the United States; the Swedish Hockey League (SHL), the top professional league in Sweden; the Kontinental Hockey League (KHL), the top professional Russian hockey league; the Czech Hockey League (ELH), the top professional hockey league in the Czech Republic; and the Australian Ice Hockey League (AIHL), the top organized hockey league in Australia.

This provides diversity of socioeconomic data between the different leagues from around the world. This selection covers competitive players of all ages across three continents. As the NHL is considered the top league in the world, we can see how other leagues compare to the NHL using NHL Equivalency Goals in table 4.1. NHL equivalency goals have been examined by Desjardins (2009) who looked at players who moved from different leagues to the NHL and analyzed how their scoring changed. As the quality value becomes closer to 1.0, the more similar that league is to the NHL.

This selection of data does not cover women's hockey and minor hockey leagues in areas of the world such as Africa, Asia or South America, which is understandable given the lack of interest in hockey from these continents. Additionally, all of these leagues are

mainly played by men. The top women's league, the Canadian Women's Hockey League, does not have enough public data to be properly analyzed in a similar fashion.

All schedules and win percentages for these leagues were extracted from `http://www.hockeydb.com` and Python and Microsoft Excel where used to prepare the data and format it as required for Weka (Witten and Frank, 2005).

For each league, we used the most recent full schedule. We calculated games in the same format as in chapter 3, giving us two vectors, one for the away team and one for the home team and they included the differentials of the three statistics, and they were labelled with "Win" or "Loss". Using WEKA, we calculated the accuracy of a number of different algorithms to see which returned the highest rate. Similar to before, we surveyed NaiveBayes, JRip, J48 Decision Trees, Support Vector Machine (SMO), MultilayerPerceptron (NeuralNetworks), Logistic Regression, SimpleLogistic, RandomForest and Majority Voting using 3 and 5 classifiers. We did not tune these algorithms due to the large number of algorithms that we were surveying, although our work has suggested the accuracies can be improved by additional 1-2% with tuning.

We compared our highest accuracies with the baseline, which is the home team win percentage, and the theoretical upper bound calculated as in the previous section. Results for all leagues can be seen in tables 4.3, 4.4 and 4.5.

4.4 Results

The results from the Monte Carlo simulation for the NHL, with the varying degrees of skill and luck, can be seen at table 4.2. We provide the p-value after using an F-Test to compare the observed standard distribution of win% of the NHL (0.09) to the Monte Carlo standard distribution win% in order to determine which league is most similar. Statistical tests allow us to see if there is any "real" difference between two models (Han et al., 2006). The p-value of a statistical test gives us the probability that the two items we are comparing are statistically different with a confidence of 95% or higher. An F-Test is a statistical test which allows us to compare two continuous probability distributions (F-Distributions) (Berk, 1996). While t-tests compare the difference between the means of two populations, we use an F-test to compare our distributions as the F-test is often used for statistically comparing standard deviations and normally distributed populations. We use the p-value to give us the probability of obtaining the confidence in the statistical significance. For example, if $p < 0.01$, this means a confidence of 99% or more.

From table 4.2 we can see that neither a league with "all-skill", where the stronger

Table 4.2: Monte Carlo Results

Luck	Skill	Theoretical Upper Bound	St.Dev Win%	F-Test p-value
0	100	100.0%	0.3000	4.90×10^{-16}
100	0	50.0%	0.0530	0.029
50	50	75.0%	0.1584	0.002
75	25	62.5%	0.0923	0.908
76	24	62.0%	0.8980	0.992
77	23	61.5%	0.0874	0.894

team always wins, nor the "all-luck" league, where each game is a coin flip, are statistically similar to the observed NHL. If we examine the results, we can see that the NHL league is much more similar to the "all-luck," (100 luck, 0 skill; where every game is a coin flip) league which is quite surprising. The results of these distributions can be seen visually in figure 4.2.

As we vary the ratios of skill to luck, we start to narrow down which league the NHL is most similar to. This turns out to be a league where 76% of games are determined by a coin flip ("luck") and 24% of games are determined by the better team winning ("skill").

The results from the second experiment where we look at predicting individual games for different leagues can be seen in tables 4.3, 4.4 and 4.5. The headers for each of the rows in the table are as follows:

- **Trained Season St.Dev**: the standard deviation of win-percentage (win%) of all teams in the single season the classifier was trained on.
- **Obs win St.Dev**: the standard deviation of win% of all observed team seasons.
- **Teams in MC**: the number of teams used to simulate a single season in the Monte Carlo method.
- **Seasons in Obs Data**: the number of observed team seasons used to calculate the Obs win St.Dev.
- **Gms/Team in MC**: the number of games a single team plays in the simulated season in the Monte Carlo method.
- **Upper Bound**: the calculated theoretical upper bound for prediction on this limit.

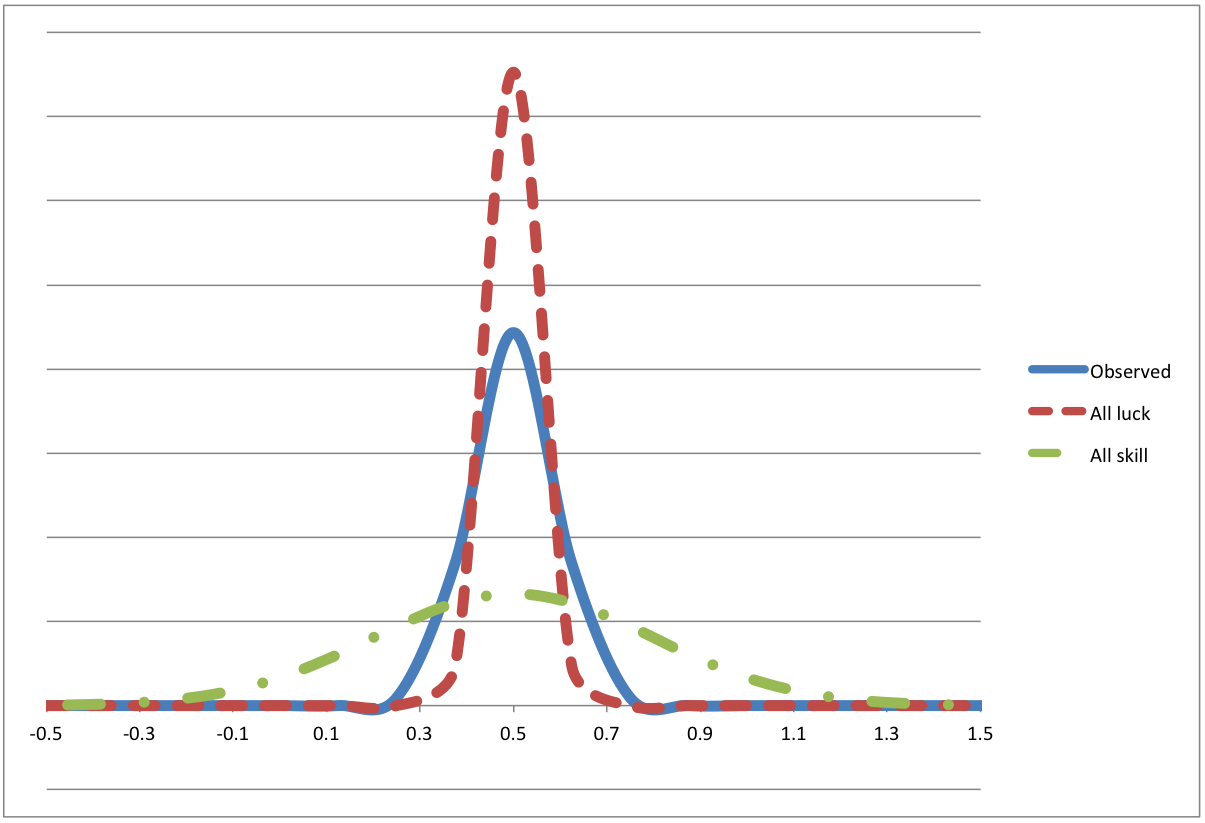

Figure 4.2: All-luck & all-skill league winning percentages vs the observed winning percentages.

Table 4.3: Experiment 2 Results - Part 1

League	NHL	AHL	ECHL
Trained Season St.Dev	0.109	0.07	0.102
Obs win St.Dev	0.09	0.086	0.11
# Teams in MC	30	30	30
Seasons in Obs Data	240	231	238
Gms/Team in MC	82	76	76
Upper Bound	**62%**	**60.50%**	**65%**
Classifier Trained # Gms	512	1140	720
Accuracy	**59.80%**	**52.58%**	**58.70%**
Limit Differential	-2.20%	-7.92%	-6.30%
Home Win%	56.80%	52.50%	55.90%
Baseline Differential	3.00%	0.08%	2.80%
Classifier	Voting	Simple Log	Simple Log

- **Classifier Trained # Gms**: the number of games the prediction classifier was trained on.
- **Accuracy**: the best accuracy a single classifier achieved in this league.
- **Limit Differential**: the difference between the Upper Bound and the Classifier %.
- **Home Win%**: the percent of games that the home team won in this trained season.
- **Baseline Differential**: the difference between the Classifier % and the Home Win %.
- **Classifier**: the algorithm that returned the higher accuracy.

These tables show us the results for each individual hockey league as well as their theoretical upper bound. We emphasized the upper bound and classifier accuracy columns to show what each league's individual upper bound is and how well a simple classifier can do in terms of accuracy with only the three features. Additional information is provided

Table 4.4: Experiment 2 Results - Part 2

League	WHL	OHL	QMJHL	BCHL
Trained Season St.Dev	0.15	0.134	0.166	0.178
Obs win St.Dev	0.141	0.143	0.146	0.155
# Teams in MC	22	20	18	16
Seasons in Obs Data	324	298	283	165
Gms/Team in MC	72	68	68	56
Upper Bound	**72%**	**71.50%**	**72.50%**	**73%**
Classifier Trained # Gms	792	680	612	480
Accuracy	**63.07%**	**63.60%**	**65.52%**	**66.88%**
Limit Differential	-8.93%	-7.90%	-6.98%	-6.13%
Home Win%	55.50%	55.50%	53.80%	52.90%
Baseline Differential	7.57%	8.10%	11.72%	13.98%
Classifier	Logistic	Logistic w/ Bagging	NaiveBayes	SMO

Table 4.5: Experiment 2 Results - Part 3

League	SHL	KHL	ELH	AIHL
Trained Season St.Dev	0.132	0.143	0.089	0.15
Obs win St.Dev	0.115	0.137	0.119	0.191
# Teams in MC	12	26	14	9
Seasons in Obs Data	204	120	238	47
Gms/Team in MC	55	52	52	24
Upper Bound	**69%**	**70.50%**	**66%**	**76.50%**
Classifier Trained # Gms	330	676	364	108
Accuracy	**60.61%**	**61.02%**	**61.53%**	**64.81%**
Limit Differential	-8.39%	-9.48%	-4.47%	-11.69%
Home Win%	52%	56%	61.50%	48%
Baseline Differential	8.61%	5.02%	0.03%	16.81%
Classifier	Neural Network	Voting	SMO	simple Log

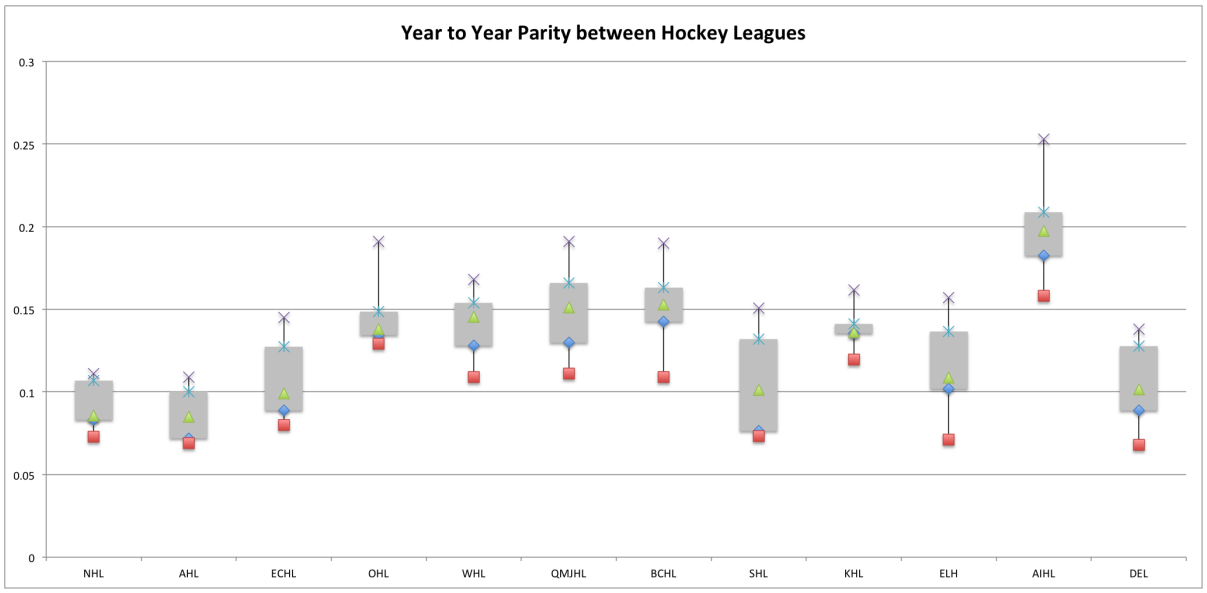

Figure 4.3: Year-to-Year parity in leagues

to show how much data was required to calculate the upper bound and how much better the classifier performed compared to the baseline.

Additionally, we graphed the year to year parity levels of each league, in figure 4.3, using the standard deviation of win% for each season in each league, starting from the most recent data in 2012-2013, going back to 1996-1997. Each league graphed shows the same data as used to calculate the Obs win St.Dev. Due to the creation of leagues at later dates, as well as the lack of historical records in some cases, it was not possible to go back as far in each case.

4.5 Discussion

From the original Monte Carlo method for the NHL at table 4.2, it appears that the NHL is most statistically identical to a league where 24% of the results are determined by skill and 76% are as a result of random chance. The F-Test supports this statement as the p-value is most similar for this league.

This would suggest that the best NHL classifier would be able to predict 24% of games

correctly while guessing 50% of the other 76% correctly. This suggests a theoretical upper bound for the NHL, as calculated in equation 4.3, of approximately 62%. The work done on producing a classifier for the NHL in chapter 3 using the three traditional features was able to predict at 59.8%, which is close to the upper bound.

$$24\% + \frac{100\% - 24\%}{2} = 62\% \tag{4.3}$$

We must re-emphasize that due to the small sample size of observed team-season win% (a set of 210 seasons; prior to the 2004-2005 season the league did not have a salary cap, which would affect the upper bound) and the assumed binomial distribution, we cannot say for certain that the upper bound for the NHL is exactly 62%, but we are fairly confident the upper bound is close to it. We can also use this upper bound across different leagues to see which are easier to predict (which would be the AIHL followed by the BCHL).

These results make sense, as hockey is a very low-event games where a single goal can often win a game. Goals can be scored for a variety of reasons outside of control of individual players such as referees determining that a team should receive a penalty when they did not break an infraction, referees missing infractions and not calling a penalty, pucks hitting players or players sticks and redirecting into the net, pucks hitting the goal posts and bouncing away from the net, injuries to key players, players riding high percentages (i.e., shooting percentage or save percentage) etc. Additionally, with the introduction of the NHL salary cap, making teams much more even in parity, this continues to lower the overall theoretical upper bound.

There are plenty of hockey analysts (Yost, 2013; Birnbaum, 2013a) who agree that luck plays a huge role within hockey. Birnbaum (2013a) goes on to describe that this is not to discount the players and say they are not talented, but rather due to the fact that players are so talented, the relative skill between teams is small. The salary cap further affects the relative skill difference between teams and the fact that games have such a low number of events lowers the overall upper bound. Because of the large role of luck within hockey, a hockey season can take 36 (Tango, 2006) or 73 (Birnbaum, 2013b) games for skill to overtake luck in the standings (depending on how you calculate it). This too suggests that leagues that play fewer games in a season will have less time for random chance to regress to the norm, making it harder for prediction.

This method for calculating the upper bound for other sports, as well as the theory on how it works appears to hold for other sports. Those sports with a higher number of events per game (American Football, Basketball), sports with more games in the

schedule, and sports with fewer restrictions on salary spending per team (Football) will have a higher upper bound. By reading the sports prediction literature, this appears to hold true. Tennis is a sport at the high end of the theoretical upper bound. Given that there can be hundreds of points (or events) in a single tennis match (with a minimum of 72 for men's games) you are less likely to see a weaker player sustain a "lucky" streak, where they play better than their true talent, over an entire match (and vice-versa with a strong player on an unlucky streak). With the number of events a game seemingly to be an important factor in determining the upper bound in sports, this would suggest that tennis would have a higher upper bound than most of these other sports.

The upper bound does not appear to be correlated with the talent of the leagues but more with the parity ($r^2 = 0.9262$). We can compare the NHL to the KHL, both considered the top two leagues in the world (and verified by table 4.1), they have two different prediction limits, as well as having two wildly different parity levels. Some of the junior leagues, with young adults aged 16-20, have more parity than the KHL. Looking at the economic structure of the NHL vs the KHL, it makes sense that there are two different parity levels. In the KHL, there is a great divide in revenue between the richest and poorest teams and as a result there is a large divide between teams being able to pay for the best players.

When we examine the year to year parity levels of each of the different leagues in figure 4.3 gives further insight into the results. Each individual data point is based off of 12-30 team seasons which are a small sample size. This is why some leagues vary wildly from year to year. Nonetheless, it gives us a good idea which leagues have more parity than others. From this, we can conclude that the more parity in a league, as they approach game outcomes becoming coin flips, the more difficult it becomes to predict, as in the AHL. Additionally, when there is less parity in a league, the upper bound rises and the simple classifier using only the three features becomes more distant from the upper bound. This suggests that in these cases we may need to find additional features, including the ones that did not provide an increase in inaccuracy compared to the NHL.

The difference in parity between leagues leads to some interesting observations. The NHL and the QMJHL are both on opposite ends of the spectrum in terms of parity levels and talent levels. The NHL has a much higher parity level, due to the talent level and spending on salaries, while the QMJHL has a lower parity level and is easier to predict. Despite this, the prediction accuracies for these leagues are not overly different. The ROC curve for these reflects the similarity in their accuracies with the NHL in figure 4.4 (0.6285) and the QMJHL in figure 4.5 (0.6963). The confusion matrices for

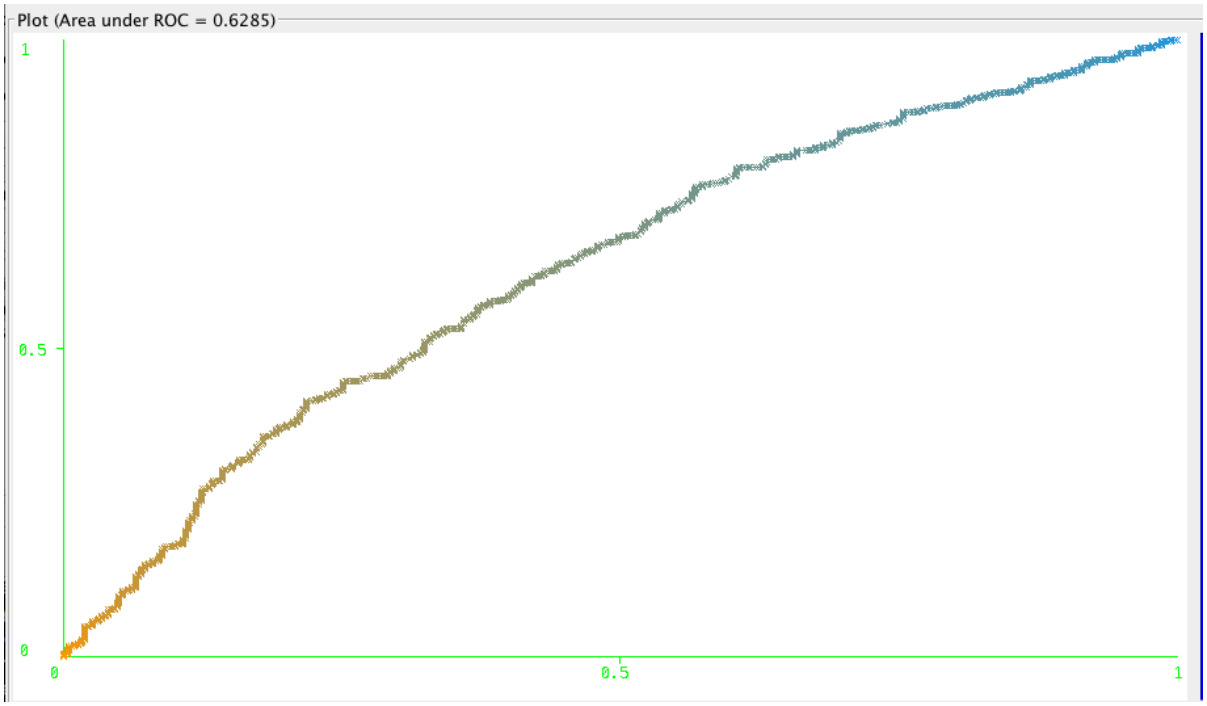

Figure 4.4: NHL classifier ROC curve

these two leagues can be seen at tables 4.6 and 4.7. A confusion matrix is "a useful tool for analyzing how well your classifier can recognize tuples of different classes... items in m rows and m columns indicate the number of tuples of class i that were labelled by the classifier as class j... ideally the non-diagonal entries should be zero or close to zero." (Han et al., 2006). Both confusion matrices suggest the classifiers are working similarly for both leagues despite the difference in parity.

By analyzing the table 4.6 and table 4.7, we can see the difference between the classifier's accuracy, the baseline and the upper bound. None of the algorithms do worse than the baseline though; we have previously discussed the effects of parity on predictions.

Table 4.6: Confusion Matrix for the NHL classifier

Predicted			
Win	Loss		
306	211	Win	Actual
211	306	Loss	

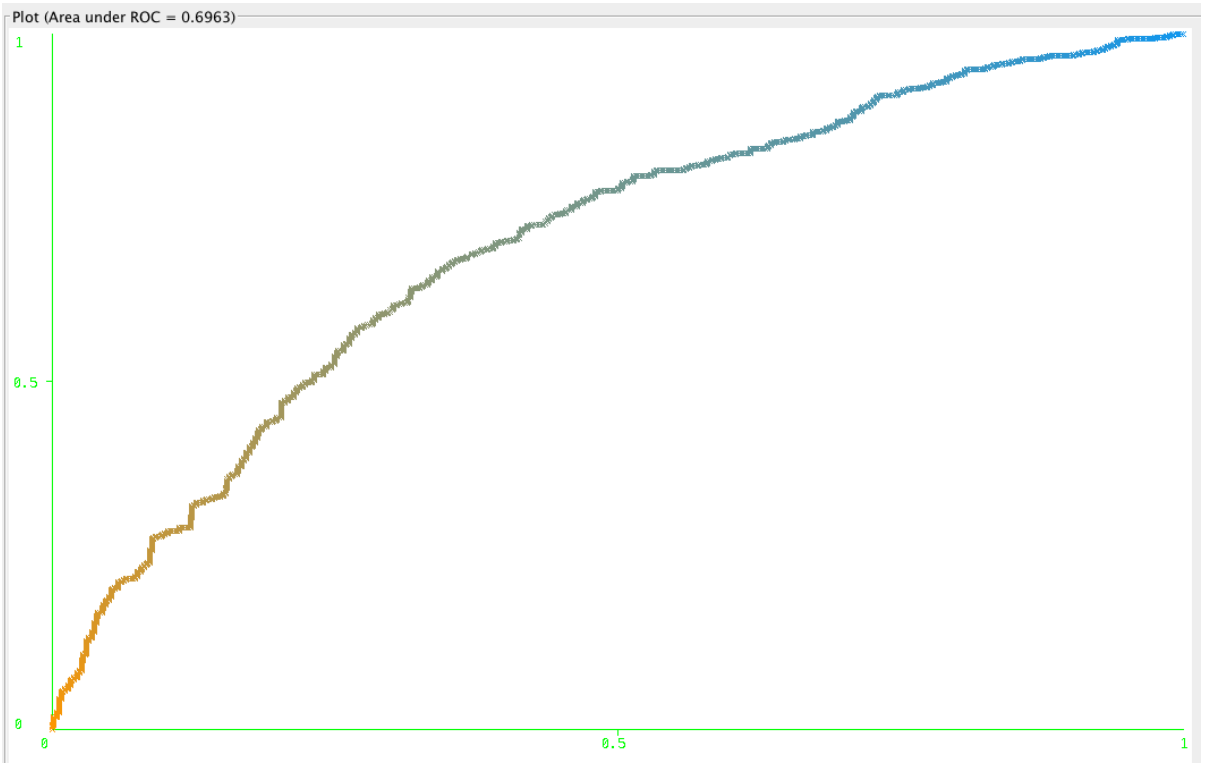

Figure 4.5: QMJHL classifier ROC curve

Table 4.7: Confusion Matrix for the QMJHL classifier

Predicted			
Win	Loss		
399	213	Win	Actual
209	403	Loss	

Table 4.8: NHL data trained on multiple seasons

Trg	2005	2005-06	2005-07	2005-08	2005-09	2005-10	2005-11
Test	2006	2007	2008	2009	2010	2011	2012
NB	58.18%	54.27%	57.69%	56.06%	52.84%	55.90%	56.05%
J48	58.14%	52.56%	58.42%	57.85%	55.41%	55.21%	56.33%
RF	52.73%	51.22%	51.87%	52.10%	52.32%	53.21%	49.24%
NN	58.30%	52.44%	56.55%	52.03%	53.50%	52.07%	51.32%
SVM	55.09%	53.70%	55.90%	56.06%	51.83%	55.90%	56.89%

Additionally, these tables suggest the importance to train on multiple seasons of data given the small sample size within one season. While the NHL model is balanced in its ratio of True/False Positive/Negatives, this is not quite the case for the presented QMJHL model. Given that these ratios are only off by 4 instances of data, this small sample of errors does not suggest the QMJHL classifier is performing worse than the NHL classifer.

We reiterated this process with NHL data re-training the classifier surveying all of our selected algorithms. We trained (Trg) the NHL classifier on one season and predicted all games in the next season. We reiterated this using all data from the 2005 to 2012 seasons and testing (Test) on the most recent season. The results can be viewed in table 4.8. The algorithms surveyed were NaiveBayes, J48 Decision Trees, RandomForest, NeuralNetworks, and Support Vector Machines (SMO). All algorithms were run in Weka using their default parameters.

With this data, we can see classifiers are able to predict at a much higher accuracy as they are no longer subject to the parity within the one season of training data. One issue that does start to arise is that we see the accuracies shift as we move into later years. This is known as concept drift, where the data the models are learning from, change over time causing the accuracy to decrease. When using SimpleLogistic regression algorithm, we see accuracies as high as 58.66%.

It is interesting to see in this case that the best algorithms came from regression algorithms, such as SimpleLogistic. Our initial results in chapter 3 suggest that Neural-Networks and SVMs were producing the best results with the noisy data. This may be due to no longer being constricted to one season.

4.6 Conclusion

In this chapter, we looked at if there exist a glass ceiling for predictions for hockey and other sports. We divided the work up into two parts, first looking at if there is an upper bound, using the NHL as an example. In the second half, we used our method to explore various other leagues to see if we can create a simple classifier that can predict close to each league's upper bound. We analyzed eleven different hockey leagues of varying ages, talent levels and geographical locations and found that the results hold throughout.

The motivation came from chapter 3 where we found it was difficult to predict a single game within hockey and that we were not able to increase the overall final accuracy higher than 60%. Finding the theoretical upper bound in sport predictions applies to all areas of statistics and machine learning as it is not always possible to predict with a 100% accuracy.

The upper bound was found using the Monte Carlo method to simulate an NHL season with varying amounts of skill and random chance required to win a game. After exploring various ratios of skill and luck, with 10,000 iterations each, we found the NHL is most statistically similar to a league were 24% of the games are determined by "skill" while the other 76% of games are determined by random chance ("luck"). This suggests the best we can predict in the NHL is at an accuracy of approximately $24\% + \frac{76\%}{2} = 62\%$.

When reviewing the results of our method, the biggest issue that might be causing errors could be coming from the small sample size of observed winning-percentages. We used a maximum of 200-300 observed seasons in this method for each league. We hesitate to expand to years prior to 2005 (in the NHL model) when the NHL introduced the salary cap, establishing parity amongst the teams.

This leads to another issue; given how dynamic the league is, we are likely to see changes in parity from year to year due to how much money teams are spending on salaries, the number of top young talent teams have acquired and from the management learning the rules of the system. Over time, this can cause the parity to become either higher or lower. We cannot trust the results when applying the Monte Carlo method to a single season, as 30 data points maximum, for a single league, is too small of a sample size for confident results.

We looked at 11 different hockey leagues, including the NHL, and created classifiers that could accurately predict games as close to their theoretical upper bound as possible. We used the three important features found in chapter 3 of location, goals against and goal differential. We compared the classifier accuracies to both the baseline and the

upper bound. In all cases, the classifiers improved from the baseline but in some leagues this improvement was small. We hypothesize this is a result of the parity within the leagues. As well, the parity within the single season data that are used for training was lower than the league's average parity.

Reflecting on the results, we feel comfortable in saying that we were successful in demonstrating our hypothesis. There does appear to be an upper bound in sports predictions that we cannot achieve an accuracy higher than. Using the Monte Carlo method, we can approximate this limit. This glass ceiling appears to be supported by hockey analysts. Given that this method can be used for every major sport, we feel confident it can be used to identify other sports' upper bounds. We would hypothesize that sports with more events per game, such as American football and basketball would have a higher upper bound, as would sports with less parity such as soccer.

When analyzing the results of the individual classifiers, we can see they all potentially suffer from the same errors as in chapter 3. Training is only limited to a single season and, in some cases, some leagues have a small percentage of the games that the NHL plays; the AIHL is a good example of this. These small sample sizes could be having an effect on our classifiers.

When looking at the results from previous work, predictions in each sport seem to be stuck at a certain threshold which all related works have not broken. This seems to be further suggested by Zimmermann et al. (2013) American Football appears to be approaching a 75% threshold while basketball is approaching a threshold in the high 80%. Basketball has many more events per game than football and both have more events than in hockey. Soccer has also approached accuracies in the 80% range and soccer has less parity than hockey. We would hypothesize these sports also have an upper bound which for which automatic methods predict close to and we believe the methods that we proposed for calculating the upper bound can be used.

Cullen (2014) describes best: "No one likes it when luck is referred to when evaluating the performance of hockey players and teams. It goes against our ingrained notion, particularly in sports, that hard work is rewarded and that players create their own luck. But, the truth is, there are so many things that happen on the ice over which a single player has little to no control."

Hockey is an extremely difficult sport to predict, especially at the elite levels, due to the relative difference in skill, combined with the low number of events a game. This has made us hypothesize that, if predicting a single game with an accuracy higher than the baseline is possible but difficult due to the existence of a low upper bound, then

predicting over a longer series of games, such as a best-of-seven play off series, could be easier, as random chance begins to even out.

Chapter 5

Playoff Prediction

5.1 Introduction

In the National Hockey League, and many other hockey leagues, at the end of the season, the best teams are sorted into a post-season tournament to determine the season's champion. Normally, there are 16 teams in the tournament, with pairs of teams playing a best of seven series. The first team of the two to win four games is declared the winner and moves on to the next round. Each round, teams are seeded by their end of season standings, with the top teams playing the bottom ranked teams. Teams play games at both of their teams stadiums; there is an advantage for each individual game for the home team, as the home team wins 56% of the time. As there is an odd number of games, the higher-ranked team plays the four of the seven games at their stadium. Home series advantage is determined by the team with the higher end-of-season ranking. The team that is last eliminated (normally after four rounds) is the champion of that season. In various leagues, teams win different trophies; in the NHL the final team wins the Stanley Cup, in the AHL the final team wins the Calder Cup and in the KHL the final team wins the Gagrin Cup.

In chapter 3, we saw the difficulty of using Machine Learning to predict a single game. Using both advanced and traditional statistics for features, we were not able to increase the accuracy over 59.8%. When we examined this further in chapter 4, we learned that random chance plays a huge role within hockey, which lowers the upper bound to approximately 62% for the NHL, and makes single games difficult to predict.

Mlodinow (2009) explains in his book "The Drunkard's Walk" that a team who can beat another team 55% of the time will still lose a 7-game series, to the weaker team, 4

times out of 10. Teams that have a 55%-45% edge in skill difference will still require a minimum of a best-of-269 game series for the better team to always emerge victorious. That being said, we hypothesize that, if one game is difficult to predict because of random chance, then over a large number of games, the random chance will start to regress to the norm, the better teams will come ahead, and predicting will become easier. We look at predicting a best-of-seven playoff series where two teams play a maximum of seven games to see which team is most likely to win. This seems to be the opposite of other areas of prediction, such as weather, where over the long term random chance plays a large role and things become harder to predict.

5.2 Data

For this experiment, we took the lessons learned from predicting a single game and used them to predict a playoff series. As we are using both traditional and advanced statistics in this chapter, this limits us to only playoff series that are available during the "BehindTheNet" era, that is all years where advanced statistics have been published by Internet hockey analysts. This limits us to all playoff series from 2007 to the present. With 15 series every year and only six seasons available, this limits us to 90 data samples. This is a smaller sample size than we would prefer, but to wait for a much larger sample would require waiting for another decade.

Rather than train each playoff series as two data vectors like we did in chapter 3, we used a single data vector which is the concatenation of the differences of the two team's features. We chose this representation as it allows us to have a stronger baseline, of picking the majority class, which is the team with the home advantage, to 56% rather than 50%. This data representation has been used in previous sports prediction work as well as our work in chapter 3 but did not find it affected the overall accuracy. $Team1'$ and $Team2'$ time were first calculated as by equations 5.1. Then the two vectors were concatenated and given a label of the team who won, i.e. $label =$ "$Team1$" or "$Team2$". The final data vector was calculated as in equation 5.2. An example can be seen in table 5.1.

$$\begin{aligned} Team1' &= Team1 - Team2 \\ Team2' &= Team2 - Team1 \end{aligned} \tag{5.1}$$

$$Data = Team1' + Team2' + label \tag{5.2}$$

Table 5.1: Training Data Vector Example

e.g.	Team1 Data			Team2 Data			Label
1	$T1'_1$	$T1'_2$	...	$T2'_1$	$T2'_2$	...	Team1

All differentials were calculated using python and then prepared in the format required by Weka. Not all statistics were used in both team's vectors in cases where teams had the same value (i.e., year of the series). This is further explained in the list of all features. All statistics are available at the end of the regular-season and known before the playoff series. 34 different features for each team were collected and they are described as follows:

- **Year** (Traditional): Labels the year of the playoff series. Was only used once in each training vector.
- **Team** (Traditional): The name of the team involved in the playoff series. This feature was not used in the machine learning algorithms but was kept during pre-processing to manage the data.
- **Win** (Traditional): If the team won or lost the playoff series. This was used to calculate the label for the data vector.
- **Location** (Traditional): If the team had home field advantage or not. In a playoff series the team with the home field advantage will play a maximum of 4 games at home and 3 away. This was demonstrated to be an important feature in predicting the winner of a single game.
- **Distance** (Traditional): This is the straight line distance between the two cities as measured by Google Maps. As teams spend many hours travelling, there may be an advantage to having to travel less and deal with time chances. This is what was attempted to be captured by including the distance of travel. There is no need to calculate a differential, so this feature was included in the *data* vector once.
- **Conference Standing** (Traditional): The ranking of the team within its own conference at the end of the regular season.
- **Balanced Schedule Winning Percentage** (BSWP) (Advanced): This feature is calculated and publicly available on the Z-Rating website[1] which is an application of

[1] `http://mattcarberry.com/ZRatings/ZRatings.html`

the Bradley-Terry system for paired comparisons, first used in comparing American collegiate ice hockey by Ken Butler. The BSWP is a team's expected winning percentage if each team played every other team an equal number of times. In the NHL, teams play each other a different number of times depending if they were in the same division/conference or not.

- **Division Rating** (Advanced): The division rating is the BSWP average of the division that the team is currently a member of.

- **Z-Rating** (Advanced): The calculated Z-Rating for the team in that current year, based on the Bradley-Terry system. This is readily available on the Z-Rating website.

- **Strength of Schedule** (Advanced): The Strength of Schedule represents how difficult a team's schedule has been during the regular season. As teams play against teams within their own division the most, by playing more against weaker competition the team might have an artificially higher standing than a potentially stronger team playing in a strong division. This is calculated as the Z-Rating of the team divided by the success ratio and is available on the Z-Rating website.

- **Season Fenwick Close** (Advanced): Fenwick is a statistic that represents puck possession. Teams who have the puck more are more likely to score and less likely to be scored against. Possession has been shown to be correlated to time in the offensive zone. Fenwick is calculated by summing up the shots for, missed shots for and goals for, divided by the same events against. "Close" refers to when the score is within 1 goal, in the first or second period, or tied in the 3rd period or overtime. This helps remove "Score Effects" where teams ahead play more defensively and losing teams play more aggressively. This is found on `BehindTheNet.ca`.

- **Score-Adjusted Fenwick** (Advanced): Fenwick of teams but rather than looking at only "Close" situations, it looks at every situation to account for score effects. This can be calculated from the Fenwick data on `BehindTheNet.ca`.

- **Season Corsi** (Advanced): Similar to Fenwick, Corsi is another statistic of possession, but includes blocked shots for and against as well. Corsi has been shown to be correlated with offensive zone time as well, but includes more events than Fenwick. Corsi for all teams is available at `http://stats.hockeyanalysis.com/`.

- **Shooting Percentage** (Sh%) (Traditional): The percentage of shots that become goals for the team. Over a season the expected norm for teams' shot-percentage have been around 8-9%. If a team has been shooting higher or lower it can lead to poor possession teams to making the playoffs and vice versa. Sh% is found on `http://www.nhl.com/`.

- **Save Percentage** (Sv%) (Traditional): The percentage of shots the team's goalies have stopped over the season. Sv% is found on `http://www.nhl.com/`.

- **PDO** (Advanced): A Summation of Sv% and Sh%, typically only at Even Strength. Over the season PDO will regress to 100%. Teams who are better than the expected norm have been playing better than the norm and are considered "lucky" while teams who are below the norm have been "unlucky". Weak possession teams ($\leq 50\%$ Fenwick Close) who have a high PDO ($\geq 100\%$ PDO) can make the playoffs while strong possession teams may not made the post-season due to a low PDO.

- **Cap Hit** (Traditional): This is the amount of money, in United States Dollars, that management has spent on the players salaries in that season. Typically, better players receive a higher salary; teams spending more money intuitively should have better players. These values are available at `http://capgeek.com/`.

- **5v5 Goals For** (Traditional): The number of goals scored by the team during the season only during even strength (removing Power Play and Penalty Kill situations). This is found at `http://www.nhl.com/`.

- **5v5 Goals Against** (Traditional): The number of goals scored against the team during the season only during even strength (removing Power Play and Penalty Kill situations). This is found at `http://www.nhl.com/`.

- **5v5 Goal For / Against differential** (Traditional): The differential between the goals scored for and against a team during even strength in the regular season.

- **5v5 Goals For/Against** (Advanced): The rate at which a team scores goals vs being scored against during even strength. This is calculated by dividing 5v5 Goals For by 5v5 Goals Against.

- **Win Percentage** (Traditional): The percentage of games played by a team that are won in that season. This data is available from `http://www.nhl.com/`.

- **Pythagorean Expected Win Percentage** (Advanced): This baseball analytic which has been adapted for the use of hockey. Using runs for and against, it tells us what the expected win % of a team should have been. In baseball, a run is scored when a player advances around all three bases and returns to the starting position and the team with the most "runs" wins. This would be similar to scoring a goal in hockey. The Pythagorean Win% is similar to PDO in that we can see who has been lucky and unlucky by comparing the Expected Win% to the Observed Win%. Instead of using "runs" for and against we use goals for and against. The formula for this can be seen in equation 5.3.

- **Points Earned** (Traditional): The total number of points (determined by a function of wins which gives you 2 points, and an overtime loss gives teams 1 point) earned by the team during the regular season. This data is available from `http://www.nhl.com/`.

- **Power Play %** (PP%) (Traditional): The success rate at which the NHL team has scored a goal during the periods the team has had a man advantage on the ice. Data is available from `http://www.nhl.com/`.

- **Power Kill %** (PK%) (Traditional): The success rate at which an NHL has prevented goals scored against them during the period when they have had one less player on the ice than the other team. This data is available from `http://www.nhl.com/`.

- **Special Team Index** (Advanced): Similar to PDO, the STI is the summation of PP% and PK% and over the season it will regress; towards 100%. This allows us to see who has been lucky and unlucky during special team performance.

- **Days Rest** (Traditional): To capture the amount of rest teams have had between series. It is intended to represent which teams have had more time to recover. Too much rest has been cited by Main Stream Media to prevent teams from being mentally prepared for the next series.

- **Games Total Played** (Traditional): This is the summation of total games a team has played in that year's playoffs. As teams play more games, they are intuitively more likely to be fatigued and exposed to injuries.

- **Fenwick Last-7** (Advanced): An NHL team's possession score over the season is represented in the Season Fenwick value but this does not necessarily represent the

current team's possession skill. Due to trades or injuries players can leave or join the team and change the current team's skill level. Adding a strong player can increase the current skill level, while adding a weaker player can have the opposite effect. This feature is intended to represent how strong a team has been over the short term, giving a better idea of how strong the current team has been. Seven games was selected, as it is the maximum number of games in a playoff series. This is calculated by finding the average of the team's Fenwick value over the previous 7 games. Figure 5.1 shows how a teams puck possession can change wildly over a season due to a number of reasons.

- **Corsi Last-7** (Advanced): Similar to Fenwick Last-7 this feature is the Corsi version rather than the Fenwick.

- **Goalie Yr Sv% Last-7** (Advanced): Similar to the previous two Last-7 features, this is to capture the goalie most likely to play in the upcoming series. Injuries to players may arise which could cause the team's starting goaltender to not be able to play; this often becomes a large hindrance to the chances of a team playing in the post season. This is calculated by using the the Year Sv% of the goaltenders starting in the previous 7 games and taking the average.

- **Goalie GAA Last-7** (Advanced): Similar to above, but rather than using the Year Sv% of each goalie, it uses the Goals Against Average, that is, the average number of goals scored against that goalie in a game during that season. GAA is a function of team performance as it is calculated by including the shots against which is a function of team performance.

- **Goal Ev Sv% Last-7** (Advanced): Similar to the above two features, but using the goaltender's Sv% calculated only from even strength goals and shots to remove a potential increase when there are uneven amount of players from both teams on the ice.

$$Win = \frac{1}{1 + (\frac{GoalsAgainst}{GoalsFor})^2} \tag{5.3}$$

Traditional features are those that are readily available from NHL.com and are usually based on simple counting and goal-based metrics. Advanced statistics are those which are calculated with more advanced mathematics, usually have a large sample size and often based on shot metrics.

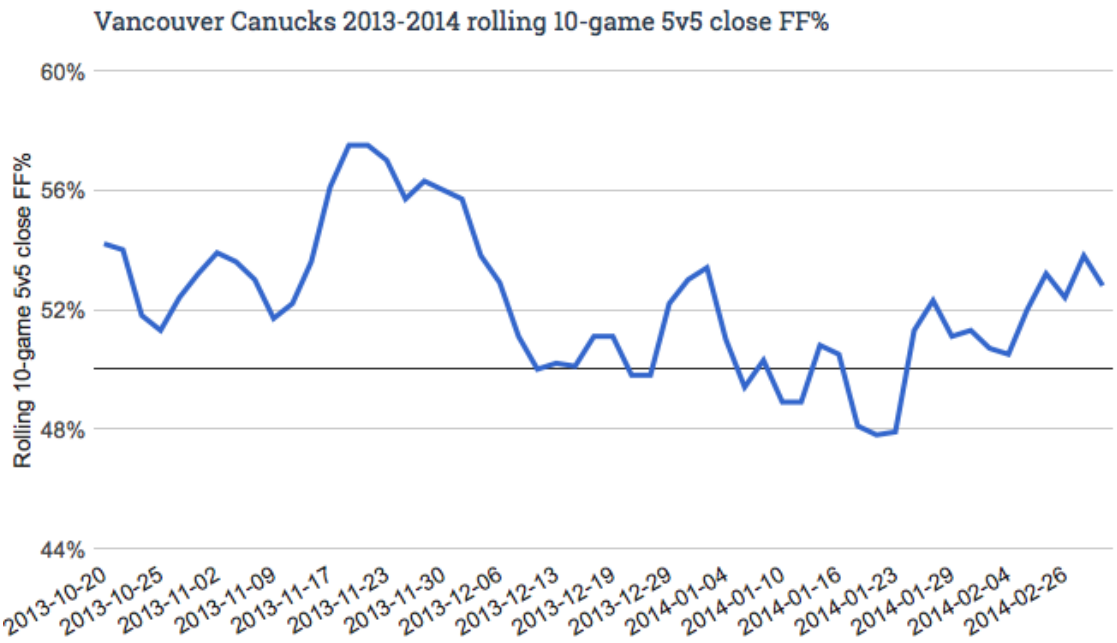

Figure 5.1: 10 Game Rolling Fenwick for the Vancouver Canucks from ExtraSkater.com

5.3 Method

All data preprocessing was done manually using Microsoft Excel and all classifications were completed in Weka (Witten and Frank, 2005) using their implementation of Machine Learning algorithms. Default values of all classifiers' parameters were used as a large number of algorithms were surveyed; further evaluation suggests we can improve the overall accuracy with tuning. The algorithms we surveyed are as follows: JRip (rule based learner), SMO (Support Vector Machines), J48 (C4.5 Decision Tree), iBK (k-nearestNeighbours), NaiveBayes, MultilayerPerceptron (Neural Network), Voting (best of 3: SVM, NB and NN), LogisticReegression, and LibSVM (Support Vector Machine witch has a speed increase over Weka's SMO). These algorithms are the same ones surveyed in the previous chapters.

All classifiers were run using 10-fold cross-validation, where the data was randomly separated into 10 subsets, trained on 9 of the sets and tested on the remaining subset. Then the procedure is repeated 10 times for the other splits. Over the 10 iterations the test data accuracies are averaged out. As SMO was the classifier with the highest

Table 5.2: Results of our Classifiers

Classifier	Accuracy
Simple Classifier	56.00%
JRip	51.00%
LibSVM	53.33%
J48	54.00%
k-NearestNeighbours	61.11%
Random Forests	64.44%
NaiveBayes	64.60%
NeuralNetworks	68.00%
Voting	68.89%
Logistic Regression	70.00%
SMO	71.11%

accuracy, additional tuning was conducted, where we modified the different parameters of the algorithm to further increase the overall accuracy. Tuning the parameters increased the accuracy by nearly 4% which suggests that the small sample size is an issue.

5.4 Results

We present the results from the several Machine Learning algorithms from Weka in table 5.2. Many of these algorithms are able to predict at accuracy higher than the baseline. The highest accuracies are produced by the Logistic Regression and SMO algorithms. SMO had further tuning which further increased its accuracy to 74.44%. This is a larger jump than we expect and we would hypothesize it as a result of the small sample size[2].

Using a paired t-test, SMO, Logistic Regression, Neural Networks and Voting are all statistically different from the baseline ($p \leq 0.05$). The tuned SMO classifier is not statistically different from the un-tuned SMO or the Neural Network, again suggesting the large increase is a result of the small data sample size.

[2]Additional k-fold cross-validations (e.g. 6-fold, leave one out cross-validation) also yield results that suggest an improvement in accuracy from the single game prediction

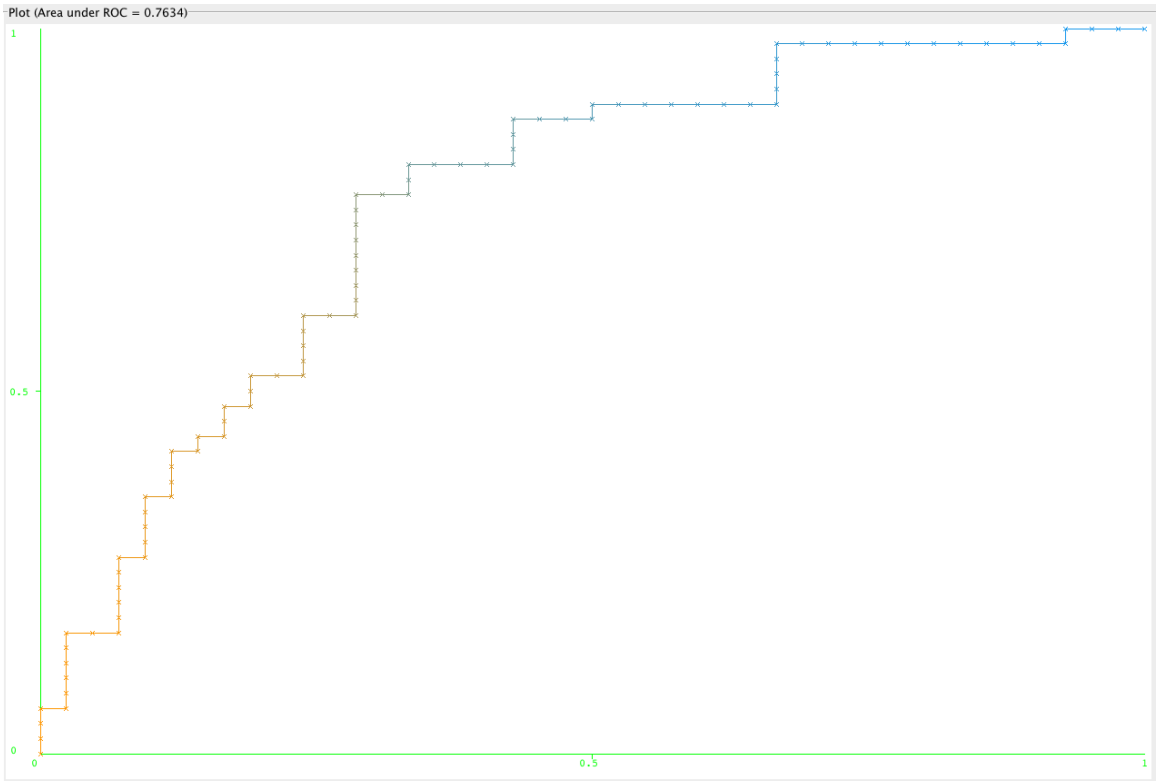

Figure 5.2: SMO classifier ROC curve

Table 5.3: Confusion Matrix for the SMO classifier

Predicted			
Team1	Team2		
37	11	Team1	Actual
12	30	Team2	

Table 5.4: Detailed Accuracy by Class for the tuned SMO classifier

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC	Class
	0.771	0.286	0.755	0.771	0.763	0.763	Team1
	0.714	0.229	0.732	0.714	0.723	0.763	Team2
Weighted Avg.	0.744	0.259	0.744	0.744	0.744	0.763	

We further analyze the tuned SMO classifier by evaluating the results of the ROC curve in figure 5.2 and the confusion matrix in table 5.3. The small sample size is prevalent in both, but the ROC has a relatively good accuracy with the ROC Curve at 0.7634, an improvement over the single game analysis. The confusion matrix suggests there is an improvement from the single game forecasting. Further breakdown of the SMO's performance can be seen in table 5.4.

Asides from the small sample size of data within predicting the playoff series, another issue that could be causing errors is that the large number of features may not be helping our model. Future work should be looking at using feature selection and possibly principal component analysis. This could help reduce the noise and the standard error and possibly increase the overall accuracy. While we saw the large role that luck plays in a single game in chapter 4, it is unlikely that we will achieve 100% accuracy over seven games, but we might be able to improve the accuracy with this method.

In addition to 10-fold cross-validation, we split the data into 66% (60 series) for training and the remaining 33% (30 series) for testing. We achieved similar results to our 10-fold cross-validation results: 74% with the tuned SMO and 70.97% accuracy with Neural Networks.

Similar to our single game, when we looked to see if traditional or advanced statistics were more valuable to prediction, we used CfsSubsetEval (Correlation-based Feature Subset Selection Evaluation) on our data to help us identify the most useful features. In this experiment, the most valuable features were the individual team Z-Ratings, the Pythagorean Expected Win% and Fenwick Last-7.

In our previous work in predicting a single game traditional statistics were much better at predicting games than advanced statistics, despite advanced statistics being considered better in the long term. This is the opposite for predicting a playoff series, where luck starts to regress to the norm and we can see that advanced statistics are much more useful.

5.5 Conclusion

In this chapter, we moved away from predicting a single game, which is difficult as shown in chapter 3, due to the large role that random chance plays. We made predictions over a longer series of games by looking at the example of the NHL best-of-seven playoff series.

While our previous work found it difficult to predict a single game and that traditional statistics were more valuable, we found the opposite to be true with predicting playoff

series. The overall accuracy has increased for the groups of seven games, and advanced statistics were shown to be more useful to our models. We suspect that in seven games the noise of random chance has not fully regressed to the norm and we suspect there is still an upper bound present.

Given the smaller sample size of the data, we have difficulty being confident that we can predict playoff series at exactly 74%, but we are comfortable saying that playoff prediction does return much higher results than single-game predictions.

There are a few directions we can take for future work in this area. The first is to wait for a few years to collect more data and ensure a larger sample size. We also suggest look at additional proxies to many of the advanced statistics which are not published before the 2007-2008 season. Tuning the individual features may have value such as Fenwick Last-n. We chose Last-7 as a playoff series has a maximum of seven games, but the optimal number might be slightly more or less. Verifying the hypothetical upper bound for playoff predictions would be valuable, but not possible with our method from chapter 4 because of the smaller sample size and because half of the teams would have a series win-percentage of 0%. One method to determine the hypothetical upper bound for the playoffs would be using a Markov Chain Monte Carlo.

Hockey is interesting in that it is very random in the short term, which makes predictions difficult, but in the long term, it becomes easier as things return to normal. This has made us look again to see if it is possible to predict a single game at a higher accuracy, or if it is possible to predict games without using team numerical data.

Chapter 6

Meta-Classifiers

6.1 Introduction

In the final portion of our thesis, we re-examine the work that we did in chapter 3 in predicting a single game. We learned in chapter 4 that random chance plays a large role within the sport, which places a theoretical upper bound for individual game predictions at 62%. While our best results lead us to an accuracy of 59.8%, we re-evaluate our model to see if we can improve this overall accuracy and look at other methods of game prediction as an alternate to using team statistical data based on in-game events.

Instead, we focused on predicting single games from using textual data and sentiment analysis to increase the overall accuracy. We also compared these methods to just predicting with in-game numeric data to see how well they perform. It is a difficult task as the window for prediction improvement is only 6% with the baseline (home team win%) at 56% and the upper bound at 62%.

We create three individual classifiers and then we feed the outputs into a second level which makes the final decision on the game prediction. The rules to determine the final output were expiremented in three different ways and we settled on the method that yields the highest accuracy.

We presented the methods from this chapter at the 2014 Canadian Conference on Artificial Intelligence (Weissbock and Inkpen, 2014).

6.2 Data

For this experiment we used data on all games in the 2012-2013 NHL season, the same season we analyzed in chapter 3. As we are not using advanced statistics we were able to use all 720 games within the season. We limited the selection to one single season to minimize any concept drift that may occur if we trained over multiple seasons.

6.2.1 Textual Data

First we collected textual pre-game reports on each game from the league's website at `http://www.NHL.com`. Text reports are written and published on the website within a day before the start of the game; this allows the text reports to capture the latest information the team's status including recent victories and losses, team acquisitions via trades, injuries and opinions of the writers. While the reports are intended to be bias free, as they are published by the league, we suspect there is still unintentional biases within the wording given the writers have knowledge of the latest news in hockey.

Each report published by the league is split into two sections, one for the home team and one for the away team. Not all text reports were clearly divisible, in which case we could not use those games for our dataset. This left us with 708 (of 720) games. An example of a pre-game report data, before pre-processing, can be see at table 6.1. For all text, we used the Python NLTK stopword list (Bird, 2006) to remove all stopwords (words deemed irrelevant, as they do not add value to our text, despite appearing many times i.e. "the" (Han et al., 2006)) as well as any punctuation marks, such as single quotation marks, to allow the text to fit within Weka's required format.

We experimented with a number of different methods to represent the text reports such as using Bag-of-Words, Term Frequency/Inverse Document Frequency (TF-IDF), with stemmer, without stemmer, and 1, 2 and 3 grams.

Stemmers reduce similar words to their group word stems to see which words are similar despite their small syntactic variants (e.g.,drugs, drugged and drug (Han et al., 2006)). n-grams look at groups of n words to see if pairs, triples (or larger) are frequent amongst the texts. We looked at different permutations of Bag-of-Words, various stemmers and with and without bigrams and trigrams. The best results from our trial experiments came from the combination of TF-IDF, no stemmer and with 1, 2 and 3 grams.

As each textual report was already split into two halves, one for the home team and one for the away team, we had two data vectors for each team to train on. This is similar

Table 6.1: Example of Pre-Game text, pre-processed

Text	Label
There are raised expectations in Ottawa as well after the Senators surprised last season by making the playoffs and forcing the top-seeded Rangers to a Game 7 before bowing out in the first round. During the offseason, captain Daniel Alfredsson decided to return for another season. The Senators added Marc Methot to their defense and Guillaume Latendresse up front, while their offensive nucleus should be bolstered by rookie Jacob Silfverberg, who made his NHL debut in the playoffs and will skate on the top line alongside scorers Jason Spezza and Milan Michalek. "I don't know him very well, but I like his attitude – he seems like a really driven kid and I think he wants to do well" Spezza told the Ottawa Citizen.	Win
Over the past two seasons, Ondrej Pavelec has established himself as a No. 1 goaltender in the League, and while Andrew Ladd, Evander Kane, Dustin Byfuglien and others in front of him will go a long way in determining Winnipeg's fortunes this season, it's the 25-year-old Pavelec who stands as the last line of defense. He posted a 29-28-9 record with a 2.91 goals-against average and .906 save percentage in 2011-12 and figures to be a workhorse in this shortened, 48-game campaign.	Loss

to chapter 3 and gives us a baseline of 50%. Consistent with our other chapters, the home team has won 56% of all games within our data, raising the baseline to 56%. With the upper bound at 62% there exists only a small gap of improvement.

The winning team's text was classified with the label of "Win", while the losing team was given the label of "Loss". For the 708, games we had a total of 1416 data vectors, one from the perspective of each team.

6.2.2 Sentiment Analysis Data

In order to detect some of the biases within the text from the writers, we created a second set of data for this set of games using Sentiment Analysis. Rather than looking to classify text by "topical categorization", Sentiment Analysis detects the opinion within text, or rather, how positive and negative text is (Pang et al., 2002). By applying this type of analysis to our textual pre-game reports, we hope to capture the biases and we would expect that the more positive review, the team more likely to win.

To capture the polarity of the sentiment within each piece of text, we use the help of a sentiment lexicon. A sentiment lexicon is a pre-populated vocabulary of words in which every word is measured by how positive or negative they are. We examined a number of different lexicons such as MPQA (Wiebe et al., 2005), and Bing Liu's (Hu and Liu, 2004); after some experimentation we settled on the AFINN (Nielsen, 2011) sentiment dictionary as it had the best results on our data in early trials.

For each textual report, we computed three sentiment features: the number of positive words that existed in the text based on the AFINN lexicon; the number of negative words that existed in the text based on the AFINN lexicon and the percentage difference between the number of positive and negative words in the textual data as calculated in equation 6.1,

$$PosNegDiff = \frac{\#positive\_words - \#negative\_words}{\#words} \tag{6.1}$$

Similar to the textual data, all games had two sentiment analysis vectors, one for the home team and one for the away team. The winning team was given the label "Win" while the losing team was given the label "Loss".

6.2.3 Team Statistical Data

We also created a third dataset of team statistical in-game data similar to the data we used in chapter 3. Given that it is extremely difficult to obtain the historical values of many of these advanced statistics, we opted to only use the three traditional statistics we found to contribute the most to forecasting wins: location ("Home" and "Away"), cumulative goals against and cumulative goal differential. As a reminder, we found that using just these three features, compared to all 12 in chapter 3 was not statistically different. For each set of teams in each game we calculated the values for all three features with a Python script by iterating through the results of the schedule and updating the values prior to each game.

For each game, there were two data vectors representing both teams. The winning team was represented with a "Win" and the losing team represented with a "Loss". The three numerical features were represented as the differential between the two teams.

6.3 Method

For this experiment, we created three multi-level meta-classifiers with the first layer containing three individual classifiers based on the three different sets of data: textual data, sentiment analysis and team statistic data. The outputs of these classifiers were passed into the second layer where we analyzed three different methods of determining the final prediction: voting (where the final decision is based on the majority prediction from the first layer classifiers), highest confidence (chooses the prediction of the classifier that produces the highest confidence on its final prediction) and a cascade classifier (uses a second Machine Learning algorithm to learn from the output of the first layer classifiers). A visual depiction of this model can be see at figure 6.1.

For each of the first layer classifiers, we surveyed a number of different Weka algorithms similar to the ones we used in chapters 3 and 5 including: MultilayerPerceptron (Neural Networks), Naïve Bayes, Complement Naïve Bayes, Multinomial NaiveBayes, LibSVM, SMO (Support Vector Machines), J48 (C4.5 Decision Tree), JRip (rule-based learner), Logistic Regression, SimpleLog and Simple NaiveBayes. The default parameters were used, given the large number of algorithms being surveyed.

We had 708 games to train, resulting in 1416 data vectors; we split this into 66% (930) for training and the remaining 33% (478) for testing. This ensured no single game was split between training and testing and it also allowed us to identify what the confidence

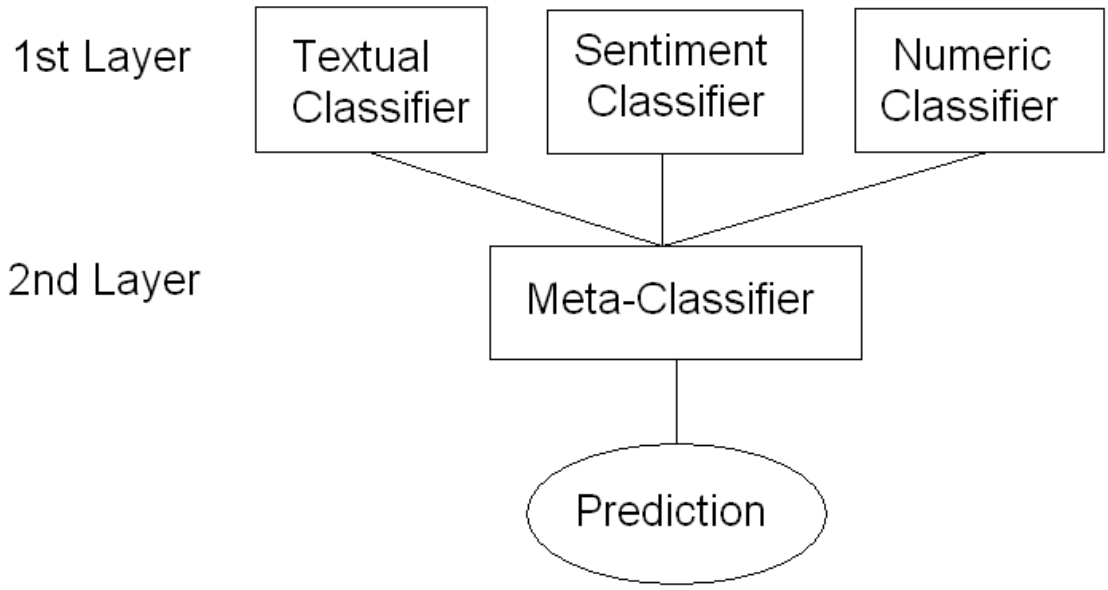

Figure 6.1: Multi-Layer Meta-Classifier

and prediction ("Win" or "Loss') each first layer classifier was producing for each specific testing example.

The results from all three first layer classifiers were post-processed in a format that Weka can read and it was inputted into Weka for the second layer analysis. The features provided to the second layer include the confidence and the predictions for each of the three first layer classifiers, as well as the actual label "Win" or "Loss" (a total of six features and the label). The cascade-classifier applied the same Machine Learning algorithms on the new dataset to learn from the results and accurately make new predictions. The second layer has the results from the 33% of test data so it uses 10-fold cross-validation in order to determine the final accuracy. The meta-classifier uses the same data set, makes predictions for each data vector and then compares it to the actual results.

The best results for each of the first layer classifiers can be seen in table 6.5, while the results of the second layer can be seen in table 6.7. The following sections give details of the two layers within the meta-classifiers.

Table 6.2: Textual Classifier Results on Algorithms

Algorithm	Accuracy
Jrip	57.32%
CNB	55.02%
MNB	55.02%
LibSVM	53.97%
SMO	53.77%
J48	52.93%
NB	52.72%
Logistic Reg	50.21%

6.3.1 Textual Classifier

In the textual classifier (or word-based classifier), we experimented with a number of permutations to represent our textual data from the pre-game reports. We settled on using TF-IDF, no stemmer and 1, 2 and 3 grams. Other options that we analyzed include: Bag-of-Words only, various stemmers and with/without bigrams and trigrams. The best results from our experimentation came from this combination.

In pre-processing the text, all stops words were removed, as well as all punctuation marks (such as single quotation marks). Stopwords were removed based on the Python NLTK 2.0 English stopword list. All text was converted to lowercase.

A number of different Machine Learning classifiers were surveyed on this singular classifier, with the best results being obtained from JRip, the rule-based learner, on the data from both teams. The accuracy achieved on the same test data was 58.32%, just slightly lower than the numeric classifier. The results from all algorithms on the textual data can be seen in table 6.2.

6.3.2 Sentiment Classifier

The second first-layer classifier is the Sentiment Analysis classifier. This classifier uses the number of positive and negative words in the pre-game text, as well as the percentage of positive words differential in the text (as seen in equation 6.1). The three features for each text were forwarded to the Machine Learning algorithms. The highest accuracy was achieved by Naives Bayes at 54.39%, lower than the other two classifiers and the

baseline. The full list of algorithms and their resulting accuracies on this data can be seen in table 6.3.

Table 6.3: Sentiment Analysis Classifier Results on Algorithms

Algorithm	Accuracy
NB	54.39%
Simple NB	53.97%
LibSVM	53.77%
SimpleLog	53.56%
SMO	53.14%
Logistic Regression	52.51%
Jrip	50.63%
MLP	50.21%
J48	50.00%

6.3.3 Numeric Classifier

The third classifier in the first-layer uses the team in-game numerical statistical data. As we saw in chapter 3, the most helpful features to use in a single game prediction are cumulative Goals Against and Differential and Location (Home/Away). For each data vector, we represented the values of both teams as a differential between the two values for each of the three features.

After surveying a number of Machine Learning algorithms in the same format, the best results from this dataset came from using the Neural Networks algorithm MultilayerPerceptron with an accuracy of 58.57%. The full list of algorithms surveyed and their results can be seen at table 6.4.

6.3.4 Meta-Classifier

In the second layer of the meta-classifier, we fed the outputs from each of the three individual first-layer classifiers. As we had separated each of the data into testing and training, we were able to label each game with the confidence and predicted output from the three individual classifiers, as well as the data's actually label. From here,

Table 6.4: Numeric Classifier Results on Algorithms

Algorithm	Accuracy
MLP	58.58%
NB	58.58%
Simple NB	58.58%
Jrip	58.16%
SMO	58.16%
J48	58.16%
Logistic Regression	56.69%
SimpleLog	56.49%
LibSVM	50.00%

Table 6.5: First-Level Classifier Results

Classifier	Algorithm	Accuracy
Numeric	MultilayerPerceptron	58.58%
Text	JRip	57.32%
Sentiment Analysis	NaiveBayes	54.39%

we experimented with three different strategies to determine the final outcome. The first method to use a cascade-classifier where we used Machine Learning algorithms, the second option was to choose the output with the highest confidence, and the third option was to use the majority vote of three classifiers.

With the cascade-classifier approach, we surveyed a number of Machine Learning algorithms, the same ones used for the first layer. The highest accuracy resulted from the SMO algorithm (Support Vector Machines) and it returned an accuracy of 58.58%. The full list of surveyed algorithms can be seen in table 6.6.

For the other two approaches, we used a python script to iterate through each of the data vectors, generate a final decision, and compare it to the actual label. In the method of selecting the outcome based on the highest confidence, the classifier in the first layer which had the highest confidence in its decision was selected. It achieved an accuracy of 57.53%.

Table 6.6: Second-Level Classifier Results

Algorithm	Accuracy
SMO	58.58%
LibSVM	58.58%
SimpleLog	58.16%
J48	57.95%
Logistic Regression	57.64%
NB	57.32%
Jrip	57.32%
Simple NB	56.90%
MLP	56.69%

Table 6.7: Second Level Classifier Results

Method	Accuracy
Cascade Classifier using SVM (SMO)	58.58%
Highest Confidence	57.53%
Majority Voting	**60.25%**

In the second approach, the three classifiers results were forwarded to the second-layer and then a final decision was made based on majority voting rules were to determine the second-layer prediction. That is, the final prediction was the label that at least half of the classifiers were in agreement with. As there are only two options ("Win" or "Loss") then it is mathematically guaranteed that at least two of the classifiers will be in agreement with their decision. This method returned an accuracy of 60.25%. The best results of all three options for the second-layer can be seen in table 6.7.

Using two-tailed paired statistical t-test to compare the differences between these classifiers, the majority-voting classifier is statistically different from the text-classifier ($p = 0.023$), but is not statistically different from the numeric classifier ($p = 0.35$).

For comparison, we placed all three features from all three data sets into a single feature set and surveyed the same Machine Learning algorithms on it. The final accuracies achieved can be see in table 6.8. The best accuracy it achieved was from Naive-

Table 6.8: All-in-One Classifier Results

Algorithm	Accuracy
NaiveBayes	54.47%
NaiveBayesSimple	58.27%
libSVM	51.56%
SMO	53.86%
JRip	54.62%
J48	50.20%

BayesSimple at 58.27%. As all algorithms surveyed used their default parameters, from our experience it may be possible to increase the overall accuracy by 1-2% by tuning these parameters.

6.4 Results

In order to review these results, we need to revisit our previous results. We saw in chapter 3 that we could predict a single game of hockey at 59.8% accuracy. From there, in chapter 4, we looked at the role of random chance within the game and found that there is a theoretical upper bound in single game prediction at 62%. With our data, the baseline was set initially at 50% but as the home team wins 56% of matches this raises the baseline. In total, this only leaves a possible 6% gap (56% - 62%) for improvement, which is a fairly narrow band. Other hockey leagues have a much higher upper bound and we would prefer to apply the methods from this chapter on a different league, but there is a lack of pre-game textual data available. At the time of experimenting and writing, the Swedish Hockey League, with an upper bound close to 70%, was the only other league with a repository of pre-game reports. The issue is there was only a corpus of 100 games worth of pre-text reports, too small of a sample size to draw conclusive evidence from. Additionally, all of the texts are in Swedish, a language without NLP toolsets available.

When we analyze the results of the individual first layer classifiers, the accuracy results are not overly impressive. Sentiment Analysis does worse than the baseline (selecting the home team). This may be due to the fact that the sentiment lexicons were not sport specific and were not able to capture the positive and negative sentiments from

a sport perspective. Using a sports-related lexicon may further increase this accuracy. When we look at only the textual data in the pre-game reports, they show an improvement over the baseline and performs nearly as well as the numerical classifier (which performs best) and is comparable to the results from chapter 3. While the model in chapter 3 uses many more features, including advanced statistics, the numerical model in this chapter has a similar accuracy with only three features, further suggesting both the difficulty in improving single game prediction and the value of these three traditional statistics.

The results become more interesting when we analyze the second layer of the cascade classifier. Using Machine Learning algorithms on the output from the algorithms in the first layer, we see a small improvement. When we look at the methods used in the meta-classifiers, the results improve even more, with majority voting returning the best results at 60.25%. This is an improvement over single game predictions from chapter 3.

It was surprising to see the all-in-one classifier did not perform very well across all of the algorithms that were surveyed with the single dataset. None of the accuracies were very high; except for Naïve Bayes Simple (NBSimple learns from the features modelled with a normal distribution). None of the algorithms surveyed on this dataset were able to perform better than the baseline, which suggests that the mutli-layer meta-classifier is the better method for predicting from these different features sets. The intuition behind this is that the numerical data provides a different perspective and source of knowledge for each game than the textual reports.

We are confident with this method of using a multi-layer meta classifier to forecast success in the NHL and that it can predict with a fairly high accuracy, given the small gap of improvement available (between 56% as the baseline for home field advantage, and 62%, the theoretical upper bound).

For external comparison, we compare these results to the results from `http://www.PuckPrediction.com` and results from "the crowd" (gambling odds). PuckPrediction.com is a fan run website that predicts the outcome of individual hockey games using a proprietary statistical model based on regression techniques and features such as shot differential. So far in the 2013-2014 NHL season, as of 15 February 2013, PuckPrediction.com has made prediction on 870 matches with 509 correct and 361 incorrect, for an accuracy of 58.51%. PuckPrediction.com compares their results to the odds presented from the gambling website `http://www.OddsShark.com` which on the same set of games has correctly predicted 511 matches correctly and 359 incorrectly for an accuracy of 58.74%. While predicting matches on different seasons, our multi-level meta-classifier

has been able to outperform both of these models. Additionally, the accuracies achieved by these external expert models continue to suggest than predicting higher than the upper bound of 62% is not possible.

With the textual data, one interesting aspect to analyze is to see which words are adding the most value to our prediction. We looked at the top 20 features with the highest InfoGain values, which are the attributes that "minimize the information needed to classify the tuples in the resulting partitions and reflect the least 'randomness' or 'impurity' in these partitions" (Han et al., 2006). These values can be seen in table 6.9. As we did not remove team or city names from the text, it is interesting to see that 7 of the top InfoGain values were referring to specific players, coaches, and cities. This list has picked up on the teams of Chicago Blackhawks and Pittsburgh Penguins. Chicago had a very dominant 2012-2013 season and ended up winning the Stanley Cup Championship. Pittsburgh were also considered a strong team in this season. Coach Barry Trotz of the Nashville Predators was a curious pick; he shows up four times in this list despite the fact that the Nashville Predators were neither a top-end nor bottom-end team.

Similarly, we analyzed the decision tree that was generated based off the textual data and we can see a similar trend. The top of the tree is formed by ngrams of players and city names. This can have a dramatic effect on the predictions, if we train on one year where a team is a championship contender and test on the next year when the team may not qualify for the post-season tournament. An example of this is in the 2009-2010 season when the Chicago Blawkhawks were a very strong team and won the Stanley Cup Championship, but in the 2010-2011 season barely made it into the playoffs.

This suggests that it would be difficult to train on text across multiple years, as concept drift would likely appear. Teams may be very good one year, but due to changes in their main roster, via trade or injuries, could become a team at the bottom of the standings the following year. This suggests that we should not be training and testing over multiple seasons. This further backs up the results that we saw in chapter 4 when we looked at testing and training over multiple seasons as per table 4.8.

6.5 Conclusion

In this chapter, we built upon our results found in the previous chapters by expanding upon the model we built in chapter 3. We used three different sources of data to forecast success in a single game. Rather than just in-game team statistical data we focused on textual data written in pre-game reports, as well as sentiment analysis to learn how

Table 6.9: Info Gain values for the word-based features

Info Gain	**ngram**	**Name/Place?**
0.01256	whos hot	No
0.01150	whos	No
0.01124	hot	no
0.00840	three	no
0.00703	chicago	yes
0.00624	kind	no
0.00610	assists	no
0.00588	percentage	no
0.00551	trotz	yes
0.00540	games	no
0.00505	richards said	yes
0.00499	barry trotz	yes
0.00499	barry	yes
0.00499	coach barry	yes
0.00497	given	no
0.00491	four	no
0.00481	pittsburgh penguins	yes
0.00465	body	no
0.00463	save percentage	no

positive (or negative) the reports are. We used TF-IDF values to represent our ngrams text features.

The outputs from the three individual classifiers, from three different data sets, were feed into a second layer meta-classifier which used a couple different methods to make a final decision. We analyzed cascade-classifiers, which use a second machine-learning algorithm, as well as majority of voting and highest confidence. The best results came from majority voting which is able to predict games at 60.25% an improvement over any of the first-level classifiers, as well as the model in chapter 3.

It continues to become more and more difficult to increase our prediction accuracy as we approach the upper bound as found in chapter 4. There is not a lot of room between the baseline, at 56%, and the upper bound, at 62%. This leaves us with a total of 6% for improvement on our classifier. It is interesting to see that text can predict nearly as well as the pure statistical data and combined together this new method can improve the overall prediction limits.

Chapter 7

Conclusion

7.1 Conclusion

In this thesis, we looked at a number of new ways to analyze and make predictions in hockey that can be adapted to other sports. We started by looking at predicting a single game of hockey to determine who will win and who will lose. We analyzed a number of different traditional and advanced statistics to see which help the most with our predictions. The best result, that we were able to obtain, was an accuracy of 59.8% with the most valuable features coming from location, cumulative goals against and cumulative goal differential. Surprisingly, it was the traditional statistics that were shown to be more helpful in a single game, despite previous work showing that advanced statistics are higher correlated with points in the standings and wins over the season.

Because of the difficulty we encountered in predicting a single game, we wanted to further analyze this. We simulated an NHL season multiple times varying the amount of "skill" and random chance ("luck") required to win a game. After various simulations, we found the current NHL is most statistically similar to a league where 24% of games are determined by the better team winning and the remaining 76% of games are a coin flip where either team has an equal chance of winning. This would suggest that a perfect machine learning model would be able to predict all 24% of the "skill" games correctly while guessing half of the "luck" games. This leaves a hypothetical upper bound of approximately 62% for machine learning predictions with hockey.

This makes sense, as random chance plays a large role in the NHL given that the talent levels of teams are so close in parity combined with the low number of events (goals scored) in a single game; a single goal can win or lose the game and goals can be

scored as a result of actions outside of the control of the players. This led us to investigate and question: if predicting a single game is difficult, because of random chance, then over a larger subset of games predictions should become easier as the random chance starts to regress?

We predicted the winner of the best-of-seven playoff series where two teams play a maximum of seven games and the first team to win 4 games advances to the next round. We used a similar method to predicting a single game and expanded our features to include many more traditional and advanced statistics. This time, our results were the opposite as predicting a single game: advanced statistics became more valuable for prediction in the long run and with the increase in prediction accuracy, it is easier to predict the better team over many games than in a single one.

In a final attempt to increase our overall accuracy of predicting a single game, we revisited our original methods. Rather than using only the in game team statistical data, we looked at using textual data in the form of pre-game reports that are published on the official NHL website. Combining the text with sentiment analysis and the traditional data in the form of a multi-level meta-classifier, we were able to increase our overall accuracy ever closer to the upper bound. These results are higher than that from external predictors such as "the crowd" (gambling odds) and third-party proprietary statistical models. We also found that predictions that only used textual data are able to achieve an accuracy better to that of predicting using only numerical data.

At the beginning of the thesis, we presented our hypothesis and it stated:

> Despite hockey's constant flow and its low number of events a game, it is possible to predict the outcome of hockey games automatically, using data mining techniques applied to numerical and textual data, more accurately than the baseline. Though this is difficult because there exists an upper bound in hockey due to the random chance that plays a large role in each game.

We feel confident in our results that we have found sufficient evidence to support our hypothesis.

Hockey is a difficult sport to analyze given the role of random chance that exists in every game, but this is what makes them entertaining for fans around the world to watch. We have learned many new facts about the role of statistics and random chance within sports from this thesis, the biggest one of all is it is not always the best team or player to win. The use of textual data to predict hockey with an accuracy better than

with numeric data was a surprise and may encourage others to use this form of data to predict areas they may have never otherwise been considered.

In addition to sports we hypothesize that using textual qualitative data could help increase prediction accuracies in other stochastic environments. Two examples of these would be using textual data from experts to predict both weather and stock market changes.

7.2 Future Work

There are many directions for future work. A few of the ideas for future work include:

- First, investigate ways to improve the overall accuracy of the individual classifiers in predicting a single game and playoff series. This can be experimented through finding new features that may better correlate to wins in a single game or over the long term. As hockey analytics is still in its infancy, it is highly likely that many features have not yet been identified. Tuning features such as "Fenwick Last-n" to find the optimal length may provide value. Determining the optimal number of games to test and train on could help as well, so the model is not over-fitting a single season while still avoiding concept drift. Alternatively, a different approach could be used for classification, for example reinforcement learning.

- Second, further pre-processing of the data should be explored which would include feature selection to reduce the number of features within the text data as well as experimenting with principal component analysis. While some of our experimentation looked at these methods, we were not able to achieve positive results with it, at this time, but our early experimentation suggests this is a path worth further exploration. Relationships between features need to be explored to see how much of an effect some features might be having on other features and the overall results. One example of this is the relationship between the Z-Rating and the Strength-of-Schedule feature used in the playoff prediction application.

- Third, while predicting a single game has shown to be difficult and predicting over a longer series of games is easier, could we use machine learning to find a way to predict final standings after n games. This would need to be explored to find out how early we can train to obtain results with a high confidence. If we are looking at predicting in the long term, at what point can we predict the post-season champion

with reasonable accuracy? Random chance is likely to play a role in these best of seven series, so this would have an effect on prediction that needs to be explored.

- Next, the upper bound needs to be confirmed. We believe we have established strong evidence that such upper bound exists, but further examination is required. Are we able to pinpoint the exact accuracy limit for our models? Is there an upper bound in predicting a best-of-seven series? Our instinct tells us that there will always continue to be an upper bound, even after playing a large sample of games; however, we would hypothesize from our research that it would be reduced over a large enough sample of games. At this point, we were not able to establish an upper bound for playoff series due to the small sample size of games played and because our method in chapter 4 would not work due to the use of the observed standard distribution of win%. Alternatively methods could be used to explore the upper bound both for the prediction of single games and playoff games through using a Markov Chain Monte Carlo.

- Rather than predicting single games, we want to move our automatic learning from data to predicting the behaviour of individual players and their performance within their teams. There is a vast wealth of traditional and advanced statistics for players as well as plenty of textual reports. These could be used to analyze how they will be performing the next season as they move towards (or away from) their physical peak. It could be used to analyze how a new player might fit in with a team after a trade. These methods could also be used to predict whether a single player is likely to experience an injury due to fatigue, overuse, or for other reasons; or to predict how long it will take a player to recover. While likely to be proprietary to the local sports club, individual player tracking data could be obtained which could further help our analysis. For example, techniques such as swarm intelligence could be used to model player interactions.

- Finally, the last idea we present is probably the most valuable, at least to teams. The way most hockey (and other sports) leagues are established is that every year there is a "Draft" which allows teams to select 18 year old players from developing leagues, and this gives teams exclusive rights to signing that player to contracts for their team in the future. Selecting players through the draft is important because the established rules of leagues make it cheaper and easier to sign contracts with your own talented young players than it is to acquire players off of free agency.

However, selecting players in the draft is difficult, as it is hard to project how an 18 year old will be able to perform at a higher talented league in five to ten years. Every year, in the NHL, teams are able to make seven selections and teams hope that one of those choices will develop into a full-time NHL player. There is only a fraction of the statistical data available for these younger players, but there are plenty of textual data from the scouts who evaluate these young players in order to help aide their organization's data. We want to use Machine Learning combined with these statistical and textual data to see if we can predict promising players better than the current method of using manual qualitative analysis.

Bibliography

ACM. Sigkdd curriculum committee, 2006. URL `\url{http://www.sigkdd.org/sigkdd-curriculum-committee}`.

Patric Andersson, Jan Edman, and Mattias Ekman. Predicting the world cup 2002 in soccer: Performance and confidence of experts and non-experts. *International Journal of Forecasting*, 21(3):565–576, 2005.

Patric Andersson, Daniel Memmert, and Eva Popowicz. Forecasting outcomes of the world cup 2006 in football: Performance and confidence of bettors and laypeople. *Psychology of Sport and Exercise*, 10(1):116–123, 2009.

Burak Galip Aslan and Mustafa Murat Inceoglu. A comparative study on neural network based soccer result prediction. In *Intelligent Systems Design and Applications, 2007. ISDA 2007. Seventh International Conference on*, pages 545–550. IEEE, 2007.

Daniel Barry and JA Hartigan. Choice models for predicting divisional winners in major league baseball. *Journal of the American Statistical Association*, 88(423):766–774, 1993.

Robert H Berk. Continuous univariate distributions, volume 2. *Technometrics*, 38(2): 189–189, 1996.

Steven Bird. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics, 2006.

Phil Birnbaum. Luck vs. talent in the NHL standings. `http://blog.philbirnbaum.com/2013/02/more-luck-in-outcomes-doesnt-imply-less.html`, 2013a. [Online; accessed 17 February 2014].

Phil Birnbaum. More luck in outcomes doesn't imply less skilled players. http://blog.philbirnbaum.com/2013/01/luck-vs-talent-in-nhl-standings.html, 2013b. [Online; accessed 17 February 2014].

Andrew D Blaikie, Gabriel J Abud, John A David, and R Drew Pasteur. NFL & NCAA football prediction using artificial neural networks. 2011.

Bryan L Boulier and Herman O Stekler. Predicting the outcomes of national football league games. *International Journal of Forecasting*, 19(2):257–270, 2003.

Jack Brimberg and William J Hurley. Are national hockey league referees markov&quest. *OR Insight*, 22(4):234–243, 2009.

Brian Burke. Luck and NFL outcomes 3, 2007. URL \url{http://www.advancednflstats.com/2007/08/luc\\k-and-nfl-outcomes-3.html}.

Stefan Büttcher, Charles LA Clarke, and Gordon V Cormack. *Information retrieval: Implementing and evaluating search engines*. Mit Press, 2010.

Samuel E Buttrey, Alan R Washburn, and Wilson L Price. Estimating NHL scoring rates. *J. Quantitative Analysis in Sports*, 7, 2011.

Douwe Buursma. Predicting sports events from past results. 2010.

Chenjie Cao. Sports data mining technology used in basketball outcome prediction. 2012.

Bradley P Carlin. Improved NCAA basketball tournament modeling via point spread and team strength information. *The American Statistician*, 50(1):39–43, 1996.

Cam Charron. Breaking news: Puck-possession is important (and nobody told the cbc). http://blogs.thescore.com/nhl/2013/02/25/breaking-news-puck-possession-is-important-and-nobody-told-the-cbc/, 2013. [Online; accessed 12-April-2013].

Hsinchun Chen, Peter Buntin Rinde, Linlin She, Siunie Sutjahjo, Chris Sommer, and Daryl Neely. Expert prediction, symbolic learning, and neural networks. an experiment on greyhound racing. *IEEE Expert*, 9(6):21–27, 1994.

Wikimedia Commons. Svm max sep hyperplane with margin, 2009. URL http://upload.wikimedia.org/wikipedia/commons/2/2a/Svm_max_sep_hyperplane_with_margin.png. File: Svm_max_sep_hyperplane_with_margin.png.

Anthony C Constantinou, Norman E Fenton, and Martin Neil. pi-football: A bayesian network model for forecasting association football match outcomes. *Knowledge-Based Systems*, 2012.

Anthony Costa Constantinou. Bayesian networks for prediction, risk assessment and decision making in an inefficient association football gambling market. 2012.

Scott Cullen. Which NHLers are getting the bounces this year?, April 2014. URL `http://www.tsn.ca/blogs/scott_cullen/?id=448092/`. [Online; posted 02-April-2014].

John A David, R Drew Pasteur, M Saif Ahmad, and Michael C Janning. NFL prediction using committees of artificial neural networks. *Journal of Quantitative Analysis in Sports*, 7(2), 2011.

Dursun Delen, Douglas Cogdell, and Nihat Kasap. A comparative analysis of data mining methods in predicting NCAA bowl outcomes. *International Journal of Forecasting*, 28 (2):543–552, 2012.

Patrick Dermody. Studying luck & other factors in PDO. `http://nhlnumbers.com/2013/1/10/studying-luck-other-factors-in-pdo`, 2013. [Online; accessed 12-April-2013].

Gabriel Desjardins. League equivalencies, 2009. URL `http://hockeyanalytics.com/Research_files/League_Equivalencies.pdf`.

Ap Dijksterhuis, Maarten W Bos, Andries Van der Leij, and Rick B Van Baaren. Predicting soccer matches after unconscious and conscious thought as a function of expertise. *Psychological Science*, 20(11):1381–1387, 2009.

Nicholas Emptage. Fooled by randomness: Why we know less than we think about chance variation in hockey, 2013. URL `http://puckprediction.com/2013/10/09/the-problem-with-random-distributions-in-the-study-of-luck`.

Vic Ferrari. Zone time, corsi and correlation to winning. `http://vhockey.blogspot.ca/2008/08/zone-time-corsi-and-correlation-to.html`, August 2008. Accessed: 2014-02-06.

Michael H Freiman. Using random forests and simulated annealing to predict probabilities of election to the baseball hall of fame. *Journal of Quantitative Analysis in Sports*, 6(2), 2010.

Jamin Brett Halberstadt and Gary M Levine. Effects of reasons analysis on the accuracy of predicting basketball games1. *Journal of Applied Social Psychology*, 29(3):517–530, 1999.

Jiawei Han, Micheline Kamber, and Jian Pei. *Data mining: concepts and techniques*. Morgan kaufmann, 2006.

Evan Heit, Paul C Price, and Gordon H Bower. A model for predicting the outcomes of basketball games. *Applied cognitive psychology*, 8(7):621–639, 1994.

Trent J Herda, Eric D Ryan, Jason M DeFreitas, Pablo B Costa, Ashley A Walter, Katherine M Hoge, Joseph P Weir, and Joel T Cramer. Can recruiting rankings predict the success of NCAA division i football teams? an examination of the relationships among rivals and scouts recruiting rankings and jeff sagarin end-of-season ratings in collegiate football. *Journal of Quantitative Analysis in Sports*, 5(4), 2009.

Adam Hipp and Lawrence Mazlack. Mining ice hockey: Continuous data flow analysis. In *IMMM 2011, The First International Conference on Advances in Information Mining and Management*, pages 31–36, 2011.

Vincent Hoekstra, Pieter Bison, and Guszti Eiben. Predicting football results with an evolutionary ensemble classifier. 2012.

Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM, 2004.

Kou-Yuan Huang and Wen-Lung Chang. A neural network method for prediction of 2006 world cup football game. In *Neural Networks (IJCNN), The 2010 International Joint Conference on*, pages 1–8. IEEE, 2010.

Kou-Yuan Huang and Kai-Ju Chen. Multilayer perceptron for prediction of 2006 world cup football game. *Advances in Artificial Neural Systems*, 2011:11, 2011.

Josip Hucaljuk and A Rakipovic. Predicting football scores using machine learning techniques. In *MIPRO, 2011 Proceedings of the 34th International Convention*, pages 1623–1627. IEEE, 2011.

Z Ivankovic, M Rackovic, B Markoski, D Radosav, and M Ivkovic. Analysis of basketball games using neural networks. In *Computational Intelligence and Informatics (CINTI), 2010 11th International Symposium on*, pages 251–256. IEEE, 2010.

Marshall B Jones. Responses to scoring or conceding the first goal in the NHL. *Journal of Quantitative Analysis in Sports*, 7(3):15, 2011.

A Joseph, NE Fenton, and M Neil. Predicting football results using bayesian nets and other machine learning techniques. *Knowledge-Based Systems*, 19(7):544–553, 2006.

Joshua Kahn. Neural network prediction of NFL football games. *World Wide Web electronic publication, URL homepages. cae. wisc. edu/˜ ece539/project/f03/kahn. pdf*, 2003.

Bernard Loeffelholz, Earl Bednar, and Kenneth W Bauer. Predicting NBA games using neural networks. *Journal of Quantitative Analysis in Sports*, 5(1):1–15, 2009.

Arlo Lyle. Baseball prediction using ensemble learning. 2007.

Brian Macdonald. An expected goals model for evaluating NHL teams and players. In *Proceedings of the 2012 MIT Sloan Sports Analytics Conference, http://www. sloansportsconference. com*, 2012.

B Markoski, P Pecev, L Ratgeber, M Ivkovic, and Z Ivankovic. Appliance of neural networks in basketball-basketball board for basketball referees. In *Computational Intelligence and Informatics (CINTI), 2011 IEEE 12th International Symposium on*, pages 133–137. IEEE, 2011.

Ian McHale and Alex Morton. A bradley-terry type model for forecasting tennis match results. *International Journal of Forecasting*, 27(2):619–630, 2011.

Nicholas Metropolis. The beginning of the Monte Carlo method. *Los Alamos Science*, 15(584):125–130, 1987.

Nicholas Metropolis and Stanislaw Ulam. The Monte Carlo method. *Journal of the American statistical association*, 44(247):335–341, 1949.

D Miljković, L Gajić, A Kovačević, and Z Konjović. The use of data mining for basketball matches outcomes prediction. In *Intelligent Systems and Informatics (SISY), 2010 8th International Symposium on*, pages 309–312. IEEE, 2010.

Katherine Miller. Predicting wins for baseball games. *Liberal Arts Scholarly Repository*, 2011.

Brian M Mills and Steven Salaga. Using tree ensembles to analyze national baseball hall of fame voting patterns: An application to discrimination in bbwaa voting. *Journal of Quantitative Analysis in Sports*, 7(4), 2011.

Byungho Min, Jinhyuck Kim, Chongyoun Choe, Hyeonsang Eom, and RI McKay. A compound framework for sports results prediction: A football case study. *Knowledge-Based Systems*, 21(7):551–562, 2008.

T Mitchell. Machine learning. 1997.

Leonard Mlodinow. *The drunkard's walk: How randomness rules our lives*. Random House LLC, 2009.

Stuart Morgan, Morgan David Williams, and Chris Barnes. Applying decision tree induction for identification of important attributes in one-versus-one player interactions: A hockey exemplar. *Journal of sports sciences*, (ahead-of-print):1–7, 2013.

Blake Murphy. Exploring marginal save percentage and if the canucks should trade a goalie. `http://www.nucksmisconduct.com/2013/2/13/3987546/exploring-marginal-save-percentage-and-if-the-canucks-should-trade-a`, 2013. [Online; accessed 12-April-2013].

Paul K Newton and Kamran Aslam. Monte Carlo tennis: a stochastic markov chain model. *Journal of Quantitative Analysis in Sports*, 5(3):1–42, 2009.

Finn Årup Nielsen. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. *arXiv preprint arXiv:1103.2903*, 2011.

JP Nikota. Leaf's attack time at the halfway mark. `http://www.pensionplanpuppets.com/2013/9/16/4727746/leafs-attack-time-at-the-halfway-mark`, September 2013. Accessed: 2014-04-29.

Nils J Nilsson. Introduction to machine learning. an early draft of a proposed textbook. 1996.

Melvin R Novick. The axioms and principal results of classical test theory. *Journal of Mathematical Psychology*, 3(1):1–18, 1966.

Sérgio Nunes and Marco Sousa. Applying data mining techniques to football data from european championships. In *Actas da 1ª Conferência de Metodologias de Investigação Científica (CoMIC'06)*, 2006.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics, 2002.

Michael Pardee. An artificial neural network approach to college football prediction and ranking. *University of Wisconsin–Electrical and Computer Engineering Department*, 1999.

John Platt et al. Sequential minimal optimization: A fast algorithm for training support vector machines. 1998.

Joel Poualeu, Yu-Han Chang, and Rajiv Maheswaran. Data mining and basketball games. 2011.

Michael C Purucker. Neural network quarterbacking. *Potentials, IEEE*, 15(3):9–15, 1996.

John Ross Quinlan. *C4. 5: programs for machine learning*, volume 1. Morgan kaufmann, 1993.

José Manuel Sánchez Santos, Ana Belen Porto Pazos, and Alejandro Pazos Sierra. Team performance in professional basketball: an approach based on neural networks and genetic programming. In *XIII IASE and III ESEA Conference of Sports, Prague*, 2011.

Mohammadi Sardar and Nasim Salehi. Forecasting the ranking of iran's national football team by fifa. two predicting models: Artificial neural networks and autoregressive-integrated moving average. *Review of Leadership Levels of the Employees Working at the Coordination Center Forworld University Winter Games 9*, page 93, 2012.

Saed Sayad. An introduction to data mining, 2012. URL \url{http://www.saedsayad.com/}.

Benjamin Scheibehenne and Arndt Bröder. Predicting wimbledon 2005 tennis results by mere player name recognition. *International Journal of Forecasting*, 23(3):415–426, 2007.

Neil C Schwertman, Kathryn L Schenk, and Brett C Holbrook. More probability models for the NCAA regional basketball tournaments. *The American Statistician*, 50(1): 34–38, 1996.

Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47, 2002.

Sascha Serwe and Christian Frings. Who will win wimbledon? the recognition heuristic in predicting sports events. *Journal of Behavioral Decision Making*, 19(4):321–332, 2006.

Phil Simon. *Too Big to Ignore: The Business Case for Big Data*. John Wiley & Sons, 2013.

Shiladitya Sinha, Chris Dyer, Kevin Gimpel, and Noah A Smith. Predicting the NFL using twitter. September 2013.

Lloyd Smith, Bret Lipscomb, and Adam Simkins. Data mining in sports: Predicting cy young award winners. *Journal of Computing Sciences in Colleges*, 22(4):115–121, 2007.

Tyler Smith and Neil C Schwertman. Can the NCAA basketball tournament seeding be used to predict margin of victory? *The american statistician*, 53(2):94–98, 1999.

ChiUng Song, Bryan L Boulier, and Herman O Stekler. Measuring consensus in binary forecasts: NFL game predictions. *International Journal of Forecasting*, 25(1):182–191, 2009.

Tim B Swartz, Aruni Tennakoon, Farouk Nathoo, Min Tsao, and Parminder Sarohia. Ups and downs: team performance in best-of-seven playoff series. *Journal of Quantitative Analysis in Sports*, 7(4), 2011.

Tom Tango. True talent levels for sports leagues. `http://www.insidethebook.com/ee/index.php/site/comments/true_talent_levels_for_sports_leagues/`, 2006. [Online; accessed 16 February 2014].

Matthew Tucker. Football match result predictor website. 2011.

Alan M Turing. Computing machinery and intelligence. *Mind*, pages 433–460, 1950.

Ben Van Calster, Tim Smits, and Sabine Van Huffel. The curse of scoreless draws in soccer: the relationship with a team's offensive, defensive, and overall performance. *Journal of Quantitative Analysis in Sports*, 4(1):1–22, 2008.

Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511. IEEE, 2001.

Nikolaos Vlastakis, George Dotsis, and Raphael N Markellos. Nonlinear modelling of european football scores using support vector machines. *Applied Economics*, 40(1): 111–118, 2008.

Na Wei. Predicting the outcome of NBA playoffs using the naïve bayes algorithms. 2011.

Joshua Weissbock and Diana Inkpen. Combining textual pre-game reports and statistical data for predicting success in the national hockey league. Proceedings from 2014 Canadian Conference on Artificial Intelligence, May 2014.

Joshua Weissbock, Herna Viktor, and Diana Inkpen. Use of performance metrics to forecast success in the national hockey league. Proceedings from ECML/PKDD 2013 Workshop on Machine Learning and Sports Analytics, September 2013.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2-3):165–210, 2005.

Rick L Wilson. Ranking college football teams: A neural network approach. *Interfaces*, 25(4):44–59, 1995.

Ian H Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.

Jackie B Yang and Ching-Heng Lu. Predicting NBA championship by learning from history data. 2012.

Travis Yost. Luck vs. talent. `http://www.hockeybuzz.com/blog.php?post_id=53415`, 2013. [Online; accessed 16 February 2014].

William A Young, William S Holland, and Gary R Weckman. Determining hall of fame status for major league baseball using an artificial neural network. *Journal of Quantitative Analysis in Sports*, 4(4):1–44, 2008.

Albrecht Zimmermann, Sruthi Moorthy, and Zifan Shi. Predicting college basketball match outcomes using machine learning techniques: some results and lessons learned. September 2013.