From the collections of the Princeton University Archives, Princeton, NJ

Statement on Copyright Restrictions

This senior thesis can only be used for education and research (among other purposes consistent with "Fair use") as per U.S. Copyright law (text below). By accessing this file, all users agree that their use falls within fair use as defined by the copyright law. They further agree to request permission of the Princeton University Library (and pay any fees, if applicable) if they plan to publish, broadcast, or otherwise disseminate this material. This includes all forms of electronic distribution.

U.S. Copyright Law (Title 17, United States Code)

The copyright law of the United States governs the making of photocopies or other reproductions of copyrighted material. Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or other reproduction is not to be "used for any purpose other than private study, scholarship or research." If a user makes a request for, or later uses, a photocopy or other reproduction for purposes in excess of "fair use," that user may be liable for copyright infringement.

Inquiries should be directed to:

Princeton University Archives
Seeley G. Mudd Manuscript Library
65 Olden Street
Princeton, NJ 08540
609-258-6345
mudd@princeton.edu

Combating Uncertainty with Context: Optimal Lineup Construction in Daily Fantasy Baseball

Jacob S. Eisenberg

Advisded by Professor Amir Ali Ahmadi

Submitted in partial fulfillment
of the requirements for the degree of
Bachelor of Science in Engineering
Department of Operations Research and Financial Engineering
Princeton University

June 2016

I hereby declare that I am the sole author of this thesis

I authorize Princeton University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

_____

Jacob S. Eisenberg

I further authorize Princeton University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

_____

Jacob S. Eisenberg

Abstract

Over the past several years Daily Fantasy Sports have emerged as a huge industry, with participants being able to compete against one other for money on a daily basis. Here we investigate how to construct an optimal lineup in Daily Fantasy Baseball specifically, using the FanDuel scoring system and lineup restrictions. The daily timescale poses the challenge of increased variability in player performance, but also allows us to add more context to our predictions by leveraging game-day information. We develop two prediction models, one for hitter point totals and one for pitcher point totals. In both cases a Generalized Boosted Model performed best with Root Mean Squared Errors of 3.014 and 5.37 respectively. We then use these predictions as an input into an Integer Program that solves for the lineup with the highest expected point total subject to the salary cap and the other constraints FanDuel imposes. This two-step process of prediction and optimization resulted in an average actual lineup point total (as opposed to the value of the objective function which is expected points) of 42.60, which was 5.53 points higher than a 'naive' prediction approach that only used a player's average Points per Game and ignored game-day factors. Furthermore this point total falls above the 70th percentile of historical point totals in Head-to-Head and 50/50 contests suggesting our approach should win the majority of the time. We then modify our optimization approach to account for the role of lineup variance in the context of different contest types drawing on concepts from Portfolio Optimization such as the efficient frontier and the Capital Market Line. As part of this process we quantify the covariance between player point totals. We see that in Head-to-Head contest one should choose the lineup with the highest expected point total with no regard for lineup variance. In 50/50 contests one should limit lineup variance to an extent due to the flat payout structure that does not reward extreme point totals. For tournaments one needs to pick a lineup with higher variance in order to have a chance at reaching the high point totals that are necessary to win among the thousands of entries. Finally, future work is discussed.

Acknowledgments

I would first like to thank Professor Amir Ali Ahmadi for advising my thesis and providing advice and support throughout the process. From ORF 363 to my junior independent work to this project, my interest in optimization has been instilled and shaped by your teaching. I can only hope I have been half as good of a teacher when it comes to the rules of baseball. Your insights and feedback were invaluable in guiding my thesis and I am grateful to have had the opportunity to work with you over the past two years.

Furthermore I need to acknowledge my friends and classmates for helping me along the way. To my brothers in Chi Phi, you have shaped my time at Princeton and have made these four years a truly unforgettable experience. Thank you for making Mondays my favorite day of the week. To my friends in TI, you have provided me with a home away from home on campus, and for that I will always be grateful. Finally to my fellow ORFE majors, I have learned so much from you guys and could not have made it through this process without you.

Lastly I would not be where I am today without the undying love and support of my family. Mom and Dad, thank you for the endless phone calls and encouragement and for always believing in me. Sophie, thank you for providing solidarity all the way from Ann Arbor and for always helping me stay positive. To my Grandparents, thank you for continually inspiring me and being the best role models I could ask for.

Contents

Chapter 1

Introduction

1.1 Overview of Fantasy Sports

The central premise of Fantasy Sports is that an individual can virtually construct a 'team' of professional athletes and receives points based on how well those players perform in reality. This gives fans the opportunity to manage their own team, an experience previously reserved for the few actual team executives. It also provides a fun and exciting way to compete with friends and showcase one's knowledge of the game. Since its inception several decades ago, Fantasy Sports has grown into a huge industry that has captured the interest of over 50 million fans [1]. This growth has been especially explosive over the past decade with the development of the Internet and other technological advances. The drafting and scoring systems are now automated and can be updated in real time, and one can compete against others remotely. Furthermore, being able to use a mobile device to change a lineup or check scores has added convenience [1].

One of the primary consequences of the immense popularity of Fantasy Sports is that it has increased fan engagement. Games that would have otherwise been uninteresting to a given fan suddenly become entertaining if a player in that game is in their fantasy lineup. This has manifested itself in 60% of Fantasy participants watching more live sports as a result of their involvement [1]. Companies in the sports industry, from the leagues themselves, to sports websites, to the TV networks that broadcast games, have all benefited from this. The TV companies increase revenue through higher viewership (helping the leagues as well), and the large Fantasy Sports providers generate revenue through advertisements and offering paid subscriptions to expert analysis and advice. There has also been a general effort by companies to provide more fantasy-specific content to attract this still-growing community of Fantasy Sports participants [2]. Notably, very little revenue is generated directly from people paying to play Fantasy Sports. While there are leagues with entry fees (with the winner receiving a prize), the vast majority of users play for free. In fact ESPN, one of the largest Fantasy Sports providers, recently announced that they will be ending all prize-eligible leagues [3]. It is this direct monetization of Fantasy Sports that Daily Fantasy Sports providers have capitalized on, with staggering results.

1.2 The Emergence of Daily Fantasy Sports

While Daily Fantasy Sports (DFS) remain true to the principals of Fantasy Sports at the highest level, there are two key differences that set it apart. One involves how a fantasy team is assembled. The traditional process by which a participant constructs a team is through a fantasy draft with the other members of their league. A fantasy league usually consists of 8-12 participants, and each league is completely separate from all other leagues. The draft involves each participant sequentially choosing players for their fantasy team until they fill their roster. As a result, no two teams in the same league can have the same player. This can be frustrating, as a participant may want a specific player for either sentimental or strategic purposes, but that player was already selected by another league member. The procedure for selecting a team in DFS is very different. Each participant is given a virtual budget and each player is assigned a virtual salary based on the DFS provider's valuation of that player on that day. It is then up to the participant to construct a valid lineup within the budget. So while participants still form leagues (or contests as DFS sites call them) with others, it is possible that two participants within the same contest can have the same lineup. This gives participants complete autonomy when choosing a team.

Of course, the most striking feature of DFS is its daily nature. Traditionally, a participant will draft a team before the season starts, and with the exception of any trades with other participants in the same fantasy league, that will be their team for the duration of the season. As its name suggests, DFS contests only last for one day, meaning a participant can choose a completely new team the next day. The short-term nature of this commitment has proved very appealing for a couple of reasons. In a season-long league, a team can be impacted greatly by factors outside of the participant's control, such as injuries, which can be frustrating and at the same time not immediately addressable. DFS allows users to hit the reset button every day, and if they don't have time to choose a lineup on a given day there are no negative consequences. In addition, the daily timescale amplifies the interest and excitement surrounding player performance. Over the course of a season, which is more than five months in baseball, an individual day is not particularly important. But if someone is playing DFS, anytime they have a player in a game it becomes 'must-see-TV.' It is primarily for this reason that the professional leagues and TV networks have welcomed the emergence of DFS [2].

The daily contests have also been crucial to the DFS providers' ability to significantly monetize Fantasy Sports. Instead of participants playing in one season-long league, they can instead play in over 100 daily leagues over the course of the season, paying an entry fee each day. The business model of DFS providers is that the majority of the entry fees for a contest go to the winners of the contest (payout structure depends on contest type) with the provider taking a constant percentage of all entry fees. Entry fees can range from $1 up to over $100. With respect to revenue, DFS has exploded over the past three years. While in 2012 the average adult who participated in Fantasy Sports was spending $5/year on DFS, in 2015 that number had increased by more than 5000% to $257 [1]. The two main DFS sites, FanDuel and DraftKings, pay out over $1 billion each annually, and are both valued at over $1 billion [4]. It is clear that DFS is no longer simply a new variation of Fantasy Sports, but it has become its own industry.

1.3 Motivation

Even though gambling is illegal in the United States, DFS providers have been allowed to issue payouts to winners as it has been judged that DFS is a 'game of skill', and thus exempt from the Unlawful Internet Gambling Enforcement Act of 2006 [4]. While the legality and lack of regulation of DFS have recently come under fire after what amounted to insider trading within the industry [5], this paper takes no stance on the issue. However, this paper is motivated by the veracity of the same claim, that DFS is indeed a 'game of skill.' If there were little to no skill involved, attempting to develop a mathematical process to construct an optimal lineup would be analogous to trying to predict a coin flip. But studies have shown that winning in DFS is far from a toss up. One fascinating statistic that provides strong evidence that there exists a methodology that usually produces a profit is that 90% of profits in the first half of the 2015 MLB season were won by 1.3% of participants [6]. While this is slightly misleading, as the contests with extremely high payouts can only have one winner, and these contests represent a significant amount of the winnings, it still shows that if a participant wins they tend to win often.

We are by no means the first to attempt to develop a process for constructing a lineup in Daily Fantasy Sports. There are several companies that have gotten into the business of providing participants with projections, advice, and in some cases even the lineup that they 'should' choose [7, 8]. While some charge a small fee, which may prevent some users from accessing them, most are free well-known sites that are frequented by many DFS participants. However, if these sources were providing lineups that were optimal, or at least not far from it, we would expect the payout distribution mentioned in the preceding paragraph to be far less heavily skewed than it is, as many participants would have access to this information. Even if we concede that there may be some negative game theoretic implications of the resulting herd-mentality that these websites can cause (see Chapter 6), the persistent lack of parity suggests that the algorithms and models that have been developed outside of the public sphere provide better lineups. Of course, the developers of these winning techniques have a strong financial incentive to keep them proprietary. We attempt to develop a novel approach (at least in the public domain) to DFS lineup construction that produces results on par with, and if not better than, the current top performers.

1.4 Data Sources and Availability

Baseball research has benefited tremendously from the vast and increasingly sophisticated data sets that have been developed over the lifetime of the game itself. From simple box scores that summarize the events of a game to complex tracking data that can provide information about the trajectory of a pitch, there is a wealth of data at our disposal. However the data on Daily Fantasy Baseball specifically is far less comprehensive. With the industry only gaining widespread popularity in the last few years the size of the data set about player point totals and the DFS salaries that are assigned to players is not very large. Furthermore DFS providers do not compile or openly present this data. Of course in order to train a prediction model and constrain our optimization problem we will need player point totals and DFS salaries respectively. To increase the size of our data set it is necessary to develop a way to determine what a player's point total and DFS salary would have been had the industry existed. Because DFS providers have a very clear and specific formula for calculating point

totals it is straightforward to impute these using Retrosheet play-by-play data as far back as we wish [9]. Unfortunately the algorithm DFS providers use to determine what a player's salary should be on a given day is proprietary, so we cannot impute it exactly. Our solution to this is described in detail in Chapter 5.

We obtained any game-specific data from Retrosheet play-by-play data, and supplemented it with season-level data from the Lahman database and FanGraphs [10, 11]. In terms of the size of our data set we use all MLB regular season games played from 2003-2013. Noting that we are hoping to make inferences from this data it is important to consider to what extent the game of baseball has changed over the last decade. The largest macro trend has been an increased rate of strikeouts and suppressed run scoring in general [12]. To try to avoid an issues this may cause with prediction we normalized all statistics to the league average for that season. Finally this paper is only concerned with constructing an optimal lineup for the DFS provider FanDuel. While one could apply the general framework of this analysis to DraftKings, the differences between the two are such that we could only focus on one.

1.5 Outline

The ultimate goal of this paper is to develop a systematic process to construct an optimal lineup in Daily Fantasy Baseball. It is most intuitive to think about this process starting from its culmination. Let us assume that we (and no one else) somehow knew exactly how every player was going to perform on a given day. If this were the case, we would formulate an Integer Program with the objective being to maximize points and the constraints simply making sure that the chosen lineup satisfies the salary cap constraint and the few other constraints that FanDuel imposes on lineups (See Chapter 4). The decision variable would be which players to choose for our lineup. This approach would result in a winning lineup 100% of the time, as it is computationally tractable and deterministic in nature. So what this process truly boils down to is predicting player performance on a given day. The framework of the optimization problem is straightforward, but its solution will only be as useful as the predictions used as input.

In Chapter 2 we discuss the explanatory variables and methodology used for predicting hitter performance. In Chapter 3 we perform the analogous analysis for pitchers. Chapter 4 develops the framework for the optimization problem described above and also compares the performance of our process with several sensible benchmarks. Chapter 5 examines how we should alter our strategy based on the contest type, emphasizing the role of player covariance and lineup volatility in selecting an optimal lineup. Finally Chapter 6 discusses areas of future work and presents a conclusion.

Chapter 2

Predicting Hitter Performance

2.1 The Pros and Cons of Daily Prediction

Predicting how a baseball player will perform over the course of a season has been a heavily researched topic for obvious reasons. Several robust projection systems have been developed in the public domain [13] and MLB teams have invested significant resources into building their own forecasting models as well [14]. While these models are certainly useful they are still unable to provide accurate predictions across the board. This is partially a result of how unpredictable of a game baseball is by nature. 162 games may seem like a long time, but the degree of randomness in the game is such that a player's outcomes may not represent his true talent, even over the course of a season. Attempting to predict a player's performance on a given *day* is even more difficult as there is minimal opportunity for randomness to be washed out. For example, let's say a hitter makes great contact on a ball, but ends up hitting it directly at a fielder. Over the course of a season this will somewhat be compensated for by weakly hit balls that find a hole, but on a daily basis this is not the case. This is further exacerbated by the discrete nature of events in baseball and the relatively small number of opportunities a hitter receives during a game. For example a great hitter might hit a home run every 15 plate appearances. But in one game a hitter almost never gets more than five plate appearances. Having a binary outcome like this (a hitter cannot hit half of a home run) further increases the uncertainty of daily prediction. At this point it may seem like an exercise in futility to predict daily performance, and while the shorter timescale does lead to more variability it also allows us to add more context to our predictions.

Before the season begins there are only a few factors that one can consider when developing a prediction model. The most important is some measure of a player's past performance, usually adjusted for age. One can also consider the effect of that player's home ballpark on different outcomes, as they will be playing half of their games there. While there may be some other more subtle factors that are included, the inputs into a projection of season-long performance cannot be much more detailed than this. For example it is implicitly assumed that a hitter will face average (within their league) pitching over the course of the season, as it is impossible to predict who the hitter will actually end up facing. One can imagine how useful this information would be to refining predictions. Our expectations for a hitter would be markedly different if we were told he would be facing the best pitcher in the league every at bat. This is where we can reap the benefits of the daily timescale.

These benefits come from being able to know detailed and accurate information that is only possible to know the day of the game, such as the opposing pitcher. A sensible analogy is weather prediction. If someone were asked in July to predict the weather on December 1st, his or her best response would presumably be the historical average temperature on that day. But if instead they were asked on November 30th, they would have more detailed and timely information, leading to a more accurate prediction. In order for our predictions to effectively combat the increased variability present over the shorter timescale we must leverage the external game conditions that can only be known on that day.

2.2 Exploratory Data Analysis

Before delving into model selection and fitting, it is useful to first perform some exploratory analysis of the data. A sensible place to start is the distribution of player point totals. Players receive (or lose) points for various offensive events that they are involved in. A description of the scoring system for hitters can be seen in Table 2.1 [15]. So for example if a player had four at bats resulting in a single, a HR, two outs, a run, and 2 RBIs, he would receive 7.5 points for that day. With this context in mind we can now examine the density of player point totals shown in Figure 2.1 (next page).

Event	Points	Event	Points
Single	1	Walk	1
Double	2	HBP	1
Triple	3	RBI	1
Home Run	4	Run	1
Out	-0.25	Stolen Base	2

Table 2.1: FanDuel Scoring System for hitters

The most striking feature of this distribution is how right-skewed it is (Skewness = 1.36). While the bulk of the density is between zero and three points, players have obtained scores as high as 30. The reason for this becomes clear when looking at the scoring system in the context of the number of opportunities a hitter will have during a game. First, events that lead to positive points are weighted more heavily than the negative event of recording an out due to their relative rarity. Additionally, in a game a hitter is guaranteed three plate appearances, but can sometimes have up to five or six if their team is scoring a lot of runs. So even if they record an out every time, their score will not drop below -1.5, which is what we see. On the other hand, if they happen to produce several rare positive events their score can climb quickly.

In Chapter 5 we will find it useful to sample from this distribution to better understand the variance of a lineup's point total. This means we need to fit a distribution to this empirical density. It is crucial that any distribution we choose is able to capture the very heavy right tail of the data, as even though these point totals are very rare they have a significant impact on the probability of winning a contest if they do occur. It seems that a Generalized Pareto Distribution would be useful here [16]. A semi-parametric approach, this family of distributions allows the bulk of he distribution to be fit non-parametrically, while at the same time fitting the tail of the distribution with a shape parameter that controls the rate of decay. Pareto distributions have significantly heavier tails than

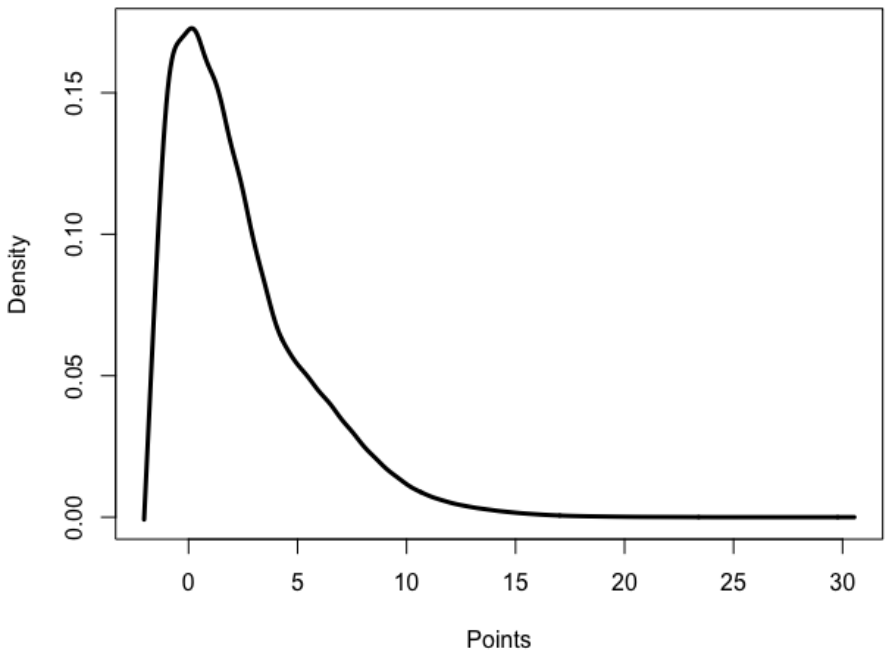

Figure 2.1: Density of Hitter Point totals

classical one-tailed parametric distributions like the exponential or log-normal distribution, as their rate of decay is polynomial in x rather than exponential. We leave the technical details until Chapter 5.

In addition to how many points players receive it is useful to understand how volatile their point totals are. While we will formalize this idea in Chapter 5, one can think of each player as a financial asset with an expected return (average point total) and risk (standard deviation in point total) associated with it. Ideally we would like to choose players with high return and low variability while avoiding players with low return and low variability. However when we look at the relationship between a player's average point total and their variance in Figure 2.2 (next page) we see that there is a constant trade off between risk and return. Our ideal player does not exist, which should not surprise us when we consider the distribution of point totals discussed above. The players with high averages are of course the better players, meaning they have really good games more often (but still not often), thus increasing their volatility. The R^2 of the linear regression of standard deviation on the average is 0.72. Further evidence that this relationship is borne out of the skewness of point totals is that when we replace the average with the median for each player the R^2 is reduced to 0.28 and the slope of the line is halved. This is simply a manifestation of the fact that in baseball even the best players fail most of the time. Understanding what external factors are conducive to players having that rare very productive game will be the key to constructing a successful DFS lineup.

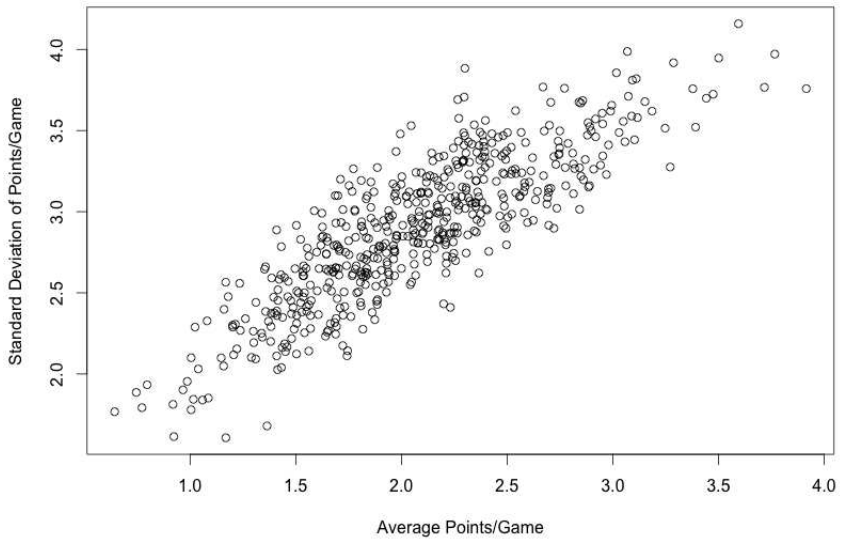

Figure 2.2: Relationship between average point total and Standard Deviation

2.3 Potential Explanatory Variables

As was mentioned in the Introduction the crux of this problem is developing a robust predictive model for player point totals. While the structure and assumptions of whatever type of model we use will be important, our success will be dictated by our ability to choose meaningful explanatory variables. Here we outline the explanatory variables we use and why they are important to predicting point totals.

Average Point Total: There is obviously a difference in skill among individual players. It is not an unbelievably large difference on the scale of an individual game, as the worst player's best game is no doubt significantly better than the best player's worst game. But our priors about individual players will be different than simply league average performance, meaning we will need a variable that accounts for each player's baseline production. The next question is what do we mean by production? The answer to this question depends entirely on the context. In our context the obvious choice is to use a player's average points per game (PPG). We should note that PPG is not the best metric by which to judge a player's overall talent or contribution, as it is very context-dependent. There has been an effort in the baseball research community to develop metrics that attempt to isolate the contribution of an individual [17], but those are only useful so far as they help predict DFS point totals.

The Opposing Pitcher: There is no doubt that the opposing pitcher is the most influential external game-day factor. A dominant starting pitcher can greatly suppress the production of hitters, and we can only account for this the day of the game. The key to getting the most out of this variable is understanding that players with different skill-sets will be impacted differently by different pitchers.

For example, let's say there is a hard-throwing pitcher who strikes out a lot of batters. If the batter is someone who also strikes out a lot, this matchup will only amplify that tendency [18]. Other skills to investigate are the ability of a pitcher to generate ground balls or fly balls, as well as their accuracy. If a well-disciplined hitter faces a wild pitcher, the probability of a walk will also be higher than normal for both parties. In terms of what specific statistics to use, it is clear that K%, BB%, GB%, and FB% are important for understanding the skill-set of the pitcher. For overall production one may be tempted to use a statistic like Fielding Independent Pitching (FIP) because this isolates the skill of the pitcher from his defense. However, we do not want to remove the proficiency of the defense from our analysis, because we care about outcomes that involve the defense. So instead of FIP it is more appropriate to use a statistic like weighted On Base Percentage (wOBA), which describes how often the pitcher gives up each type of hit (single, double, triple, Home Run). Another important characteristic of the pitcher is his handedness relative to that of the batter. It has been shown conclusively that in general hitters fare better against pitchers of the opposite handedness of themselves [19]. This is known as the 'platoon advantage'. The degree of this advantage varies by hitter and pitcher, so it will be necessary to use each individual's platoon advantage as well as the interaction between them. If a lefty-pitcher with a large platoon advantage is pitching against a lefty-hitter who really struggles against lefties, one would expect the overall platoon advantage to be larger than that of either individual.

The Ballpark: One unique aspect of baseball is that it is the one major sport in which there are no regulations about the dimensions of the outfield. This means there are sizable variations in the distance to the fence and just the overall space in the outfield among MLB parks. As one might imagine this makes certain parks more conducive to specific events. To quantify this, past research has developed what have become known as "park factors [20]." At the most fundamental level these numbers represent the frequency of a given event in a specific ballpark relative to league average. So if a specific park has an overall park factor of 1.05, that means that the number of runs scored at that park is 5% higher than league average. It is vital to move beyond overall park factors and look at park factors for all individual events that impact point totals. Each park has a park factor for singles, doubles, triples, home runs, strikeouts, and walks. While it is easier to understand how different parks could have different frequencies of singles or home runs based on dimensions, it is less clear how a park could impact walks or strikeouts. There are two main ways this can occur. One is what is called the hitter's backdrop. This is whatever colors/shapes are in the stands in center field, which is where the hitter is looking as he awaits the pitcher's delivery. The other is the weather that is typical of a given park. The thickness of the air impacts the rotation of a pitch, which in turn impacts how much that pitch will move on its way to the plate [21]. This can influence both strikeouts and walks as the trajectory of pitches is changed.

When it comes to using these park factors to improve predictions it will first be necessary to understand the profile of points scored by an individual player. What we mean by this is that players derive their value from different aspects of the game. Some hit a lot of home runs, some draw a lot of walks, and some steal a lot of bases. We need to model the impact of the park only so far as it affects the style of play of a given player. If there is a player who never hits home runs, playing in a park with a Home Run park factor of 0.80 will not impact his point total. It will also be crucial to model the interaction between park factors and the weather on a particular day.

Weather Conditions: This variable epitomizes the difference between DFS and traditional fantasy sports. Given the unpredictable nature of weather, specifically temperature, wind, and rainfall in this case, it really cannot factor into one's decision making in traditional fantasy sports. But in DFS one can easily obtain very accurate weather data minutes before the game begins. The temperature is important for two reasons. One is that temperature, like altitude, affects air density, which impacts how far a batted ball will travel [22]. Additionally, it has been shown that temperature impacts the level of performance of players in general. Players have shown fatigue at high temperatures, and stiffness and a lack of quickness at cold temperatures [23]. Wind speed and direction are important for obvious reasons. If the wind is strongly blowing towards the outfield, batted balls will carry more leading to more extra base hits and home runs. The opposite is true of wind blowing in from the outfield. One can also see how the impact of certain weather conditions could be amplified or dampened by specific ballpark features.

Pitcher-Catcher Combination: The FanDuel scoring system places a relatively high value on stolen bases, awarding two points for a successful steal and not assessing a penalty for being thrown out stealing. If a player hits a single and then steals second bases they will receive three points. But if a player hits a double they will only receive two points even though in both cases the player ends up on second base. Furthermore a double will likely lead to a run if there are any runners on base while a stolen base does nothing to advance other runners. This represents a potential inefficiency that can be exploited, with a premium being place on faster players who can steal bases.

The main external factor that influences the likelihood of a stolen base is the ability of the pitcher and catcher to both keep runners from trying to steal and throwing them out if they do attempt a steal [24]. The pitcher impacts the probability of a stolen base attempt by delivering the ball to the plate quickly. Additionally, a catcher with the reputation of being able to throw out runners trying to steal can act as a deterrent for potential base stealers. Again this variable will likely be unimportant for the majority of players, but for players who derive substantial value from being able to steal bases this will be important.

Lineup Spot: A baseball lineup consists of nine hitters and is cyclical, meaning that after the ninth hitter bats the first hitter is up next. In general teams tend to put their stronger hitters earlier in the lineup as they will get more plate appearances. Because hitters can get credit for events that involve other players (specifically runs and runs batted in (RBI)), a hitter's spot in the lineup is relevant to predicting DFS points. For example if there was a hitter that never had any runners on base when he came up it would be impossible for him to record any RBIs unless he hit a home run. While this extreme scenario does not occur, it is true that hitters at the bottom of the lineup have fewer opportunities to drive in runs due to the lower quality of the players around them in the lineup. Another result of the quality of hitters around a player is it can impact how a pitcher approaches them. If a pitcher knows there is a weak hitter up next he may 'pitch around' the current hitter. This concept is known as lineup protection and can influence the quality of pitches a batter sees [25].

Proportion of Points from each Event: Baseball players have a wide variety of skill-sets, or 'tools' as they are referred to, and therefore provide value in different ways. To quantify this we have calculated the proportion of a player's FanDuel points that come from each event that a player can receive points for. While this set of variables may not be predictive on its own it will allow us to model the interaction between various game-day conditions and a particular player's skill set.

A common thread through much of this analysis is the importance of the interaction between different skills and external factors. Baseball is a game with few if any dominant strategies. It is like an infinitely more complicated version of 'Rock, Paper, Scissors.' One skill-set might be very beneficial in a certain scenario, but ineffective in another. We have already discussed several examples of this phenomenon, and will definitely need to account for them in whatever model we choose to use. Again, we should stress that we are only able to account for the myriad of potential interactions because we are on a daily timescale, and have very specific and timely information about all the players. Being able to fully leverage these interaction terms will be the key to sound predictions.

2.4 Model Selection and Results

With these features in mind we can now think about what type of model we should use. Before discussing the pros and cons of several approaches, it is useful to consider some general characteristics that are important in this application. As we have continuously emphasized, any model we use must be able to capture the interactions between variables. Some of these interactions are likely nonlinear as well. Additionally, it will necessary to use a relatively high number of features, so we need to be cognizant of how the model handles high-dimensional data. And finally, we are really only concerned with prediction, so model interpretability is not a priority. With this in mind let us first briefly discuss several modelling approaches that we tried and then the model we ultimately used to make our predictions.

2.4.1 Linear Regression

The potential benefits of linear regression are that it is very interpretable and it is parametric meaning it can easily handle a high number of features. Of course this simplicity can also be a drawback as its linear structure may not be able to fully capture the relationship between the response and explanatory variables. We built a linear model using various subsets of the variables explained above, including several interaction and nonlinear terms to try to provide a little more flexibility. The results were not very promising with an R^2 in the range of 0.05-0.07 depending on the exact model and an RMSE on the testing set of about 3.25. This lack of predictive power highlights both the amount of randomness in player point totals as well as the fact that some of the relationships are likely nonlinear. For example, one would expect the relationship between temperature and performance to be nonlinear with no significant effect within the range of typical temperatures but potentially a significant and opposite effect at the two extremes.

2.4.2 Generalized Additive Model

After quickly realizing that a linear method was unlikely to work we turned towards more flexible approaches. Generalized Additive models (GAM) assume that the response (or a transformed response if the identity link is not used) is equivalent to a sum of functions of the explanatory variables [26]. These functions are usually fit non-parametrically but structure can be imposed on some or all of them if desired.

$$E(Y|X_1, X_2, ..., X_p) = s_0 + \sum_{i=1}^{p} s_i(X_i) \tag{2.1}$$

The primary benefit of this approach is that if the functions are fit non-parametrically it should capture nonlinear relationships. While the GAM performed better than the simple OLS regression with a RMSE of 3.06 it did not prove as predictive as our ultimate approach.

2.4.3 Kernel Regression

Kernel Regression is a non-parametric local regression technique that weights each observed response value by how close it is to the observation one is currently trying to predict a response for [16]. It takes the form:

$$Y = f_{b,K}(x) = \frac{\sum_{i=1}^{n} y_i K(\frac{x-x_i}{b})}{\sum_{i=1}^{n} K(\frac{x-x_i}{b})} \tag{2.2}$$

where x is the vector of predictors for the observation one is trying to predict a response for, K is the kernel function, and b is the bandwidth. While the choice of kernel function is not too consequential, the choice of bandwidth is extremely delicate and needs to be determined by cross-validation. A smaller bandwidth will increase the locality of the regression. The largest drawback to using this method is that it does not handle high-dimensional data well. As one expands the number of dimensions, the number of observations that will be considered close to the new observation will decrease. This is known as the 'curse of dimensionality' [27]. Keeping this in mind we attempted to use kernel regression with the subset of variables that were the most influential predictors. The resulting RMSE on the testing set was 3.13, still not an improvement over the best performing method.

2.4.4 Generalized Boosted Models

A Generalized Boosted Model (GBM) is an iterative non-parametric regression method that uses the techniques of gradient descent and subsampling to find a function of the explanatory variables that minimizes a given loss function [28]. While it is not a very interpretable model the subsampling and ability to shrink the gradient step size allow it to deal with noisy data well. Additionally the direction of descent is determined by using a regression tree which explicitly models the interaction between explanatory variables. Finally, even though it is non-parametric the nature of this model means that it does not suffer from the 'curse of dimensionality' the same way Kernel Regression does. The details of the full algorithm as well as how it is implemented in R can be found in [28].

We fit a GBM model to our training data (all games played before September 1st across all years) using the 'gbm' package in R [29]. There are several parameters that the user must choose. One is the loss function, which we are trying to minimize. Given the continuous nature of our

response variable we had to choose between the L2 and L1 loss functions. One feature of the L1 loss function is that it is more robust to extreme points. However in our application these extreme points (when hitters have very good games) are very important and we do not want the model to neglect them. For this reason we chose the L2 loss function. We also have to choose the interaction depth of the regression trees that are used to determine the direction of steepest descent. Given our desire to allow for complex interactions between variables we set the interaction depth to 5, which is the upper limit of the range suggested by the package's author [28]. There is also the matter of setting the number of iterations to perform and the shrinkage parameter, which modifies the step size of the gradient descent. As one can understand the optimal value of these parameters depend on each other. If a very small step size is used it will be necessary to run through more iterations. We chose to set the shrinkage parameter at a small value of 0.005 and then used the *gbm.perf()* function to determine the optimal number of trees, which turned out to be slightly over 3000. Lastly this method uses a subsample of the training data to fit the regression tree at each iteration. We set the subsampling proportion at the customary value of 0.5. After deciding on

Parameter	Value
Loss Function	Gaussian
Interaction Depth (K)	5
Number of Trees (T)	3097
Shrinkage (λ)	0.005
Subsampling Proportion (p)	0.5

Table 2.2: Parameter Values for the GBM

Variable	Relative Influence
PPG	49.35
Pitcher wOBA by Handedness	25.66
Player wOBA by Handedness	18.72
Temperature	1.33
Home Run Park Factor	1.15
Lineup Spot	0.70
Double Park Factor	0.65
Successful SB%	0.62
Single Park Factor	0.62
Proportion of Points from SB	0.38
Attempted SB%	0.33
Platoon Advantage	0.17
Pitcher Walk Rate	0.16
Proportion of Points from HR	0.06
Wind Speed	0.04
Pitcher K% by Handedness	0.03
Player K% by Handedness	0.02

Table 2.3: Relative Influence of Explanatory Variables in Descending Order

these parameter values we used our GBM to predict point values for our testing data set. While the improvement over our other methods is not the most substantial it was clearly the best performer with an RMSE of 3.014. From our results we can also see which variables had the largest impact on

point totals through a metric called relative influence [28]. Before analyzing these results, some of these variables deserve more explanation. Whenever a variable is 'by handedness' it means the value of that variable solely against opponents of the relevant handedness. So if we were predicting the point total for a right-handed hitter, the pitcher wOBA by handedness would be the wOBA of the opposing pitcher against right-handed hitters. The successful SB% variable is the average of stolen base success rates against the pitcher and catcher respectively weighted by how many opportunities they have had. So if over all the games the current pitcher has thrown this year, 13/20 stolen base attempts have been successful and in all the games that the current catcher has caught this year 8/11 stolen base attempts have been successful, the raw value of this variable would be 0.677. The Attempted SB% variable is defined analogously, expect instead of success rate it is just the number of times the runner attempted to steal per steal opportunity (defined as any situation with a runner on first and no runner on second).

Overall the results are quite intuitive. Unsurprisingly the most influential predictor of how many points a player will score is his average PPG. Another sensible finding was that the overall skill of the pitcher (and his defense since we are using wOBA) is the second most important variable. One can see that overall batter skill and overall pitcher skill gets us most of the way there in terms of prediction. However it is the factors external to these two central variables that really capture the full context of a game. The ballpark is clearly important, as is where the hitter is placed in the lineup. Probably the most surprising result was the relatively high impact of temperature. While there are strong theoretical grounds for it having an impact, the magnitude is noteworthy. We should also point out that many of these explanatory variables are related to each other, meaning the effects of some have been engulfed by others. An example of this is the perceived unimportance of the 'platoon advantage' with a relative influence of only 0.17. If we had not adjusted the wOBA variables to be by handedness the platoon advantage would be much more important. On the following two pages we show several figures that highlight the flexibility of the GBM. All of these figures show the impact of the included variables on player point totals holding all other variables constant. Furthermore the values shown in the sidebar represent the predicted point total as a function of just the given variables. We first see that holding all else equal hitting earlier in the lineup results in higher point totals. We also see that the ability of a pitcher-catcher combination to prevent stolen bases is not relevant when a given hitter doesn't accumulate points through stolen bases. There is also an intuitive relationship between the Home Run park factor and the temperature, with high temperatures in home run conducive parks leading to higher point totals. Finally we can see the nonlinear relationship between a player's PPG and the opposing pitcher's wOBA.

It might seem underwhelming that our best model has a RMSE that is higher than the average point total. This again comes back to the randomness inherent in the game as well as the skewed distribution of point totals. A sensible model will simply never predict a player will have one of these extreme games. This leads to a very skewed residual distribution where a large portion of the overall error come from relatively few points. For example if we looked at the Root *Median* Squared Error we get a value of 1.982 (this improvement is not specific to the GBM). But the accuracy of our predictions only matter in so far as it results in the optimization problem choosing the best players on a given day. All we can hope for is that the model nudges the optimization in the right direction, leading it to choose players who will go on to have these great games. With this in mind we should not judge the efficacy of the model before seeing the results of the optimization problem.

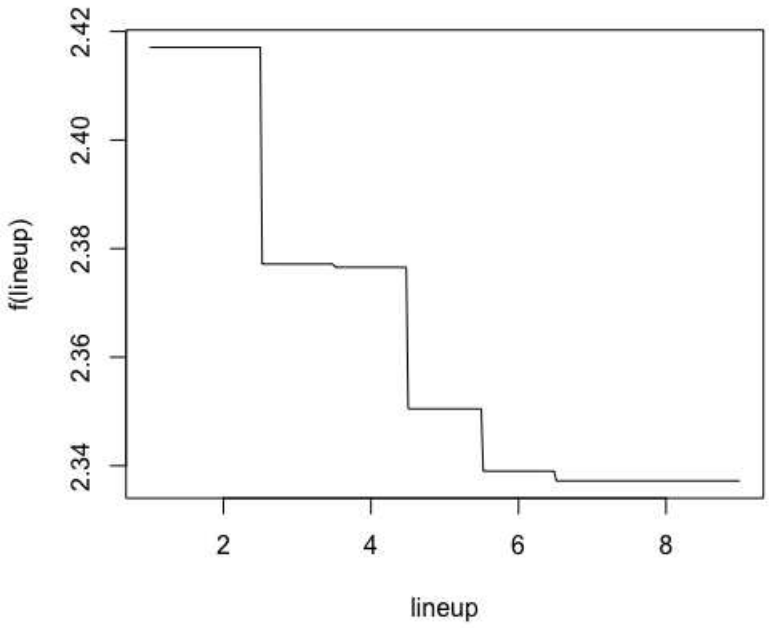

Figure 2.3: The Impact of Lineup Spot on Point Total

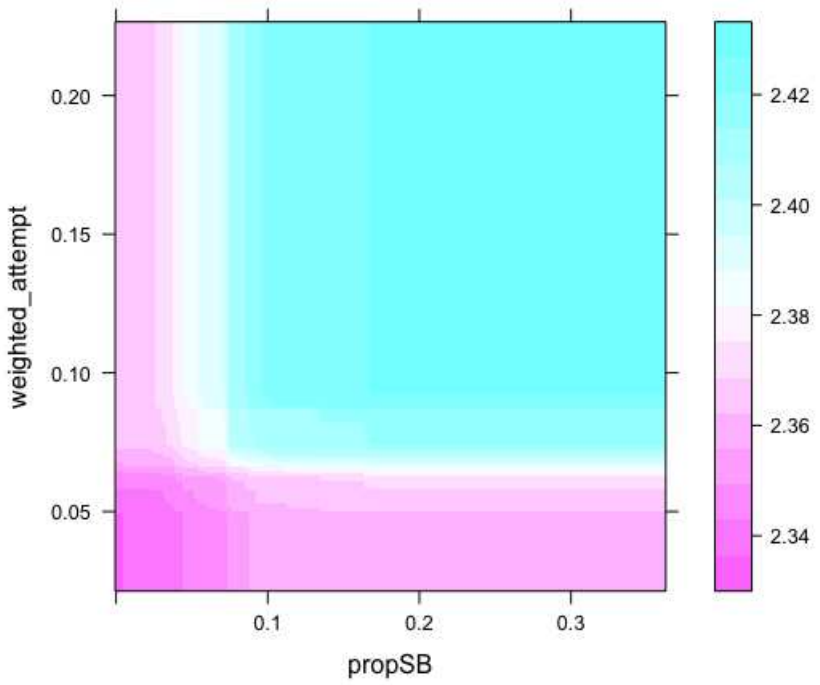

Figure 2.4: The Impact of the Proportion of Points obtained from Stolen Bases and the Ability of the Pitch-Catcher combination to prevent Stolen Base Attempts

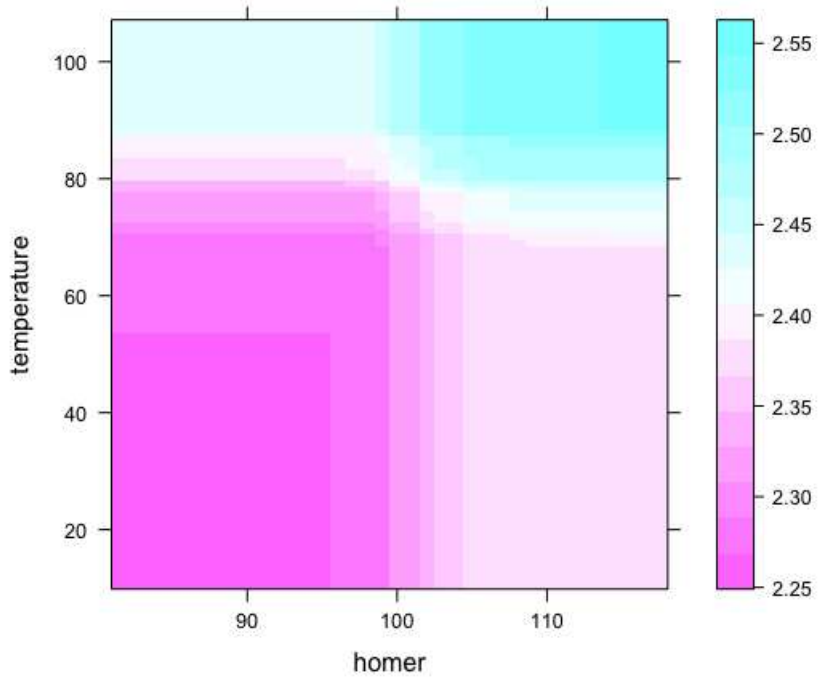

Figure 2.5: The Impact of the Home Run Park Factor and the Temperature

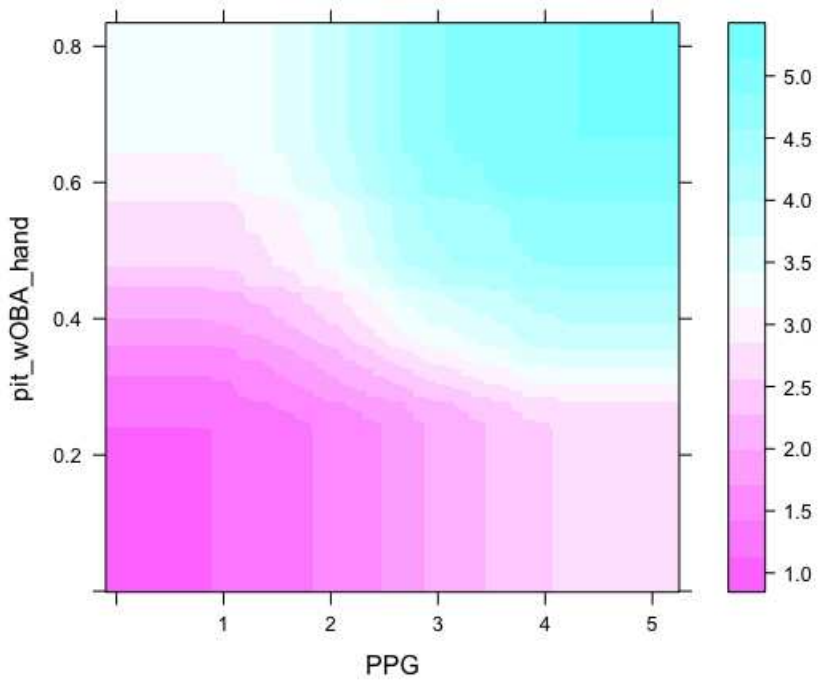

Figure 2.6: The Impact of Average PPG and Pitcher Skill against the Relevant Handedness

Chapter 3

Predicting Pitcher Performance

3.1 Exploratory Analysis

We now turn our attention to performing a similar analysis for pitchers. There will be many similarities to the previous chapter on hitters in terms of potential explanatory variables and modelling but also some important differences. Each FanDuel lineup contains one hitter from each defensive position (eight hitters) and one pitcher for a total of nine players. One might look at the composition of this lineup and assume a pitcher is not very important relative to all the hitters. However the FanDuel scoring system (see Table 3.1) is designed in such a way that choosing the right pitcher is absolutely crucial to winning. The average points scored for a pitcher in FanDuel is 9.31, while for a hitter it is 2.37. So a pitcher will accumulate the same amount of points as four hitters combined on average. For example if a pitcher throws six innings, strikes out four hitters while allowing three earned runs, and records a Win he will receive 11 points. This is an ordinary game for a pitcher, but a hitter would have to have a great game to score 11 points. A starting pitcher records a Win if he pitches at least five innings, his team is in the lead when the pitcher leaves the game (or if they take the lead before another pitcher enters the game), his team never relinquishes the lead, and his team eventually wins the game [30].

Event	Points
Strikeout	1
Inning Pitched	1
Earned Run	-1
Win	4

Table 3.1: FanDuel Scoring System for pitchers

As with hitters it is useful to look at the distribution of pitcher point totals. Recall that this distribution was very right-skewed for hitters. However this could not be further from the case for pitchers. In Figure 3.1 we see that this distribution appears to be normally distributed, which is confirmed by examining a QQ-plot. This is a result of the increased number of opportunities pitchers receive during a game as well as the way the scoring system is constructed. While a hitter will usually only get 3-5 plate appearances in a game, a starting pitcher will tend to face upwards of 20 hitters. Increasing the number of opportunities will have a two-fold effect that decreases variance.

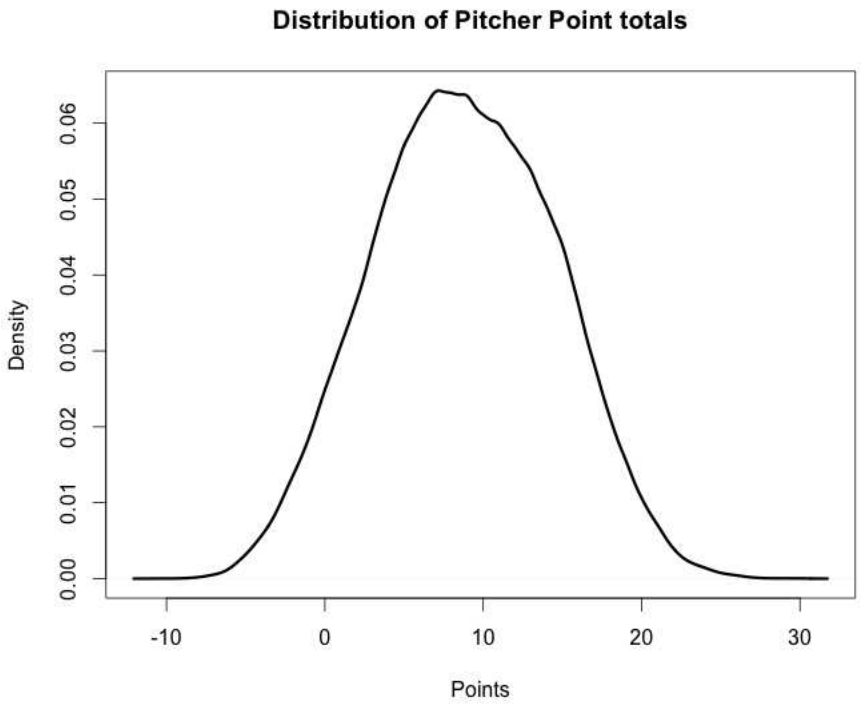

Figure 3.1: Distribution of Pitcher Point Totals

One is simply that it will be more likely for randomness to be washed out. The weakly hit ball that barely got through the infield is more likely to be compensated for by the hard line drive right at a fielder if the number of plate appearances is higher. Also it means that the expected value of all of the counting statistics (strikeouts, innings pitched, earned runs) will be greater than one. In fact the sample averages for these per game were 4.43, 6.01, and 2.78 for strikeouts, innings pitched, and earned runs respectively. In this way the outcomes for pitchers are far less binary in nature. Recall the comment in chapter 2 about the discrete nature of home runs leading to higher variance due to the fact that a hitter may only hit one every 15 plate appearances (but only has 3-5 chances). In that case the expected value is close to zero leading to a higher variance. This is compounded by the scoring system rewarding these rare offensive events with multiple points. While for hitters there were multiple events that contributed more than one point, and these events could occur more than once, the only event for which this is the case for pitchers is getting a Win. A pitcher either records one win or no wins, so it is impossible for him to accumulate a lot of points through one event. Also the negative event (allowing an earned run) is weighted equally to the positive events. This lowers the 'floor' of possible point totals leading to more symmetry. For hitters an out was only -0.25 points.

We can also look at the relationship between a pitcher's average PPG and the standard deviation of their PPG. For hitters this was a very strong, positive relationship due to the skewed point distribution. Given the symmetry of the distribution for pitchers we should expect that this relationship is far less significant. Figure 3.2 shows this relationship.

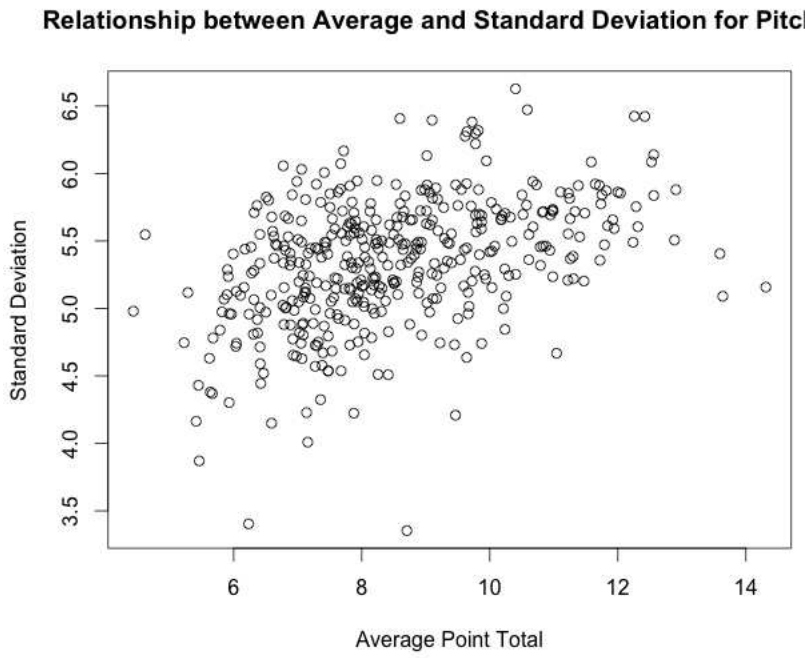

Figure 3.2: Relationship between Average PPG and the Standard Deviation of PPG for pitchers

We immediately notice that there is almost no relationship between the average and standard deviation with a regression slope of 0.134 and an R^2 of 0.083. What this means is that some pitchers dominate others with respect to the risk-return trade off as they have a higher average and a lower standard deviation. This is not to say that some hitters aren't strictly better than others, just that over the course of one game it is hard to cut through the noise to see this. But the increased number of opportunities for pitchers allow the truly better ones to separate themselves. Another relevant observation when it comes to thinking about prediction is the relative lack of volatility in point totals. The standard deviation is 5.79 compared to an average point total of 9.31. So the standard deviation is 62.2% of the mean for pitchers while it was 133% for hitters. This decreased volatility should make prediction easier as we turn towards building our model.

3.2 Potential Explanatory Variables

Some of the variables we use for pitchers are equivalent to those used for hitters. These are average PPG, park factors, temperature, and wind speed. We will not discuss them in detail again here, but note that the effect will be reversed for the latter three as we are performing this analysis from the pitcher's perspective (i.e a high park factor will lead to lower points). Another difference with pitchers is that they have to face nine different opponents rather than just one. So all of the variables we use that concern opposing hitter quality have been averaged over the entire opposing lineup. This includes the average strikeout percentage of the lineup against the handedness of the pitcher in question as well as the average wOBA. We have also included the opposing team's Runs per Game average from that season. To calculate a pitcher's expected strikeout rate against a lineup we

took an average of the pitcher's strikeout rate against left-handed hitters and right-handed hitters weighted by the number of hitters of each handedness in the opposing lineup. We did the same for expected wOBA against. Now we can discuss the several new variables that we did not use for hitters.

Pitches per Plate Appearance: When deciding what variables to use it is necessary to keep in mind how a pitcher accumulates points. This variable seems like an important one when it comes to how many innings a pitcher will throw. Starting pitchers have been throwing fewer and fewer innings as teams have become more sensitive to the injury risks of overworking a pitcher [31]. The primary statistic used to decide when to take a pitcher out of a game is how many pitches he has thrown. It is rare for a pitcher to throw more than 100 pitches in today's game. So if a given pitcher tends to have longer at bats (most likely due to poor control or the lack of a viable strikeout pitch) he will not be able to work as deep into games. On the other side, some teams have a more patient approach than others at the plate, thus making the pitcher throw more pitches. So we will include variables for the average pitches per plate appearance for the pitcher and also for the opposing team.

Designated Hitter: In the American League instead of the pitcher batting, teams are allowed to have a player be the Designated hitter. This hitter does not play in the field, and just hits in place of the pitcher. In the National League the pitcher hits for himself and there is no Designated Hitter. In general pitchers are well below-average hitters as they have devoted most of their development to becoming better pitchers and do not get the opportunity to hit often. There are two conflicting effects of the Designated Hitter on a pitcher's point total. In the American League pitchers face a tougher lineup as a below average hitter is being replaced by typically a very strong hitter. This will lower the expected number of strikeouts and raise the expected number of earned runs allowed. However in the National League where pitchers have to hit, the manager will eventually replace the pitcher with a pinch hitter, thus removing the pitcher from the game. This will have a negative impact on the pitcher's innings pitched. We will see which of these effects outweighs the other after we build the model.

Bullpen wOBA against: As was mentioned above it is increasingly rare to see a starting pitch work deep into games. This has led to the bullpen, which is the term for the group of relief pitchers that come in after the starter, becoming more important for teams. The quality of the bullpen will have two competing effects on a starting pitcher's point total. If the bullpen is effective it will be more likely to preserve a lead, increasing the probability of the starter winning the game. However if the manager trusts his bullpen he will be more willing to pull the starting pitcher early if there are signs of trouble. This would decrease the starter's innings pitched. Again, we will see which of these effects wins out as we turn towards developing the model.

3.3 Model Selection and Results

3.3.1 Predicting each Event Separately

One significant difference between the scoring system for hitters and the scoring system for pitchers is the number of events they can get points for. For hitters it would be extremely unlikely to

project that the expected value for any specific event is over one. This is just due to the high number of possible outcomes and the small number of opportunities. But for pitchers there are only four possible events, with the three counting statistics of strikeouts, earned runs, and innings pitched having expected values significantly higher than one. The nature of these statistics makes it more reasonable to try to predict them on their own, and then calculate expected points from the individual predictions. Although, predicting whether or not a pitcher will win a game is quite different. While the other three events directly occur during the game, a win is just a way to describe how the game went. Quantitatively we can think of the probability of recording a win as a function of the other three events. First we can predict strikeouts, innings pitched, and earned runs and then use these predictions to predict whether or not the pitcher records a win.

Another feature of the pitcher scoring system is that the three events are very related to each other. For example a strikeout is also 1/3 of an inning pitched. And the fewer runs a pitcher is giving up the more innings he will be able to throw. We can see the magnitudes of these correlations in the correlation matrix below.

$$\begin{array}{c} \begin{matrix} K & IP & ER & W \end{matrix} \\ \begin{pmatrix} 1 & 0.43 & -0.30 & 0.23 \\ 0.43 & 1 & -0.51 & 0.44 \\ -0.30 & -0.51 & 1 & -0.49 \\ 0.23 & 0.44 & -0.49 & 1 \end{pmatrix} \end{array} \begin{matrix} K \\ IP \\ ER \\ W \end{matrix}$$

Due to the high correlation between these variables it would be beneficial to use a prediction method that can take this into account and predict all three response variables at the same time. After reviewing the relevant literature this led us to a technique known as Curds and Whey Regression.

The Curds and Whey algorithm was developed by Brieman and Friedman and is an extension of multiple linear regression that can predict multiple response variables from the same set of explanatory variables [32]. It uses the canonical correlation between the matrix of the explanatory variables and response variables to transform the response variables to canonical coordinates. Then it regresses each transformed response variable on the explanatory variables separately. Finally it applies an element of shrinkage to these transformed responses and then transforms them back to the original coordinates. The use of the canonical correlation means the relationship between the response variables will factor into the prediction. However after applying Curds and Whey to our data it was again clear that the assumption of linearity was too restrictive. Unfortunately an extension of Curds and Whey to nonlinear or non-parametric regression has not yet been developed.

The primary reason we wanted to predict these three events separately was that it would greatly help us predict whether or not a pitcher would record a win or not. Seeing that a Win is worth four points, our predicted probability of a win is very influential when deciding which pitcher to choose. It turns out that if we had been able to develop reasonable predictions for the three other events we could have been quite successful. Because a win depends not only on how many runs a pitcher gives up, but also how many runs his team scores we built a logistic regression model that used the actual earned runs allowed and innings pitched from both starting pitchers in a game as explanatory variables. We also included the strength of each bullpen as it is up to the bullpen to preserve the lead. This gave us six explanatory variables in total with the response being whether

or not a win was recorded. This model was able to predict the correct outcome 89% of the time. We should again stress that this was using the actual observed values for the explanatory variables instead of our predicted values. The success of this model does suggest that if one were able to develop a method for effectively predicting strikeouts, innings pitched, and earned runs allowed separately (ideally taking into account the correlations between them), it would ultimately lead to a better prediction than trying to predict points directly. However we found that the benefit of being able to predict wins sequentially is outweighed by the inaccuracy of predicting the three counting statistics separately with the currently available methodology. So we again turn to Generalized Boosted Models to predict pitcher points directly.

3.3.2 Generalized Boosted Models

We have already discussed GBMs in chapter 2 and the full algorithm can be found in [28]. While we tried other methods for predicting pitcher point totals the GBM again performed the best on the test data set with an RMSE of 5.37. All parameters were identical as the GBM for hitters except the optimal number of trees was slightly lower. On a relative basis this is an improvement over our predictions for hitters, as the average point total for a pitcher is almost twice as high as the RMSE. This is due to the decreased variability of pitcher performance on a daily timescale relative to hitters. We can also look at the relative influence of the explanatory variables.

Variable	Relative Influence
PPG	76.05
Opposing lineup wOBA	9.52
Opposing lineup K%	5.35
Pitcher wOBA by handedness	3.37
Opposing Team Runs per Game	1.25
Temperature	1.03
Double Park Factor	1.02
Pitcher K% by handedness	0.66
Home Run Park Factor	0.53
Lineup Fly Ball%	0.39
Lineup Pitches per Plate Appearance	0.23
Pitcher's Team Runs per Game	0.20
Designated Hitter	0.09
Proportion of Points from HR	0.08
Bullpen wOBA	0.06
Pitcher BB%	0.03

Table 3.2: Relative Influence of Explanatory Variables in Descending Order

For pitcher's their average PPG is even more influential than for hitters. This is because pitchers are less volatile and less impacted by the external game conditions. We also see a nice parallel between pitchers and hitters in that the skill of the opposition is the second most influential variable. Note that K% is more important for pitchers because pitchers directly receive points for strikeouts while for hitters all outs are treated equally. Again we also see the relative importance of temperature, although for pitchers a higher temperature leads to lower points. Finally we see that the Designated Hitter and Bullpen variables have very low influence. This could be because the competing effects we mentioned earlier more or less cancel each other out, or these variables may simply be less important

than we thought. Again many of these explanatory variables are related to each other. For example it may seem surprising that the opposing lineup's K% is found to be far more influential than the pitcher's K%. This is just becasue most of the information contained in the pitcher's K% is already taken into account through the average PPG variable.

We again show a few figures that highlight the findings of the GBM. First we can see the negative impact of temperature on pitcher point totals. Notice that there is only a real impact at above average temperatures. This is most likely a result of both the effect on air density and the fact that pitchers will fatigue quicker in hot weather. On the next page we see the interaction between park factors and the strength of the opposing offense. We also see the impact of the two most influential variables, average PPG and the opposing lineup's wOBA, on pitcher point totals. Now that we have our predictions for both pitchers and hitters we can finally construct our optimization problem.

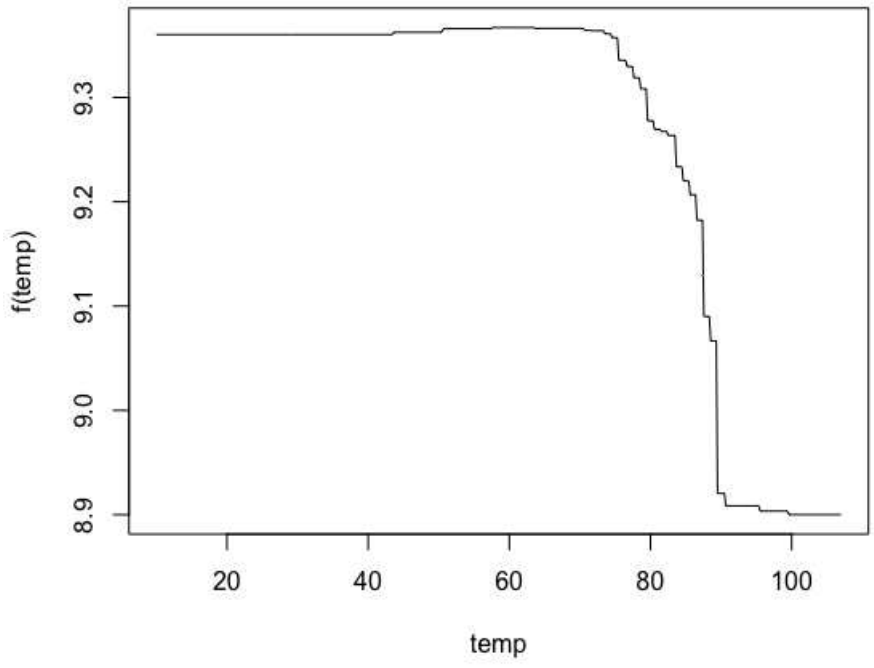

Figure 3.3: The Impact of Temperature on Point Total

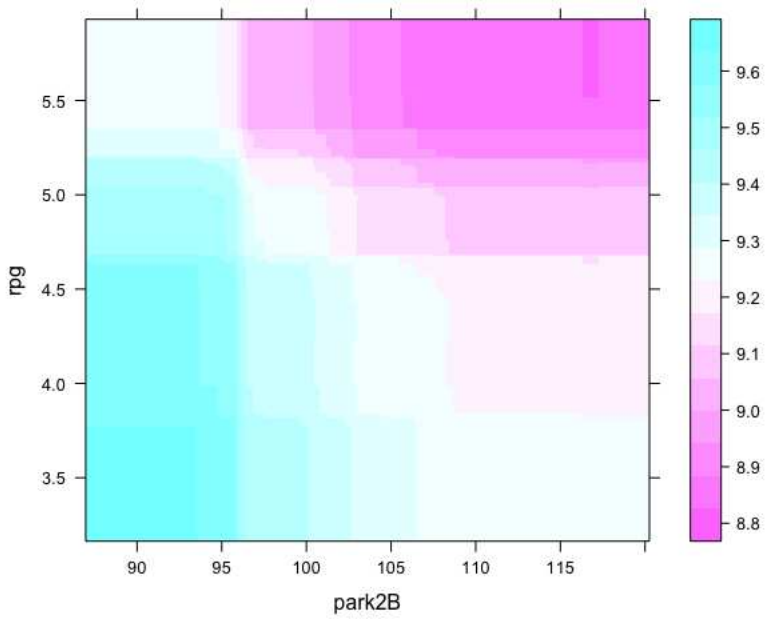

Figure 3.4: The Impact of the opposing team's Runs per Game and the ballpark's park factor for doubles

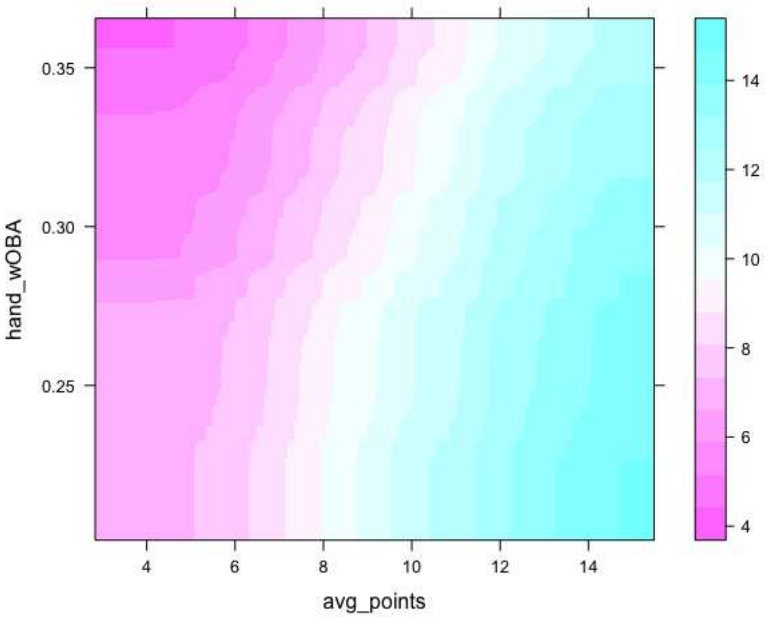

Figure 3.5: The Impact of the opposing team's wOBA by pitcher handedness and the pitcher's average PPG

Chapter 4

The Optimization Framework

To determine which players to choose for our lineup on a given day we construct an Integer Program where the decision variable is a vector of all the players playing that day. Let us begin constructing this problem by first examining the three constraints that FanDuel imposes on lineups.

4.1 Daily Fantasy Baseball Lineup Constraints

4.1.1 The Salary Cap

By far the most important constraint is the salary cap. Every day FanDuel will use a proprietary algorithm to determine the DFS salary for each player. Every participant is given a virtual salary cap of $35,000 to spend. One can of course choose to spend less than $35,000 but cannot exceed it. As we mentioned in the Introduction we do not have this information for our data set, as it simply does not exist. To deal with this we took advantage of the fact that we had data from the entire 2015 MLB season containing FanDuel salaries [33]. We built a model to predict a player's FanDuel salary using the entire 2015 season as our training set, and then used the model to predict what player salaries would have been for our data set.

The full details of the models used to predict salaries for hitters and pitchers separately can be found in the appendix. We used a linear regression model for both and the R^2 values were 0.56 and 0.71 for hitters and pitchers respectively. While it is clear from the unexplained variance in model that FanDuel takes external game-day conditions into account, they do not vary salaries too much for a given player. The average standard deviation for a hitter's salary as a proportion of his mean salary was 0.088. For pitchers this number was 0.072. We should also note that FanDuel rounds all salaries to the nearest $100 and never assigns a hitter a salary less than $2200. After we made predictions for our data set we made these modifications as necessary to stay consistent with FanDuel's system. We of course understand that using these imputed salaries is not ideal but we feel that the models explain a sufficient amount of the variance in salaries to justify it. Also when we evaluate our optimization results against several benchmarks we will use these same salaries to constrain those benchmarks so our lack of certainty about the salaries should not bias our results in any particular direction.

4.1.2 The Position Constraint

FanDuel imposes the constraint that every lineup must consist of one player from each defensive position. So each lineup must have one Catcher (C), one First Baseman (1B), one Second baseman (2B), one Thrid Baseman (3B), one Shortstop (SS), three Outfielders (OF), and one Pitcher (P). Note that while one is required to have three Outfielders FanDuel does not require that a lineup has one left fielder, one center fielder, and one right fielder. One reason for this constraint is to more closely simulate the experience of choosing a real team, where a manager must also consider what defensive position each player can play. Another reason they impose this constraint is because it has been shown that offensive performance varies by defensive position [34]. This is due to the fact that some positions are harder to play, so at these positions there is premium put on defensive value at the expense of offensive production. The current consensus on the relative difficulty of defensive positions is as follows: C > SS > 2B > 3B > CF > RF > LF > 1B [34]. One can see this reflected in the data by looking at the average PPG across the different defensive positions. The results are completely consistent with the defensive spectrum.

Position	C	SS	2B	3B	CF	RF	LF	1B
PPG	1.95	2.16	2.27	2.36	2.54	2.56	2.58	2.62

Table 4.1: Average PPG by Defensive Positon

4.1.3 The Team Constraint

Another constraint that FanDuel imposes is that a lineup cannot have more than four players on the same team. This includes hitters and pitchers, so one cannot have four players from the same team and then that team's pitcher as well. They impose this constraint to prevent a strategy known as 'stacking.' This refers to a participant stacking their lineup with all the starting players from one team. We will see why this can be an effective strategy in the next chapter but for now we will just be sure to include this constraint in our optimization problem.

4.2 Setting up the Optimization Problem

With these constraints in mind we can now formally construct our optimization problem. Our objective function will simply be to maximize expected points, and our decision variable will be an Nx1 vector x where $x_i = 1$ if player i is in our optimal lineup.

$$\begin{aligned} \max_{x} \quad & C^T x \\ \text{s.t.} \quad & S^T x \leq 35,000 \\ & Ax = b \\ & Tx \leq 4 \\ & x_j = \ 0 \text{ or } 1 \ \forall j \in N \end{aligned} \tag{4.1}$$

Let us formally define all of the matrices and vectors in this problem, starting with the constraints. The vector S is an $Nx1$ vector, where N is the number of players playing on that day, and element S_i contains the FanDuel salary for player i. $S^T x$ will produce a scalar which must be less than or equal to the salary cap. A is a 7xN matrix where element A_{ij} is equal to one if player j plays position i and zero otherwise. Note this matrix only has seven rows instead of nine because all three outfielder positions are considered the same. The vector b is a 7x1 vector whose first six elements are one and whose seventh element is equal to three (for the three outfield lineup spots). Finally, T is a 30xN matrix where element T_{ij} is equal to one if player j is on team i and zero otherwise. The result is a 30x1 vector where each element must be less than or equal to four. Finally there is the constraint that all elements of the decision vector must be zero (if not in the lineup) or one (if the player is in the lineup). *The vector C in the objective function is the result of all the work in the previous two chapters.* This Nx1 vector contains our predictions for every player on a given day. Our objective is to maximize the expected points scored by our lineup. We note that although C is random in reality, as we do not know how many points each player will score while choosing our lineup, we treat it as deterministic. We can also easily substitute different C vectors that were calculated by different methods. This allows us to evaluate our predictions relative to several other approaches for determining C.

4.3 Optimization Results

As we mentioned in the previous chapters, the most precise metric by which to evaluate our prediction model is to see how it performs when used to actually choose a lineup. Consequently we do not care about the value of the objective function but only the optimal solution x^*. Once we use our predicted point totals to allow the optimization problem to choose a lineup we are only concerned with how the players chosen *actually* performed, not how we expected them to perform. This is the true value of any lineup. In order to evaluate our predictions in this context it will be necessary to compare our approach to a few benchmarks. We will calculate the vector C four different ways and see how each performs over the same sample of days. First we will use what we call the 'Naive' approach, assigning each element C_i to the average PPG of player i. This approach ignores all external game-day conditions. We will also use what we call the 'Omniscient' approach, assigning each element C_i to the actual points scored by player i on that day. This approach will pick the truly optimal lineup every time as it can see into the future. Lastly we wanted to investigate how much a slight improvement in our predictions would help. To do this we artificially improved all predicted point totals by moving them 1.5% closer to the actual values. Note that this means that the larger the discrepancy between the prediction and actual value the bigger an improvement this will cause. We will refer to this as the 'Improved' approach.

We solved this optimization problem for each day that was in the testing set of our data. This includes every day in which there was more than one game in the month of September from 2003-2013 for a total of 324 days. For each day we solved the optimization using the four different C vectors described above. Seeing as this is an integer program, and therefore non-convex, we used the Gurbobi Optimization software package in R to solve the problem [35]. The table on the following page summarizes the performance of each approach.

Metric	Naive	Our Model	Improved	Omniscient
Average Points Scored	37.07	42.60	51.23	101.39
Standard Deviation	11.15	13.55	13.88	11.11
Days it Outperformed the Naive Approach (%)	N/A	65.12	83.6	100
Days the Lineup Salary was over $34,700 (%)	100	94.44	95.68	28.40

Table 4.2: Lineup Performance for Different Prediction Approaches

The most important result in this table, and perhaps in this paper, is that our approach beats the naive approach by 5.53 points on average (the distribution of the difference is on the next page), and outperforms it 65% of the time. Both of these statistics have p-values essentially equal to zero for the null hypothesis of equal performance (n=324). *This improved performance is solely a result of taking into account external game-day factors that the naive approach does not consider.* Furthermore, while five points may not seem like the most significant improvement, both these approaches are in the range of point totals in which each marginal point greatly increases the probability of finishing in the money, so when translated to win probability this difference is significant. Another fascinating result is how much our results would improve if we were able to increase the accuracy of our predictions by just 1.5%. We knew that the 'Improved' approach would dominate our approach by construction, but the sensitivity of performance to better predictions is noteworthy. The Omniscient approach gives us the point total of the truly optimal lineup every day, with an average score of over 100 points. This means that each player in this lineup is scoring over 10 points on average. One instructive finding from the omniscient approach was that the salary of the best lineup was far less likely to cost close to the full salary cap than the lineups selected by the other approaches. This is further evidence of how unpredictable baseball is on a daily basis and that even the DFS providers cannot predict how players will perform (or else they would set salaries in such a way that the optimal lineup cost close to the full salary cap).

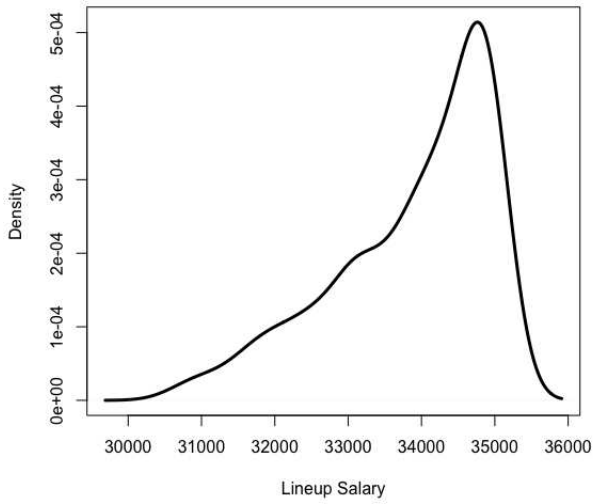

Figure 4.1: The Distribution of Lineup Salaries for the Truly Optimal Lineup

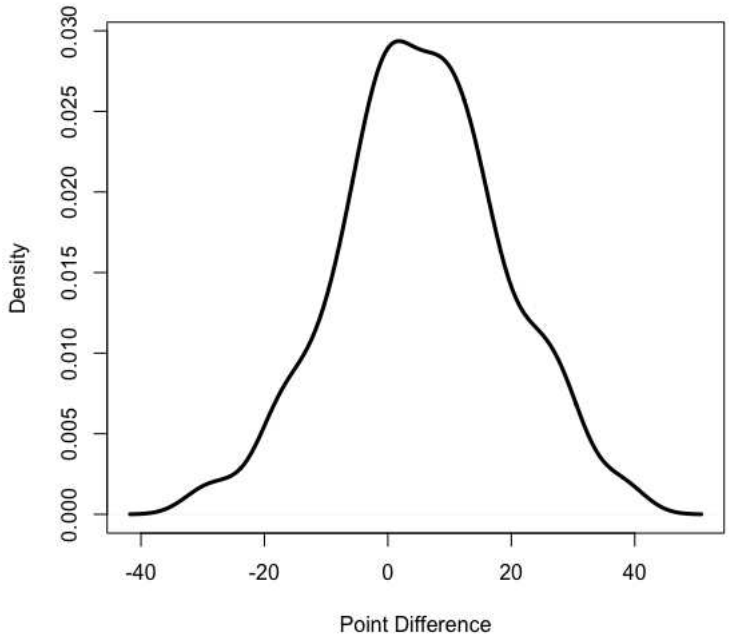

Figure 4.2: The Distribution of the Difference in Lineup Performance between our prediction approach and the naive approach. A positive value indicates that our approach performed better that day

As we alluded to above what we really care about is our probability of winning given how many points our lineup scores. However this relationship is not perfectly understood due to a lack of data. We can of course assume that the probability of winning increases monotonically with points scored. It is also reasonable to assume that this relationship is nonlinear with an inflection point around the most commonly obtained point values. One study done by one of the most popular DFS strategy websites found that in Head-to-Head contests with an entry fee between \$1-\$5 the 50th and 70th percentiles of point totals were between 30-32 and 36-38 respectively [36]. If this study is just in the right ballpark (and there is no reason to believe it is not with a sample size of about 1000) this means that our approach will win at least 70% of the time. Even if we adjust our average point total to account for the decreased offensive output in baseball (as this study was done in 2015) our approach would still produce roughly 40 points on average. However this is just for low stakes H2H contests. One would assume that as the stakes rise the quality of opponent increases leading to a higher necessary point total. Unfortunately there is not enough data from these high-stakes contests to draw any conclusions. Additionally there are other contest types that will require higher point totals to win. The next chapter defines all of these different contest types in detail and how we must adjust our strategy for each one. Nevertheless the fact that our approach produces lineups that routinely score above the 50th percentile in Head-to-Head contests strongly suggests that it can be implemented with profitable results.

Chapter 5

Optimization by Contest Type

FanDuel offers many different contest types, each with their own payout structure. In this chapter we will focus on the three most common contest types and how their different payout structures should alter our strategy when constructing a lineup.

5.1 Different Contest Types

The simplest contest type is Head-to-Head. Two participants each pick a lineup, and the lineup with the higher score wins. If for example each participant entered $5, the winner would receive $9 (as the website takes a constant 10% off the top). This contest type lends itself to the optimization problem in chapter 4, as the sensible strategy is to just try to maximize expected points. The symmetric and binary nature of the payout structure means considering the variance of a lineup's point total is less important. The next contest type is called a 50/50 contest. In this contest the top half of lineups all receive the same payout and the bottom half all receive no payout. So if there were a 50/50 contest with 20 participants and a $5 entry fee the owners of the 10 highest scoring lineups would all receive $9 and the owners of the bottom 10 lineups would lose. This flat payout structure discourages participants from trying to choose lineups with high upside, as finishing 10th provides the same payout as finishing 1st. So here it may be beneficial to minimize the variance of the lineup's point total to some extent. Finally we have what DFS sites refer to as tournaments. These are contests with a high number of entries (upwards of 10,000), a low entry fee, and a payout structure that is heavily skewed towards the top lineups. For example, one can enter a tournament with 20,000 entries with an entry fee of $1 and if their lineup has the highest score they will receive $2000. There are payouts for the top 4,000 entries but the dollar amount is very skewed to the top 10 or so finishers. These contests can be very appealing due to the extremely high payouts relative to the entry fee. In tournaments participants need to have lineups with a high upside if they want any chance at finishing in the money. So here it makes sense to seek a lineup with high variance, perhaps even at the expense of a lower expected value. Clearly the variance of a lineup is relevant, but before we can include it in our optimization we first have to quantify it.

5.2 Quantifying Player Variance and Covariance

5.2.1 Player Variance as a function of Skill Set

As we have stressed throughout the paper different players have different skill sets and derive value from different events. This can lead to players having the same average PPG but very different variances around that average. For example if we have one hitter that either hits a home run or gets out and another who hits a lot of singles and gets out less often it is possible that they will have similar averages. However the variance of the home run hitter will be much higher, as he doesn't record points as often but when he does he records multiple points. To investigate this we look at the variance of a player's point total as a function of what events he accumulates his points from. Here we show this relationship for the proportion of points from singles as well as the proportion of points from home runs. The results are very intuitive and show that players will have a different variance based on their skill set. When we are building a lineup we can now incorporte the variance of a player's point total.

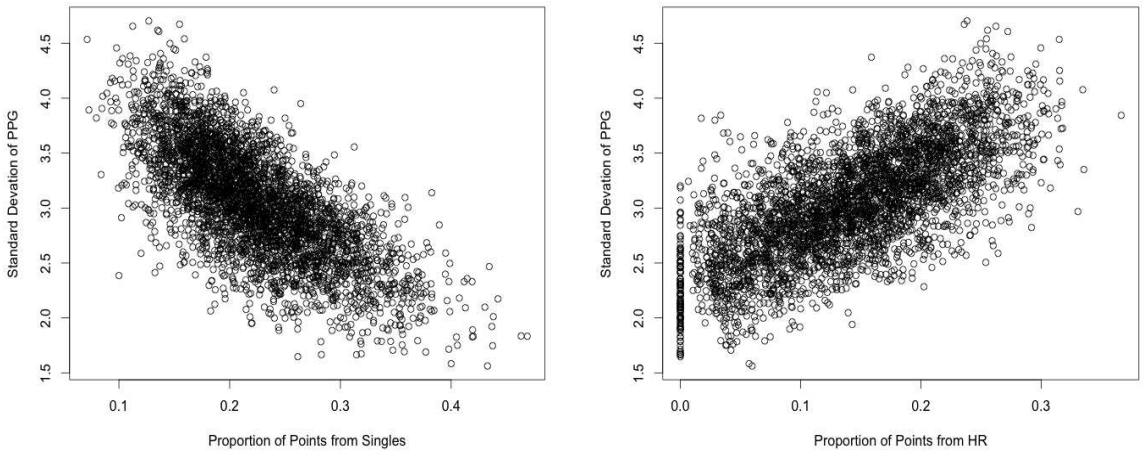

Figure 5.1: Standard Deviation of Player Point totals as a function of the proportion of points from singles (left) and home runs (right)

5.2.2 Different Types of Player Covariance

Player point totals can also be related to each other. Of course if players are not in the same game their point totals will be independent, but for players in the same game several factors will lead to a nonzero covariance. Estimating this covariance for different relationships will be necessary to compute the variance of a lineup. First we begin with the simple relationship of inter-team pitcher covariance. While the two opposing pitchers never compete directly in a game (except when one is hitting, but pitchers do not receive points for offensive events) they can influence each others point totals by changing the probability of their opponent recording a win. By definition if one pitcher records a win the other does not. This is impactful because a win is worth four points. The empirical covariance of opposing pitchers point totals is -7.56 points. There is also the relationship between

the hitters of one team and the pitcher of the other team. This is more straightforward as they are direct opponents and baseball is somewhat of a zero-sum game in that if a pitcher is doing well it means the opposing hitters must not be doing well on the whole. The empirical covariance here is -6.35. Furthermore there is a relationship between the hitters and pitcher on the same team. This again comes back to the pitcher's likelihood of recording a win. If his hitters are scoring a lot of runs he will be more likely to record a win, so we would expect this covariance to be positive, but not as strong as the preceding two. Indeed this is the case, with an empirical covariance of 1.59.

Lastly is the more subtle relationship between the hitters on the same team. One common factor that will lead to a positive covariance is that all of the hitters are facing the same pitcher. So if the pitcher is effective the performance of all the hitters will be depressed and vice versa. However there is another source of covariance due to the nature of the scoring system that needs to be addressed. In DFS the same offensive event can lead to points for multiple players. If there is a runner on second base and the hitter hits a single which scores the runner, the runner gets credit for a run while the hitter gets credit for the run batted in (RBI) as well as the single. So an event that is usually given one point (a single) results in three points being generated. This is why FanDuel limits the number of players one can have from the same team. By stacking a lineup with players on the same time one can take advantage of this 'double-counting' to achieve high point totals. Given the nature of this dependence one would expect that the covariance between point totals would be higher for hitters who are closer to each other in the batting lineup. It is also necessary to remember that a baseball lineup is cyclical in nature, meaning the 1-hitter is only two spots away from the 8-hitter, not seven. To quantify this relationship we computed the covariance matrix of the point totals by lineup spot for hitter on the same team.

1	2	3	4	5	6	7	8	9	
9.98	1.65	1.74	1.41	0.94	0.89	1.11	1.12	1.23	1
1.65	9.91	1.84	1.60	1.10	0.97	0.94	1.05	1.00	2
1.74	1.84	11.94	2.00	1.38	1.01	0.82	0.95	1.00	3
1.41	1.60	2.00	12.06	1.56	1.30	1.21	0.91	0.79	4
0.94	1.10	1.38	1.56	10.85	1.58	1.29	0.98	0.87	5
0.89	0.97	1.02	1.30	1.58	9.69	1.38	1.18	0.87	6
1.11	0.94	0.82	1.21	1.29	1.38	8.91	1.29	1.07	7
1.12	1.05	0.95	0.91	0.98	1.18	1.29	8.22	1.11	8
1.23	1.00	1.00	0.79	0.87	0.87	1.07	1.11	6.98	9

We see that the covariance between lineup spots does indeed decay as hitters get further apart from each other in the lineup. We also see that the covariance between players tends to be higher for hitters earlier in the lineup. This is not surprising as these tend to be the better hitters, meaning it is more likely for them to be involved in scenarios in which points are essentially double-counted. We calculated the average covariance as a function of the distance between the two lineup spots. The result shown in the table on the next page are very intuitive. The covariance decreases very linearly, with a change of roughly -0.20 points for every lineup spot. Now for each hitter on the same team we can quantify the covariance between their point totals by assigning them the appropriate value in the covariance matrix shown above.

Lineup Spots Away	1	2	3	4
Average Covariance	1.52	1.30	1.08	0.90

Table 5.1: Average Covariance as a function of Lineup Spots apart

5.3 The Distribution of a Lineup's Point Total

With these covariances defined we next turn to calculating the expectation and variance of a lineup's point total, as well as investigating the distribution of a lineups' point total. A lineup can be thought of as a sum of random variables where each random variable is the point total for a player in the lineup. By the linearity of expectation the expected point total of a lineup is the sum of the expected point totals for each player.

$$E\left(\sum_{i=1}^{9} Y_i\right) = \sum_{i=1}^{9} E(Y_i) \tag{5.1}$$

The variance is slightly more complex due to the potential lack of independence between players that we addressed in the previous section.

$$Var\left(\sum_{i=1}^{9} Y_i\right) = \sum_{i=1}^{9} Var(Y_i) + 2\sum\sum_{j<k} Cov(Y_j, Y_k) \tag{5.2}$$

Of course if every player in the lineup is independent from each other all the covariance terms will be zero. While these formulas describe the mean and variance of a lineup's point total they do not necessarily describe the full distribution. The lineup can be thought of as the sum of eight random variables that follow the skewed distribution of hitter point totals and one random variable that follows the normal distribution of pitcher point totals. We wish to randomly sample from these distributions to simulate many lineup point totals to understand how the distribution of a lineup's point total behaves. We know that if we were summing significantly more random variables that the distribution could be approximated by a normal distribution due to the Central Limit Theorem. But we are only summing nine random variables (eight of which are not normally distributed) so we can not immediately be sure the normal approximation is a sound one.

Before we can draw samples from these distributions we first must fit them. The distribution of pitcher point totals can be well described by a normal distribution with $\mu = 9.31$ and $\sigma^2 = 33.56$. The distribution of hitter point totals is fit by using a Generalized Pareto Distribution (GPD) as mentioned in Chapter 2. A one-sided GPD is characterized by a scale parameter λ, a shape parameter ξ and a threshold value m [16]. We fit a one-sided GPD here becasue only the right side of the distribution has a heavy tail. The threshold value dictates where the non-parametrically fit bulk of the data ends and the parametrically fit tail of the data begins. While the density of the one-sided GPD cannot be explicitly defined below the threshold, above the threshold the density is [16]:

$$f_{m,\lambda,\xi}(x) = \frac{1}{\lambda}\Big(1 + \frac{\xi}{\lambda}(x-m)\Big)^{1+\frac{1}{\xi}} \tag{5.3}$$

The GPD we fit has a threshold of m = 7, a shape parameter $\xi = 0.023$, and a scale parameter $\lambda = 2.64$. The GPD was fit using the *fit.gpd()* function in the R package 'Rsafd' [37]. The value of the threshold was chosen such that a sufficient percentage of data points (roughly 10% in this case) were in the tail.

We then use these two distributions to simulate lineup point totals to get a sense of the nature of the distribution. We show a representative QQ-plot of the quantiles of our simulated distribution against a normal distribution. While the plot shows that our distribution has a heavier upper tail due to the right-skewed GPD we are sampling from, a normal distribution is not a poor approximation. Furthermore the parametric nature of the normal distribution will allow us to more easily perform the analysis in the next section and while it may cause a slight change in the quantitative result it does not change the more important qualitative take away.

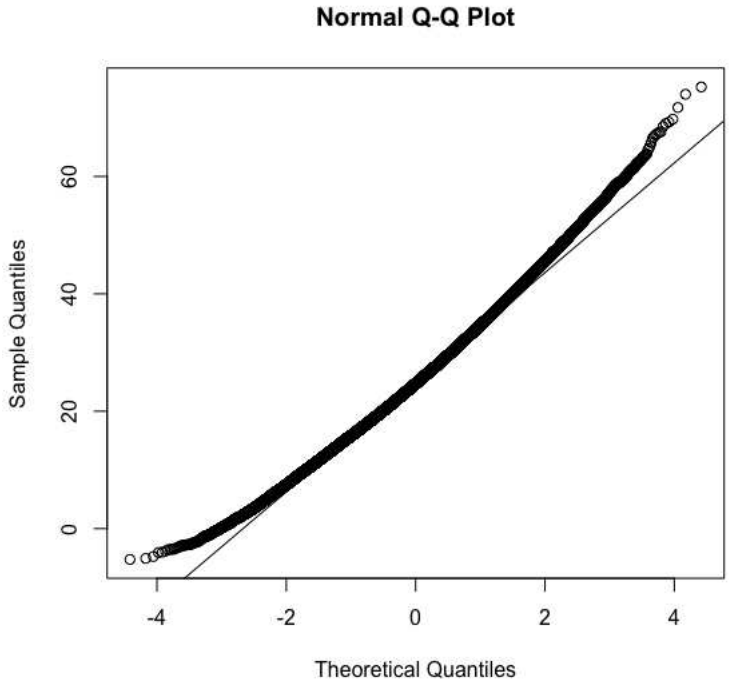

Figure 5.2: A QQ Plot of our simulated distribution against a Normal distribution

5.4 Probability of Winning for each Contest Type

5.4.1 Probability of Winning as a function of Points

A very important part of evaluating the value of a lineup is translating the point total to the probability of winning a contest. Of course this will vary by contest type as we discussed earlier. We

have also mentioned that this relationship is not perfectly defined due to a lack of data. However we can use some reasonable assumptions to make a function that should closely resemble the truth. We propose a logistic function to model the relationship between points and win probability. A logistic function is sensible because its range is (0, 1) and there is an inflection point where the marginal value of an additional point is greatest. The form of the function is as follows:

$$P(win) = f(points) = \frac{1}{1 + e^{-k(points - x_0)}} \tag{5.4}$$

The scalar x_0 is the point value of the inflection point and the scalar k reflects the steepness of the curve. A sensible choice for x_0 is whatever we believe to be the average point value needed to win a contest, as the probability of winning at x_0 is 0.50. The steepness reflects the marginal value of an additional point. Let us create three of these logistic functions that could reasonably reflect the probability of winning for the three different contest types.

Contest Type	H2H	50/50	Tournament
Inflection Point	30	30	60
Steepness	0.12	0.25	0.25

Table 5.2: Parameters of the Logistic Function by Contest Type

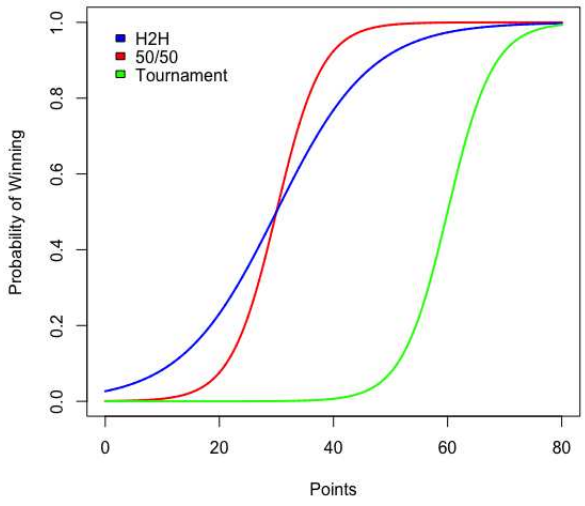

Figure 5.3: The Shape of the three Logistic Functions by Contest Type

The noteworthy differences are that it takes a significantly higher point total to win a tournament and that there is more variance surrounding the necessary point total required to win a H2H contest than a 50/50 contest because there is only one opponent rather than 19. We will now use these functions to determine the probability that a lineup with a given average and variance will win a specific contest type.

5.4.2 The Mean-Variance Trade-off by Contest Type

In most applications risk is considered a negative feature. This is most commonly seen in finance, where an asset with the same expected return but lower risk than a second asset is strictly preferred. One can easily draw parallels between the field of Portfolio Optimization and this application. If we think about each player as a risky asset with an expected return and variance, we can think of a lineup as a portfolio. Furthermore we have quantified the covariance between players allowing us to either hedge our risk or speculate based on the contest type. However in our application the value or risk is far more dynamic. As a thought experiment imagine a contest in which a participant wins if their lineup records more than 60 points. Now let us say they can choose between two lineups, both of which have random point totals that are normally distributed. Lineup 1 is N(44, 64) and Lineup 2 is N(40, 196). For the player to win using lineup 1 the lineup would have to perform two standard deviations above its mean. Lineup 2 would have to perform about 1.5 standard deviations above its mean, which is a far more likely outcome. So in this scenario a lineup with a *lower* expected return and *higher* volatility is preferred. In this way our application can diverge from the classical mean-variance trade-off where one must be compensated with a higher expected return to tolerate higher risk. We will now see that the contest type, specifically the corresponding logistic function relating points to win probability, dictates the nature of this trade-off.

If we know the expected value and standard deviation of a lineup, which we can calculate from equations 5.1 and 5.2 respectively, and we assume that the distribution of the lineup point total is normal, we can calculate the win probability. From probability theory we know:

$$P(win|\mu,\sigma) = \int_{-\infty}^{\infty} P(points = x)P(win|\mu,\sigma,points = x)dx \tag{5.5}$$

Since we are assuming the distribution of the lineup point total is normal, which is a reasonable but not perfect assumption as we saw in Section 5.3, $P(points = x)$ can be written as the Normal PDF with the mean μ and variance σ^2. Additionally, $P(win|\mu,\sigma,points = x)$ is simply the logistic function we specified above. So we rewrite equation 5.5 as:

$$P(win|\mu,\sigma) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \frac{1}{1+e^{-k(x-x_0)}} dx \tag{5.6}$$

For computational purposes it is beneficial to rewrite this as a sum. This allows us to easily calculate the probability of winning for many different combinations of μ and σ as we search for the optimal combination for each contest type. We sum over all possible points from zero to 100 as we have seen that a lineup will rarely finish outside of this range, and when it does the marginal change in win probability is negligible. The plots of win probability as a function of μ and σ for all three contest types are on the next page.

$$P(win|\mu,\sigma) = \sum_{x=0}^{100} \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \frac{1}{1+e^{-k(x-x_0)}} dx \tag{5.7}$$

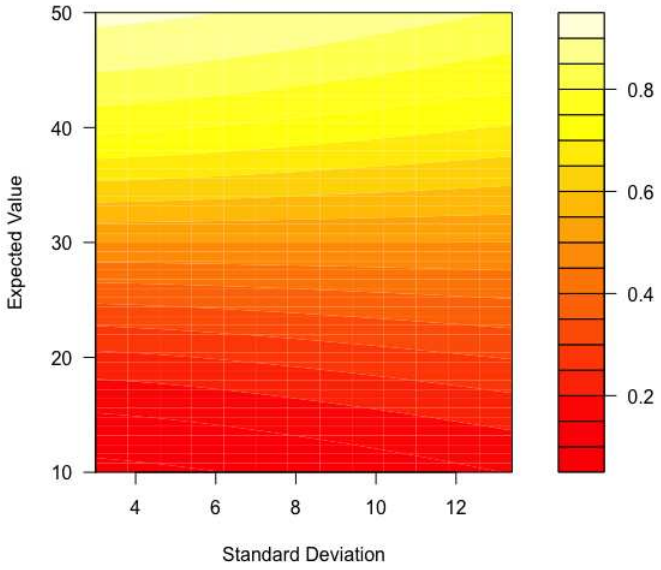

Figure 5.4: The Probability of winning a H2H contest as a function of lineup mean and standard deviation

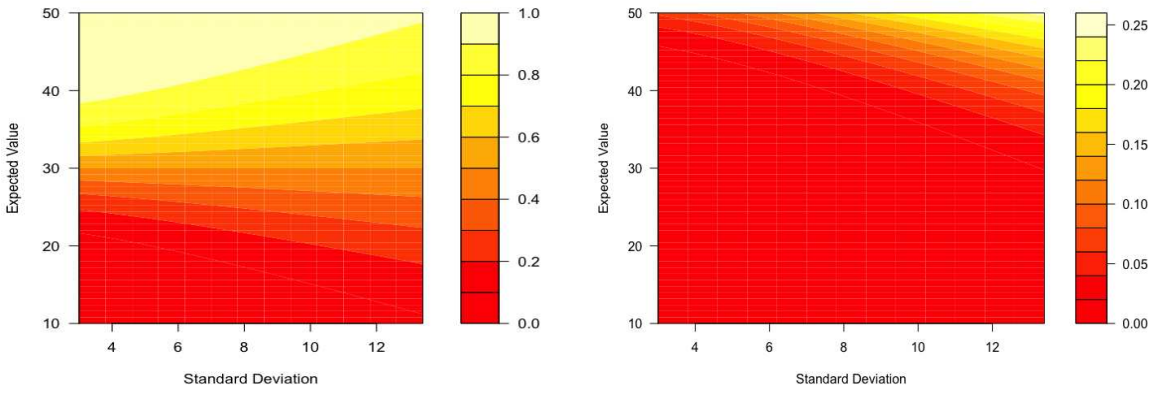

Figure 5.5: The Probability of winning a 50/50 contest (left) and a tournament (right) as a function of lineup mean and standard deviaition

In Figure 5.4 we see the win probability for H2H contests if we assume the logistic function given in Table 5.2. An obvious observation is that for any standard deviation, the win probability increases monotonically with expected value. A more interesting observation is the behavior of the contour lines with respect to the inflection point (30). Below an expected value of 30 the contour lines are downward sloping, indicating that one should sacrifice some expected value for a higher standard deviation. The opposite is true for expected values above 30. At an expected value of 30, the standard deviation has no impact on the win probability. This is an intuitively pleasing result and suggests that our choice of a logistic function is a reasonable one. If one can construct a lineup that is more likely than not to win (above the inflection point) just in terms of expected value they would like to minimize the variance to make sure they stay on the favorable side of the inflection point. But if someone has a lineup with an expected value below the inflection point they are willing to sacrifice expected return for increased volatility in an effort to move above the inflection point. So for H2H contests the direction of the mean-variance trade-off depends on the perceived mean.

At first glance the figure for 50/50 contests may look identical. This is because we assigned the two logistic functions the same inflection point, only changing the steepness. But upon closer inspection we see that the contour lines for 50/50 contests are steeper than for the H2H plot, both above and below the inflection point. This reflects the amplified risk-return trade-off relative to the H2H contest. In a 50/50 contest, because there are many entries, the point total required to finish in the top half will be more stable than in the H2H contest where the necessary point total depends on only one opponent. The increased steepness reflects this, showing that a participant with an expected value above the inflection point must be compensated by a higher increase in expected return to take on increased volatility in a 50/50 contest. But below the inflection point a participant is more willing to tolerate additional risk than in a H2H contest because they need to get over the inflection point if they want a chance. So again we see that the mean-variance trade-off is dependent on the mean itself. Finally we examine the win probability plot for a tournament, which we have assigned an inflection point of 60 points. This plot is starkly different from the other two given its very skewed payout structure and high point total necessary to win. Because the inflection point is not within the range of the plot (as no lineup constructed with sensible predictions will have an expected value of 60) there is no expected value at which higher variance is not beneficial. We should also note that the scale of this plot does not exceed a win probability of 0.25, while the other two contests have mean-variance combinations within the range of the plot that will win 100% of the time. Our choices for the range of both expected value and standard deviation were chosen based on the range of values observed in practice. We set the bounds on expected value by looking at the results of the optimization problem in Chapter 4. To determine the feasible range of standard deviation we ran two separate optimization problems, one minimizing lineup variance and the other maximizing it (while satisfying the lineup constraints). The average minimum standard deviation over all days was 3.44 and the average maximum standard deviation was 13.20. It is more than evident from these results that we need to modify our optimization approach to account for the important role that lineup variance plays in the context of each contest type.

5.5 The Modified Optimization Problem

In order to integrate the variance of a lineup into our optimization problem we must first build a covariance matrix Σ. This will be an NxN matrix where N is the number of players playing on a given day. It will also be a very sparse matrix because players in different games will have a covariance of zero. But for players in the same game whose relationship is mentioned in section 5.2.2, Σ_{ij} will take the appropriate value based on the relationship between player i and player j. Of course the diagonal entries of Σ will be each players observed variance. It will be useful to express the variance of a lineup, which is formulated in equation 5.2, in matrix notation. The equivalent expression is

$$Var(Lineup) = x^T \Sigma x \tag{5.8}$$

We can now formulate our new optimization problem that takes a lineup's variance into account. Note that the only difference is the modified objective function, as the lineup constraints imposed by FanDuel are still in place.

$$\begin{aligned} \max_{x} \quad & C^T x + \lambda x^T \Sigma x \\ \text{s.t.} \quad & S^T x \leq 35,000 \\ & Ax = b \\ & Tx \leq 4 \\ & x_j = \ 0 \text{ or } 1 \ \forall j \in N \end{aligned} \tag{5.9}$$

Now our objective function contains a term for expected points as well as the variance of the point total. The scalar λ can take any real value, positive or negative. Note that only the relative weights of the two terms matter so we do not need another scalar in front of $C^T x$. If λ is greater than zero we are encouraging a lineup with higher variance. The opposite is true for a negative value of λ. We can tune this parameter such that the balance between expected value and variance is optimal for each contest type. It is also useful to see over what range of λ the solution actually changes. At a certain negative value of λ the variance term will dominate leading the solution to be the minimum variance lineup. At a certain positive value of λ the variance term will again dominate leading the solution to be the maximum variance lineup. On the next page is a representative plot of lineup variance as a function of λ on a particular day. We can see that the range of λ over which the solution is changing is quite small. Outside of the range [-0.5, 0.5] the solution is essentially constant. This is due to the high value of lineup variance relative to expectation, which again reflects the high degree of uncertainty on a daily timescale. Because of this, λ does not have to have a high magnitude to force the solution to be one of the extreme variance lineups. In terms of computational time this will be helpful as we will see in the next section when we discuss how to choose the optimal λ for each contest type on a given day.

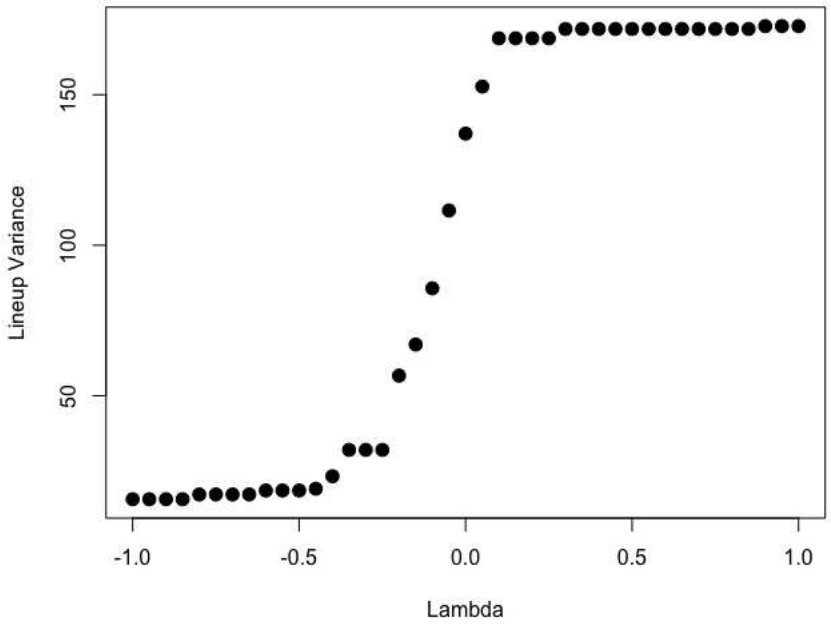

Figure 5.6: Lineup Variance as a function of Lambda

5.6 Optimal Lambda by Contest Type

If we vary λ over the range in which the solution is sensitive to its value we will get a different lineup for every value of λ. Each of these solutions will have an expected value represented by $C^T x^*$ and a variance represented by $x^{*T}\Sigma x^*$. So for each lineup we can calculate its win probability in each contest type using the method detailed in section 5.4.2. *The optimal lineup for each contest type will be the lineup whose mean and variance lead to the highest win probability.* So the optimal λ value, λ^* will be the value that produced the optimal mean-variance lineup. Because computational time is not an issue (each instance can be solved witiin 10 minutes) we can solve the optimization problem over a range of λ values and just choose the best solution. Once we have a sense of the range of optimal lambda values for each contest we can narrow our search and decrease the step size between λ values. This process of varying λ allows us to create our version of the efficient frontier, a well-known concept in Portfolio Optimization. The efficient frontier is the set of portfolios that give the highest expected return for a certain amount of risk [38]. We have done the exact same thing with lineups, with each λ value providing the optimal lineup given how we have weighted the value of risk. On the next page we can see an example of this for a particular day. The value of λ was varied from -0.5 (the left endpoint) to 0.5 (the right endpoint). We see that as λ increases standard deviation increases monotonically. And while the expected return intially increases as we allow more variance, as λ approaches 0.5 we see that return is sacrificed for additional variability.

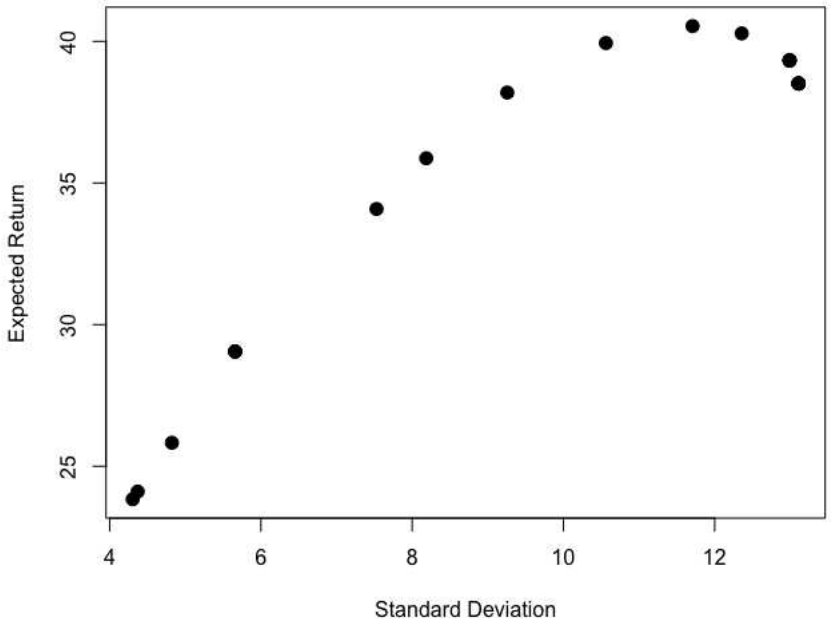

Figure 5.7: The Efficient Frontier for Daily Fantasy Baseball Lineups

We can use this to visually determine our optimal λ value for each contest type by continuing to draw from Portfolio Optimization theory. Once the efficient frontier has been determined the truly optimal portfolio is the portfolio on the curve that is tangent to the Capital Allocation line with the highest slope (this line is known as the Capital Market Line [39]). In our application we are looking for the lineup that is tangent to the highest contour line of win probability for each contest type. To see this we must overlay the efficient frontier on the various win probability plots in section 5.4.2. Here we just show the contour lines as opposed to the heat map for ease of visualization. It is immediately apparent that the optimal λ value will be different for each contest type, reflecting the relative value of variance. The optimal value, or range of values, of lineup mean and variance is shown by the green dots. The λ values that produced these lineups are displayed in the table below.

Contest Type	λ^*
Head to Head	0.00
50/50	-0.05 - 0.00
Tournament	0.05 - 0.10

Table 5.3: Optimal λ values by Contest Type

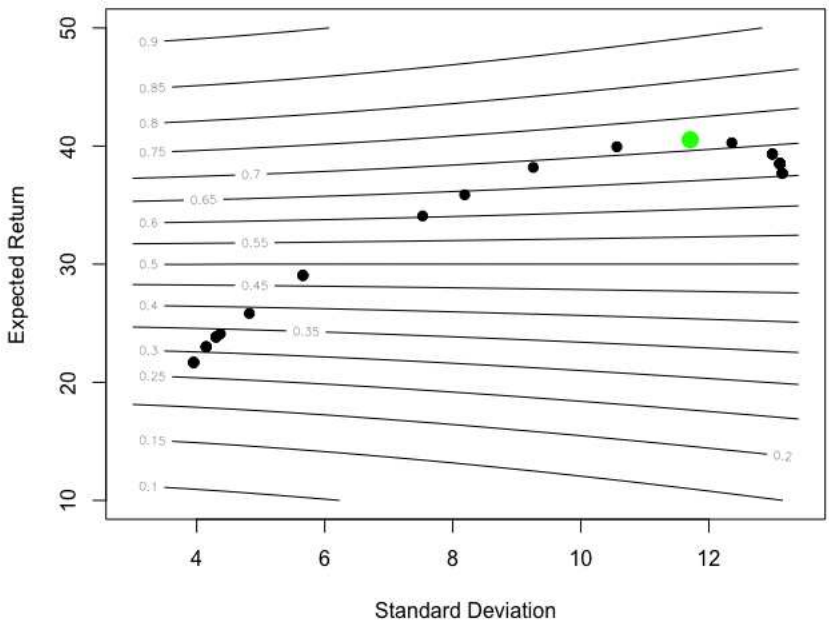

Figure 5.8: The Optimal Mean-Variance lineup for a H2H Contest

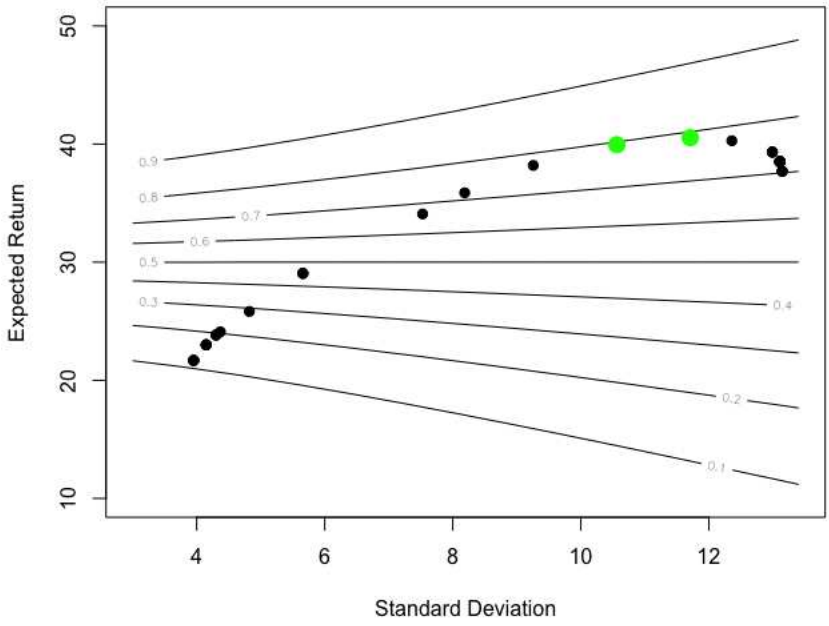

Figure 5.9: The Optimal Mean-Variance Lineup for a 50/50 Contest

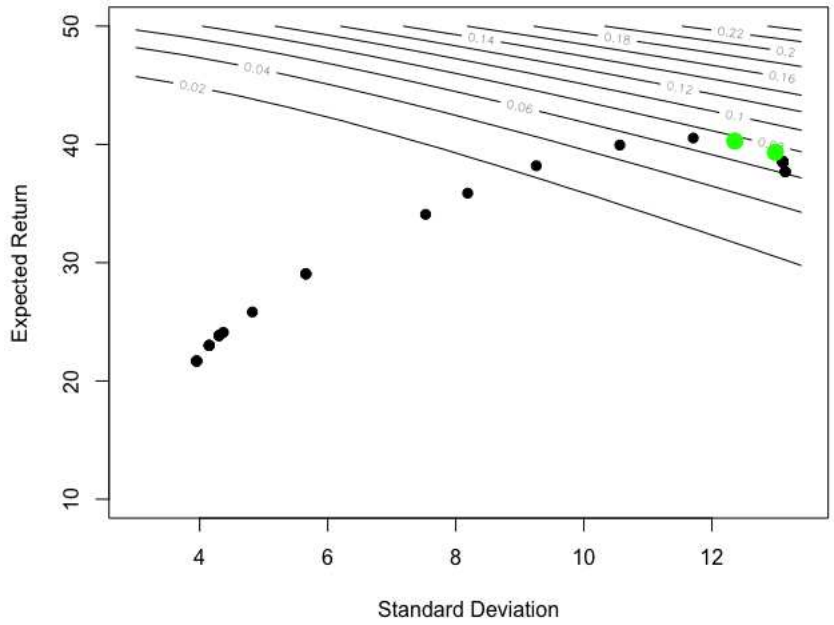

Figure 5.10: The Optimal Mean-Variance Lineup for a Tournament

These results are exactly what we were expecting. In a H2H contest it is best to just try to maximize expected points while disregarding lineup variance. Also note that the win probability of the highest feasible contour line is slightly above 0.72. For 50/50 contests it is beneficial to slightly limit lineup variance due to the increased steepness of the contour lines. We can see that the optimal lineup is found by a λ value somewhere between -0.05 and 0.00, being closer to -0.05. Again note the high win probability of 0.80. Finally we see that in tournaments one wants to really emphasize lineup risk, with the optimal lineup being very close to the maximum variance lineup. Although, even the optimal lineup only has a win probability of 0.08 due to the high point total required to win.

We have now completed our analysis of optimal lineup by contest type. We should however note a few key assumptions and limitations. First we have assumed that we know exactly the win probability as a function of point scored for each contest. We have also assumed that the distribution of point totals for a lineup can be approximated by a normal distribution. Finally we need to stress that the optimal λ value for a given contest can vary each day based on the location and curvature of the efficient frontier. While the contour lines will remain the same, there will be some days where a certain mean-variance lineup is feasible and other days when it is not. So one must tune λ every day, but will be able to narrow the range of possible values for each contest type over time with more observations of λ^*. Even with these limitations, which can be addressed, we have developed an approach that takes into account lineup variance to construct an optimal lineup for each contest type.

Chapter 6

Future Work and Conclusion

6.1 Future Work

6.1.1 Win Probability as a Function of Points Scored

Throughout Chapter 5 we had to make an assumption, albeit a reasonable one, about win probability as a function of points. It is extremely important that this function is determined as we have seen that it drives the behavior of the mean-variance trade-off that our approach is centered around. We recommend a Bayesian method for learning this function, drawn from the field of Optimal Learning [40]. This field deals with situations in which one is uncertain about the underlying truth and on top of that any observation drawn from the true function is noisy as well. Here the function we wish to learn is win probability as a function of points. The observation we can record is whether or not we won the contest with a particular point total. This binary outcome is of course extremely noisy, especially in the region of most interest where the win probability is close to 0.5. We can use the logistic function described in Chapter 5 as our prior belief about the true function. It would also be sensible to correlate our beliefs about the win probability at point values that are close to each other. For example if our prior is that the win probability of a 35 point lineup is 0.40, but 9/10 of our 35 point lineups have won, our belief about the win probability at 36 points should also change even if we have had no observation at that point total [40]. The function should also be constrained to be monotonically increasing with the point total.

In order to learn this function in the most efficient manner we would like to be able to see how lineups with a wide range of point totals perform. There are a few issues with this. One is that we cannot know in advance how a lineup will perform. We can only enter lineups with different *expected* point totals. This will cause some of our observations to be at undesired point totals. Another problem with entering lineups with low expected point totals is that we have to deal with the very real monetary consequences of likely losing that contest in order to learn about the function. This is an example of an online learning problem, where the consequences of each experiment (lineup entered) have a cost [40]. Lastly we have to determine how to choose the expected point total of the lineup we want to enter in order to learn. The method by which we choose this is known as a policy. Here we propose the use of the 'Knowledge Gradient' policy [40]. This policy maximizes the expected value of information that can be obtained from one additional experiment. Specifically we recommend the KG policy with correlated beliefs in an online setting. For the reader interested in

the details of this approach or more information about Optimal Learning in general please see [40].

6.1.2 Game Theoretic Implication of Lineup Selection

A defining feature of Daily Fantasy Sports, and Fantasy Sports in general, is that success is not determined by absolute performance but relative performance. The point totals of the other participants in the contest dictate how well a lineup does. So it is necessary to take into account how other participants may be choosing their lineups. As we mentioned in the Introduction and briefly in Chapter 4, websites that provide advice, projections, or even a proposed lineup can change the behavior of many participants. For example let us imagine a scenario where there is a website that provides what they believe to be the optimal lineup for a day and we know that every participant will choose that lineup exactly. How should we respond when constructing our lineup? The most sensible strategy would be to choose the exact same lineup, except replace the player that our prediction system is most bearish on *relative to the other prediction system* with the player we believe will perform the best that still satisfies all the FanDuel lineup constraints. By doing this we have turned a 9-on-9 contest into a 1-on-1 contest in which we believe we have a distinct advantage.

Of course the above scenario is unrealistic, but it shows that we should consider the proportion of other participants that we think will choose a given player when we choose our lineup. We can construct a variable to describe how popular each player seems to be across the various sources of information available. Whether or not we want to choose a player that is popular on a given day depends on our value of that player relative to the third-party values. If we too think this player is going to perform well we should choose him as we do not want to fall behind the pack. But if we disagree, we should not choose that player as we have an opportunity to go against the consensus and hopefully gain an edge. Mathematically this could be added to our optimization problem by including another term in the objective function that quantifies the expected similarity of our lineup to the rest of the field. We would also need to put a tunable parameter in front of this term that would change by contest type. For example to win a tournament one would have to build a lineup that goes against consensus in addition to having high variance.

6.1.3 Improving Predictive Performance

In Chapter 4 when we evaluated our prediction model we saw that by improving the predictive ability of our model by just 1.5% we could increase the expected point total of our lineup by about nine points. This high degree of sensitivity is motivation to continue refining our predictions. While it is certainly possible that there is another model type that could lead to improved accuracy it is more likely that more powerful explanatory variables will be the key to real improvements. There are some variables that we did not include that could prove useful. One is some measure of recent performance. Analysis has shown that players can indeed get 'hot' or 'cold' over short periods of time [25]. While the magnitude of this streakiness is not very large, it may still be enough to provide an edge on a daily timescale and is another example of a variable that could not be used in a season-long setting. Additionally one can consider the historical results of a specific batter-pitcher matchup. Work has shown that this information is only significant when a matchup has occurred dozens of times, which is quite rare [25]. But for those matchups this information may be useful. We also concede there are probably some other explanatory variables that would be meaningful that

we have simply neglected to think of. Finally the quantity and sophistication of data about baseball is continuing to grow. With the recent addition of Statcast [41] we can quantify many aspects of the game that were previously only analyzed qualitatively. This could lead to new statistics or relationships that were previously not understood, thus improving prediction.

6.1.4 Hedging Risk with Multiple Entries in a Contest

One feature of DFS tournaments that we did not address is that a participant is allowed to enter multiple lineups in the same tournament. For example, there are tournaments with a total size of 20,000 where one person could enter up to 250 lineups. And given the nature of the payout structure, if one entered 250 lineups and one won while all others finished out of the money, the payout multiple would still be close to 10x ($1 entry fee/lineup, $2000 for first place). By entering multiple lineups one is creating a portfolio of portfolios, or a 'fund of funds' as it is known in finance. Now instead of treating each player as an asset we treat each lineup as an asset with an expected return and variance as calculated in Chapter 5. To understand the relationship between the point totals of any two lineups we must compute the covariance between lineups as follows:

$$Cov\Big(\sum_{i=1}^{9} Y_i, \sum_{j=1}^{9} X_j\Big) = \sum_{i=1}^{9}\sum_{j=1}^{9} Cov(Y_i, X_j) \tag{6.1}$$

where $Cov(Y_i, X_j)$ can be obtained from Σ. For tournaments we would like to find lineups with a very high negative covariance so that at least one of them is likely to do well. Finding such a pair of lineups is not trivial. Additionally there is the question of the optimal number of lineups to enter which will not be independent from the magnitude of the covariances between lineups. At a certain point adding an additional lineup may not diversify the portfolio in a significant enough way to justify the additional entry fee. Lastly is the simple matter that as one enters more lineups the probability of winning increases just because there is one less competing lineup. This is clearly a very complex problem, but is most definitely worthwhile given the extremely high payouts that tournaments offer.

6.2 Conclusion

This paper has developed a quantitative process to construct an optimal lineup in Daily Fantasy Baseball. Furthermore we have extended this analysis to make our solution specific to the payout structure of the three most popular contest types. We have shown that this approach performs well and can be implemented.

The emergence of Daily Fantasy Baseball has greatly altered the strategy involved in choosing a Fantasy team. The shorter timescale results in much more uncertainty, but at the same time allows participants to have access to detailed and timely data that can shift our predictions significantly from the baseline for each player. In Chapters 2 and 3 we leveraged this added context to develop predictive models for player point totals. This was done using Generalized Boosted Models, a non-parametric regression technique. These predictions were used in the objective function of our optimization as we attempted to maximize expected points.

In Chapter 4 we formulated an optimization problem with the constraints modelling the restrictions FanDuel places on lineups. We saw that our predictions resulted in significantly better lineups than a 'naive' approach that did not take advantage of game-day information. We also found that our approach produced lineups with point totals far above the 50th percentile of point totals for Head-to-Head contests, meaning our chances of winning any given contest were firmly above 50%. This suggests this strategy should be implemented in these contests with profitable results.

We then proceeded to modify this optimization problem in Chapter 5 by incorporating the variance of a lineup into our framework. In this way we are able to use the increased variance of a daily timescale to our advantage. This allowed us to tailor our strategy to each contest type by weighting the value of risk differently. We also drew on several principals of Portfolio Optimization such as the efficient frontier and the Capital Market Line to guide our decision making. We found that each contest type had a different λ^* value as a result of the different payout structures. In Chapter 6 we discuss several fruitful areas for future work in this very undeveloped area.

As far as we can tell this work is the first of its kind in the public domain. While plenty of websites have developed prediction models for DFS point totals and have implemented an Integer program to find the 'optimal' lineup no one has modified this approach to take into account lineup variance. Even without this modification our prediction system seems to be on par with the ones currently out there as the typical expected values of our optimal lineup are similar to theirs. In this still very young area of study this paper has made a valuable contribution by developing a sound mathematical framework for optimal lineup construction in Daily Fantasy Baseball.

Chapter 7

References

[1] Fantasy Sports Trade Association. http://fsta.org/research/industry-demographics/

[2] Nicholas David Bowman, John S.W. Spinda, and Jimmy Sanderson. *Fantasy Sports and the Changing Sports Media Industry: Media, Players, and Society.* March 2016.

[3] Ryan Devault. *ESPN ends Fantasy Sports Prize Leagues: Daily Fantasy leagues to blame?.* February 26th, 2016. Inquisitir. http://www.inquisitr.com/2830998/espn-ends-fantasy-sports-prize-leagues-daily-fantasy-leagues-to-blame/

[4] Brad Tutlte, textitWhy Betting on Fantasy Sports is Legal but Betting on Regular Sports is not. September 10th, 2015. Time Magizine. http://time.com/money/4029443/fantasy-sports-betting-legal/

[5] Joe Drape and Jacqueline Williams. *Scandal Erupts in the Unregulated World of Fantasy Sports.* October 5th, 2015. The New York Times. http://www.nytimes.com/2015/10/06/sports/fanduel-draftkings-fantasy-employees-bet-rivals.html

[6] Ed Miller and Daniel Singer. *For daily fantasy-sports operators, the curse of too much skill.* September, 2015. http://www.mckinsey.com/insights/media-entertainment/for-daily-fantasy-sports-operators-the-curse-o f-too-much-skill

[7] SaberSim. https://www.sabersim.com

[8] RotoGrinders. https://rotogrinders.com/

[9] Retrosheet. http://www.retrosheet.org/

[10] Sean Lahman Database. http://www.seanlahman.com/baseball-archive/statistics/

[11] FanGraphs. http://www.fangraphs.com/

[12] Tyler Kepner. *No runs, no hits, new era: Baseball Ponders legal ways to increase offense.* April 15th, 2015. The New York Times. http://www.nytimes.com/2015/04/05/sports/baseball/baseball-2015-preview-in-apitching- rich-era-baseball-ponders-legal-ways-to-boost-offense.html

[13] FanGraphs Library, Projection Systems. http://www.fangraphs.com/library/principles/projections/

[14] Evan Drellich. *Astros' Formula for success builds on its own data bank.* March 8th, 2014. The Houston Chronicle. http://www.houstonchronicle.com/sports/astros/article/Astros-formula-for-success-builds-on-its-own-5300746.php

[15] FanDuel Scoring and Rules. https://www.fanduel.com/rules

[16] Carmona, Rene. *Statistical Analysis of Financial Data in R.* Published by Springer, 2014.

[17] FanGraphs Library, WAR. http://www.fangraphs.com/library/war/

[18] Steve Staude, *Better Matchup Data: Forecasting Strikeout Rate.* June 12th, 2013. FanGraphs http://www.fangraphs.com/blogs/bettermatch-up-data-forecasting-strikeout-rate/

[19] Matt Klaassen. *Estimating Hitter Platoon Skill.* February 8th, 2010. FanGraphs. http://www.fangraphs.com/blogs/estimating-hitter-platoon-skill/

[20] FanGraphs Library, Park Factors. http://www.fangraphs.com/library/principles/park-factors/

[21] Alan Nathan. *The Physics of Baseball.* http://baseball.physics.illinois.edu/

[22] Chris Constancio. *Temperature Effects.* October 23rd, 2006.
The Hardball Times. http://www.hardballtimes.com/temperature-effects/

[23] Gerald Schifman. *Cold Weather, Positons and Penalties.* January 7th, 2016. The Hardball Times. http://www.hardballtimes.com/cold-weather-positions-and-penalties/

[24] Max Weinstein. *Who Deserves Credit for Throwing Out Baserunners?* July 18th, 2003. Beyond the Box Score. http://www.beyondtheboxscore.com/2013/7/18/4522508/who-deserves-credit-for-throwing-out-baserunners

[25] Tom Tango, Mitchel Litchman, and Andy Dolphin. *The Book: Playing the Percentages in Baseball.* April 28th, 2014.

[26] Hastie, T and Tibshirani, R. *Generalized Additive Models.* Statistical Science, 1986 Vol. 1, No. 3, 297-318

[27] Eamonn Keogh and Abdullah Mueen. *Curse of Dimensionality.* The Encyclopedia of Machine Learning pp 257-258.

[28] Greg Ridgeway. *Generalized Boosted Models: A guide to the GBM package.* August 3, 2007

[29] Greg Ridgeway. *Package 'gbm'.* https://cran.r-project.org/web/packages/gbm/gbm.pdf

[30] MLB Official Rules.

[31] Mike Petriello. *Game Changers: No nore 'starter' and 'releiver' labels.* November 23rd, 2015. MLB.com. http://m.mlb.com/news/article/157383440/starting-pitchers-should-throw-fewer-innings

[32] Breiman, L. and Friedman, J. H. (1997), *Predicting Multivariate Responses in Multiple Linear Regression.* Journal of the Royal Statistical Society: Series B (Statistical Methodology), 59: 3–54.

[33] Dave Hall. *2015 Daily Fantasy Baseball Data.* http://rotoguru1.com/base/mlb-dbd-2015-notes.txt

[34] FanGraphs Library, Positional Adjustment. http://www.fangraphs.com/library/principles/park-factors/

[35] Gurobi Optimization R package. https://www.gurobi.com/documentation/6.5/refman/solving-models-with-the-gu.html

[36] *The Numbers Game: Analyzing Daily Fantasy MLB Data.* 2014. https://rotogrinders.com/articles/the-numbers-game-analyzing-daily-fantasy-mlb-data-135463

[37] Rene Carmona. *The 'Rsafd' package.* http://www.princeton.edu/ rcarmona/SVbook/svbook.html

[38] Markowitz, H. (1952) *Portfolio Selection.* The Journal of Finance, Vol. 7, No. 1, pp. 77-91. March. 1952.

[39] *Capital Market Line.* Investopedia. http://www.investopedia.com/terms/c/cml.asp

[40] Powell, W and Ryzhov, I. *Optimal Learning.* Wiley Series in Probability and Statistics, 2012.

[41] Paul Casella. *Statcast Primer: Baseball will never be the same.* April 24th, 2015. MLB.com. http://m.mlb.com/news/article/119234412/statcast-primer-baseball-will-never-be-the-same

Appendices

Appendix A

Predicting FanDuel Salaries

Below we show the results of the linear regressions used to predict a player's FanDuel salary. The first table is for hitters and the second is for pitchers.

Variable	Coefficient	p-value
Intercept	1980.33	$< 2^{-16}$
PPG	354.27	$< 2^{-16}$
wOBA	351.35	0.00102
HR	22.93	$< 2^{-16}$
SB	9.36	$< 2^{-16}$
Lineup Spot	-47.45	$< 2^{-16}$
Position	-18.25	$< 2^{-16}$

Table A.1: Regression Results for hitters

Variable	Coefficient	p-value
Intercept	993.02	0.074
W	47.058	$< 2^{-16}$
Games Started	-87.45	$< 2^{-16}$
Innings Pitched per Start	22.60	$< 2^{-16}$
K per 9 innings	201.09	$< 2^{-16}$
PPG	3917.36	$< 2^{-16}$

Table A.2: Regression Results for pitchers

As we mentioned in chapter 4 the R^2 value for the hitter model was 0.56 and the R^2 for the pitcher model was 0.71. The residual standard errors were 465.4 and 934.6 respectively.

Appendix B

Optimization Code and Output

```
#function that returns the optimal lineup
#specifically it returns a list where the first element is the data from that day
#and the second element is the output of the Gurobi Optimzation command
#takes a season (integer), date (4 character string), risk tolerance (lambda),
#vector of predictions, constraint matrices, and the covariance matrix

opt_lineup <-function(season, date, lambda, Team_mat, Salary_mat, Position_mat, Cov_mat, C_pred)

  #extracting data from the desired day
  daily_data = opt_final[opt_final$season==season & opt_final$date == date, ]
  #treat all OF the same
  daily_data$field = ifelse(daily_data$field>6, 7, daily_data$field)
  #N is the number of players to choose from
  N = nrow(daily_data)
  #rename the quadratic constraint matrix



  linear_constraint = as.matrix(rbind(Team_mat, Salary_mat, Position_mat))
  A = linear_constraint
  #the objective vector is projected points
  C = C_pred

  ##############
  #building the Gurobi model
  model = list()
  #defining the objective function
  model$modelsense = "max"
  model$obj = C
  #including the risk parameter lambda
  model$Q = lambda*Cov_mat
```

```
  #setting the linear constraint
  model$A = A
  #quick way to define equality signs and RHS constraints for
  #an arbitrary number of teams
  relation = c()
  right_side = c()
  for(i in 1:nrow(A)) {
    #positions
    if(i <= 6)  {
      relation = append(relation, "=")
      right_side = append(right_side, 1)
    }
    #you get 3 OF
    if(i==7) {
      relation = append(relation, "=")
      right_side = append(right_side, 3)
    }
    #for teams
    if(i > 7 & i < nrow(A)) {
      relation = append(relation, "<=")
      right_side = append(right_side, 4)
    }
    #salary
    if(i==nrow(A)) {
      relation = append(relation, "<=")
      right_side = append(right_side, 35000)
    }

  }

  #setting the right hand side of the constraints
  model$sense = relation
  model$rhs = right_side
  #all variables are binary
  model$vtype = rep("B", ncol(A))
  #the result of the model
  model_result = gurobi(model, params = NULL)
  output = list(daily_data, model_result)
  #return the data and the optimization result
  return(output)
}
```

```
Warning for adding variables: zero or small (< 1e-13) coefficients, ignored
Optimize a model with 35 rows, 206 columns and 618 nonzeros
Coefficient statistics:
  Matrix range    [1e+00, 1e+04]
  Objective range [7e-01, 2e+01]
  Bounds range    [1e+00, 1e+00]
  RHS range       [1e+00, 4e+04]
Found heuristic solution: objective 25.5584
Presolve removed 3 rows and 9 columns
Presolve time: 0.00s
Presolved: 32 rows, 197 columns, 565 nonzeros
Variable types: 0 continuous, 197 integer (197 binary)

Root relaxation: objective 4.062406e+01, 15 iterations, 0.00 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0   40.62406    0    2   25.55835   40.62406  58.9%     -    0s
H    0     0                       40.3955250   40.62406  0.57%     -    0s
H    0     0                       40.5387761   40.62406  0.21%     -    0s
     0     0   40.60439    0    4   40.53878   40.60439  0.16%     -    0s

Cutting planes:
  Cover: 1

Explored 0 nodes (16 simplex iterations) in 0.03 seconds
Thread count was 8 (of 8 available processors)

Optimal solution found (tolerance 1.00e-04)
Best objective 4.053877613176e+01, best bound 4.053877613176e+01, gap 0.0%
```

Figure B.1: Example of the output of the Gurobi Optimization Code