MLB Moneylines as Investment Assets

Robert Axelsen[†]

November 6, 2021

Abstract

In this paper, I apply prediction market theory to Major League Baseball (MLB) moneyline pricing. Applying various machine learning models to a comprehensive data set of past game and player data, I calibrate probability estimates of teams' chances to win games. With these probability estimates, I backtest profitable investment strategies using modified versions of the Kelly criterion staking strategy. Finally, I implement a profitable real-world betting strategy using the techniques developed herein over the first three months of the 2021 MLB season.

[†]Princeton University: Master in Finance (M.Fin.)

Prediction markets appear to be undergoing an interesting renaissance period currently, providing the public with more accurate and timely probability estimates of events occurring than has been possible historically. In addition to the broadly distributed informational benefits, the explosion in popularity of websites like PredictIt, FTX, and PolyMarket also allow forecasters to generate returns (or losses) proportional to their prescience in many different domains. Curiously, sports betting, a domain which demands largely similar analysis techniques to prediction markets more broadly, appears to have received relatively little attention. In particular, baseball moneyline betting remains largely unexplored in the academic literature despite its similarities to political betting and binary options trading. While there has been some detailed exploration into notions of market efficiency in the MLB moneyline market, the literature on betting strategies themselves is extremely sparse. Building on these past explorations into the structural characteristics of the MLB moneyline market, this paper explores the analytical challenges of accurately predicting game outcomes and whether these challenges and other practical implementation challenges can be overcome in pursuit of a profitable investment strategy.

The problem of constructing a profitable investment strategy can be broken down into two components: the probability prediction component and the investment decision. Because MLB games can only have two outcomes (away team wins or home team wins), the problem of predicting game outcome probabilities is naturally conceptualized as a probabilistic classification problem. Thus, in attempting to accurately predict game outcomes, I apply an assortment of the classic probabilistic classifiers (logistic regression, penalized logistic regression, neural networks, support vector machines, boosted trees, etc.) to a feature set primarily composed of historical player statistics. Given a quality dataset, it appears that almost any model can easily achieve better-than-chance performance. However, given the investment use-case of these predictions, performance must be benchmarked versus historical sportsbook predictions in order to ascertain whether a given model can be implemented profitably, a benchmark that this paper reveals to be quite difficult to beat. Specifically, I compare model outputted probabilities against sportsbook "implied" probabilities, using calibration plots to differentiate relative performance over different subintervals of $[0, 1]$. Given this benchmark, I find *xgboost* classifiers with hyperparameters tuned via cross-validation to generally outperform other models tested. Once one has a model that reliably outputs well-calibrated probability estimates, the task becomes finding a mapping between (model probability estimates, sportsbook line) to the final investment decision: how much to bet on any given game. To approach the investment decision problem, I apply the Kelly criterion in a relatively straightforward manner, also trying some slight modifications to account for violations in its assumptions. While the investment decision problem is interesting in its own right, moneyline pricing is so conceptually basic relative to most financial assets that almost all the difficulty in the overall task is in the first step: constructing a sufficiently predictive probabilistic classifier. Prices and

probabilities are one in the same in this market, and so, once one has such a classifier, almost any sensible "staking algorithm" will yield profitable results. As a true out-of-sample test, the last section of the paper is spent discussing a successful real-world application of the concepts presented in the previous sections. In particular, I detail a specific model and staking algorithm I used over the first three months of the 2021 MLB season to generate 131% returns betting against New Jersey sportsbooks.

I Key Principles of Moneyline Betting and Literature Review

A Key Principles

Many different types of wagers are available on sports games, and while the most popular type of bet in many sports is the point spread, the predominant wager type available in the baseball betting market is the moneyline. Moneyline betting is simply wagering on which team will win the game, with payouts awarded based on the sportsbook's estimate of each team's probability of winning the game. Correctly predicting a lower probability outcome is associated with a greater payout. In a fair, risk-neutral world, a bet's "payout multiplier" will be the reciprocal of the probability of the outcome occurring. Equivalently, this fair bet priced by risk-neutral agents has precisely 0 expected return. This formula is the fundamental relationship governing moneyline pricing.

$$m = 1/p \tag{1}$$

While many prediction markets simply quote these payout multipliers (decimal odds), baseball moneyline betting is quoted using the American odds format, which gives information regarding one's payout in the event of a correct bet. While this convention adds an arguably unnecessary layer of complexity, fortunately one can easily convert one-to-one into an equivalent implied probability of the team winning a given game and a payout multiplier. Positive odds display the net amount earned on the bet for each \$100 wagered while negative odds display the required wager to return \$100. The team quoted in negative odds is considered the favorite team while the team quoted in positive odds is the "underdog." For example, consider a hypothetical game between the Boston Red Sox and the New York Yankees for which the following odds are quoted by a sportsbook:

New York Yankees (+104) @ Boston Red Sox (-114)

In this scenario, if one placed a \$100 moneyline wager on the Yankees to win, he would receive \$104 plus his initial wager of \$100 for a total of \$204 if the Yankees win and would receive nothing if the Yankees lose. If one bet \$100 on the Red Sox to win, the bet would return (\$100/\$114)*\$100 = \$87.72 plus the initial wager of \$100 for a total of \$187.72. The following table illustrates the two possible moneyline bets on this game and payouts of a \$100 wager under each outcome:

	Yankees win	Red Sox win
Bet on Yankees	\$204.00	\$0
Bet on Red Sox	\$0	\$187.72

Table 1: Possible outcomes and associated payouts.

Table 1 reveals the underlying structure of these assets: moneyline bets are binary options which settle at the end of the game. The sportsbook implied probability of each team winning is simply the probability which gives each bet an expected net return of \$0 (gross return of \$100). That is,

$$\begin{cases} \$100 = \mathbb{P}_a(\text{Yankees win}) * \$204.00 + \mathbb{P}_a(\text{Red Sox win}) * \$0 \\ \$100 = \mathbb{P}_h(\text{Yankees win}) * \$0 + \mathbb{P}_h(\text{Red Sox win}) * \$187.72 \end{cases} \tag{2}$$

$$\implies \begin{cases} \mathbb{P}_a(\text{Yankees win}) = .4902 \\ \mathbb{P}_h(\text{Red Sox win}) = .5327 \end{cases} \tag{3}$$

One can quickly convert from American odds to implied probabilities for both the home and away side in the following way:

$$\mathbb{P}_{imp}(\text{Team Wins}) = \begin{cases} \dfrac{100}{100 + \text{American odds}}, \text{ American odds are positive} \\ \\ \dfrac{-\text{American odds}}{-\text{American odds} + 100}, \text{ American odds are negative} \end{cases} \tag{4}$$

where $\mathbb{P}_{imp}$ is the general form of $\mathbb{P}_a$ and $\mathbb{P}_h$. In the above example,

$$\begin{cases} \mathbb{P}_a(\text{Yankees win}) = \dfrac{100}{100 + 104} = .4902 \\ \\ \mathbb{P}_h(\text{Red Sox win}) = \dfrac{114}{114 + 100} = .5327 \end{cases} \tag{5}$$

which are the same implied probabilities obtained just above.

Note the importance of differentiating between the away side (a) probabilities and the home side (h) probabilities. One cannot use the same set of probabilities in both equations as there is no such feasible set (they would sum to more than 1). This is simply a consequence of a divergence between the payouts the sportsbook offers on a given game and the "fair payout." Equivalently,

$$\mathbb{P}_a(\text{Yankees win}) + \mathbb{P}_h(\text{Red Sox win}) > 1. \tag{6}$$

I define the margin on an individual game as the difference between this sum of probabilities and 1:

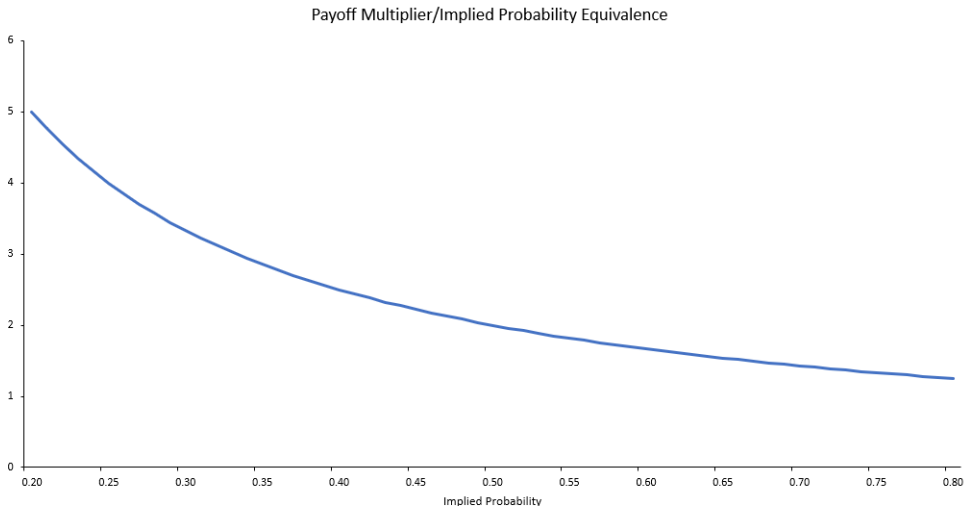
$$margin = \mathbb{P}_a(\text{Away Team wins}) + \mathbb{P}_h(\text{Home Team wins}) - 1. \tag{7}$$

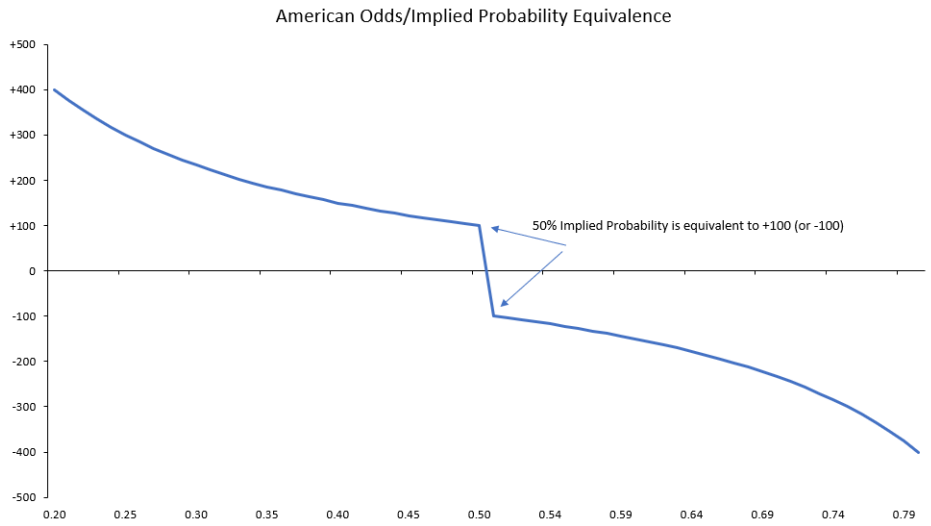
The margin on a particular line is a measure of the implicit cost a sportsbook imposes on the bet through artificially lower payouts. To the extent possible, it should be minimized (by selecting a sportsbook with low margin) in order to maximize profitability. In practice, margin is usually between 2% and 5%.

The away and home payout multipliers are defined as the gross amount received on a correct \$1 bet. These quantities are equivalently the reciprocals of $\mathbb{P}_a$ and $\mathbb{P}_h$, respectively:

$$\begin{cases} m_a = 1/\,\mathbb{P}_a(\text{Away Team wins}) \\ m_h = 1/\,\mathbb{P}_h(\text{Home Team wins}) \end{cases} \tag{8}$$

To reiterate the earlier point, it is crucial to recognize that payout multipliers and implied probabilities are more than loosely related. In fact, they are two ways of representing the same quantity; as the above equations show, they are inversely proportional. When thinking in terms of payoff multipliers rather than implied probabilities, one can see the effect of margin from a more direct vantage point: the implied probabilities are artificially high (they sum to more than 100%) and so the payoff on a correct prediction is artificially low (if one bet \$1 on each side of the game, he would receive less than \$2 back). In other words, MLB moneylines are generally negative expected return assets when accounting for the sportsbook's cut.





A.1 Fair Implied Probabilities

In order to differentiate the effect of sportsbook skill in predicting probabilities and the effect of simply imposing implicit costs, the concept of fair implied probabilities is useful. Coercing the implied probabilities to sum to one gives the "fair" implied probabilities, a more accurate estimate of the sportsbook's true outlook on the game. Mathematically, the fair implied probabilities remove the effect of margin as follows:

$$\begin{cases} \mathbb{P}_f(\text{Away Team wins}) = \mathbb{P}_a(\text{Away Team wins})/(1 + margin) \\ \mathbb{P}_f(\text{Home Team wins}) = \mathbb{P}_h(\text{Home Team wins})/(1 + margin) \end{cases} \tag{9}$$

In the example above, the fair implied probabilities are:

$$\begin{cases} \mathbb{P}_f(\text{Yankees win}) = .4792 \\ \mathbb{P}_f(\text{Red Sox win}) = .5208 \end{cases} \tag{10}$$

In this example, the sportsbook's true opinion is that the Yankees have a 47.92% chance of winning the game; however, they will reward a bettor who correctly predicts this outcome only as if the probability is 49.02%, reducing his payout multiplier from 2.09 to 2.04. In this sense, inverting these fair probabilities will give payoff multipliers not available in reality; these are not the relevant probabilities against which to compare the outputs of a predictive model. However, they will be useful in assessing the historical calibration of sportsbook implied beliefs with actual game outcomes and can be themselves used as predictive features.

A.2 The Connection to the One-Period Pricing Model

An alternative way of conceptualizing the available bets and associated payouts derives from formal asset pricing (Brunnermeier, 2014). One can formalize the two moneyline bets available on each baseball game as a market with two securities (each moneyline bet) and two possible future states of the world (home team wins or away team wins - there are no draws in baseball). Letting X denote the security structure and $\vec{p}$ denote the price vector (in this case, the amount wagered), one finds that

$$\vec{q} = \begin{bmatrix} \mathbb{P}_a(\text{Away Team wins}) \\ \mathbb{P}_h(\text{Home Team wins}) \end{bmatrix} \tag{11}$$

constitutes a state price vector.

To illustrate, consider the 2x2 market in which the two available securities are \$1 moneyline bets on the Yankees and the Red Sox to win, respectively. Then, for $\vec{q}$ to be a state price vector, it must be that $\vec{p} = X^T \vec{q}$ and indeed, by construction:

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} m_a & 0 \\ 0 & m_h \end{bmatrix} \begin{bmatrix} \mathbb{P}_a(\text{Away Team wins}) \\ \mathbb{P}_h(\text{Home Team wins}) \end{bmatrix} \tag{12}$$

$$\iff \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2.04 & 0 \\ 0 & 1.88 \end{bmatrix} \begin{bmatrix} .4902 \\ .5327 \end{bmatrix} \tag{13}$$

It directly follows from this framework that one can think of moneyline bets as Arrow-Debreu securities, which could be potentially useful in exploring the theoretical characteristics of the moneyline market.

A.3 Expected Return and "Beating the Line"

A number of probabilities have been introduced thus far. From this point forward, I will reference the simplified notation of these probabilities shown below:

Description	Probability	Shorthand
Home Team Implied Win Probability	$\mathbb{P}_h$(Home Team wins)	p_h
Away Team Implied Win Probability	$\mathbb{P}_a$(Away Team wins)	q_a
Home Team Fair Implied Probability	$\mathbb{P}_f$(Home Team wins)	$p_f = p_h/(1 + margin)$
Away Team Fair Implied Probability	$\mathbb{P}_f$(Away Team wins)	$q_f = q_a/(1 + margin)$
Home Team True (Oracle) Probability	$\mathbb{P}$(Home Team wins)	$\tilde{p}$
Away Team True (Oracle) Probability	$\mathbb{P}$(Away Team wins)	$\tilde{q}$
Home Team Model Output Probability Estimate	$\mathbb{P}_{model}$(Home Team wins)	$\hat{p}$
Away Team Model Output Probability Estimate	$\mathbb{P}_{model}$(Away Team wins)	$\hat{q}$

Table 2: Shorthand probability notation.

The only pair of probabilities which may sum to more than 100% is (p_h, q_a). Throughout the paper, I assume that true probabilities $(\tilde{p}, \tilde{q})$ exist (i.e., the outcome is not predetermined) but are unobservable. Thus, perfect efficiency $(p_f = \tilde{p},\ q_f = \tilde{q})$ has probability 0 for any given game. Under this framework, the expected net returns of betting \$1 on the away team and the home team to win, respectively, are:

$$\begin{cases} \mathbb{E}[\text{Away Bet Return}] = \tilde{p} * 0 + \tilde{q} * m_a - 1 \\ \mathbb{E}[\text{Home Bet Return}] = \tilde{p} * m_h + \tilde{q} * 0 - 1 \end{cases} \tag{14}$$

$$\iff \begin{cases} \mathbb{E}[\text{Away Bet Return}] = \tilde{q} * m_a - 1 \\ \mathbb{E}[\text{Home Bet Return}] = \tilde{p} * m_h - 1 \end{cases} \tag{15}$$

As the above equations show, a bettor can only ultimately have a positive expected return, and hence be profitable in the long run, if he is able to bet on games which are mispriced. In other words, the bettor needs to find games such that $\tilde{p} > p_h$ or $\tilde{q} > q_a$ using a model which consistently outputs probabilities ($\hat{p}$ and $\hat{q}$) closer to the true probabilities than are the sportsbook implied probabilities. That is, the bettor must find games with inefficiently high payout multipliers. This is the fundamental challenge of constructing a profitable betting strategy: the market must be inefficient, at least sometimes.

If one trusts that a given model's probabilities, $(\hat{p}, \hat{q})$ are better calibrated than those of the sportsbook, (p_h, q_a), and are unbiased estimators of the true probabilities, $(\tilde{p}, \tilde{q})$, the decision on whether or not to bet should be made based on whether or not the model perceives a positive expected return opportunity:

$$\begin{cases} \hat{p} > p_h, & \text{Bet on home team} \\ \hat{q} > q_a, & \text{Bet on away team} \\ \hat{p} < p_h \text{ and } \hat{q} < q_a, & \text{Do not bet} \end{cases} \tag{16}$$

The bettor is at an additional disadvantage in that the sportsbook will try to calibrate p_f, rather than p_h, to be close to $\tilde{p}$ (and q_f, rather than q_a, to be close to $\tilde{q}$). The margin serves as a buffer to protect the sportsbook in the event that p_f and q_f are slightly mis-calibrated; their probabilities (p_h and q_a) are allowed to exceed 100%. In other words, to find mispricing, the bettor must find games in which the sportsbook has not only underestimated one of the team's chances of victory but has underestimated this probability by at least approximately $margin/2$, or anywhere from 1% to 2.5%. Fortunately, the bettor enjoys the advantage of having numerous opportunities to realize such a situation and is free to not bet until such a situation is realized. MLB currently has 2,430 total regular season games from April to September, equating to 15 games per full day. Later sections explore where to find these opportunities and how to structure daily portfolios of moneyline bets to capitalize as they appear.

B Market Efficiency Literature Review

While there is considerable academic literature on prediction markets in general, the baseball-specific literature is relatively sparse. However, some authors have explored similar themes in the past and their papers provide useful ways to conceptualize certain characteristics of the market. In particular, previous papers by Linda and Bill Woodland and Matthew Bouchard, provide useful perspectives for analyzing the notions of market efficiency relevant to the MLB moneyline market to develop ex-ante expectations of where mispricing may appear.

B.1 Woodland and Woodland: The Favorite-Longshot Bias

In their 1994 paper, Market Efficiency and the Favorite-Longshot Bias: The Baseball Betting Market, Linda and Bill Woodland apply various statistical tests of weak-form market efficiency to the baseball betting market using games from 1979-1989 (24,603 games) (Woodland and Woodland, 1994).

In their most telling statistical test, the authors sort each game by offered line pair (i.e., by (p_h, q_a)) and analyze whether these probabilities match the true frequency of outcomes, an approach philosophically similar to using calibration plots, a tool introduced in Section IV. For a given line, l, with n_l total observations, they use the following test statistic to examine whether or not the null hypothesis $H_0 : p_f = \tilde{p}$ holds. Letting

$p_{o,l}$ represent the true observed frequency of the home team winning in games in which line l is offered, the relevant test statistics are:

$$z_l = \frac{p_{o,l} - p_{f,l}}{\sqrt{\frac{p_{f,l} * q_{f,l}}{n_l}}}, \ \forall l \in \{1, 2, ..., 26\} \tag{17}$$

The authors find that 3 of the 26 lines constructed lead to rejection of the null hypothesis at the 10% confidence level, a result they take to imply a great deal of efficiency, but which seems quite inefficient from my perspective. Another test of efficiency performed on a regression model conducted by Woodland and Woodland, considering all offered lines together, finds a rejection of the null hypothesis of market efficiency at the 10% confidence level. As shown in Section IV, (p_f, q_f) closely (but not perfectly) corresponds with the true observed probabilities.

To conclude the paper, the authors find an interesting degree of deviation from efficiency even when considering nothing but the given prices/lines (a violation of weak-form market efficiency). In particular, they find that always betting on the underdog yields losses significantly lower than randomly betting on games, although this simple strategy does not violate market efficiency significantly enough to generate profits when accounting for margin. In general, it seems fair to say that the baseball betting market is at least somewhat efficient in that naive strategies considered by Woodland and Woodland and in Section VIII within this paper are never profitable, given transaction costs (margin). However, the market appears to have notable pockets of inefficiency, even at the weak-form level, which suggests that constructing a profitable investment strategy is certainly possible.

B.2 Bouchard: Information and Market Efficiency

In his 2019 paper, Matthew Bouchard analyzes the evolution of the MLB moneyline market's efficiency over time using a comprehensive dataset of past sportsbook lines from 1977-2018 (Bouchard, 2019). The paper seeks to answer a few key questions regarding when one might expect markets to have higher or lower degrees of efficiency. Of particular relevance to this paper, Bouchard explores whether efficiency has increased over time and whether listed lines are more efficient later in the season. The paper also explores interesting questions such as whether more drastic line movements from the time the line is originally made public is predictive of the line's prediction error and what effect the onset of the NFL season has on the accuracy of available lines.

With regard to whether sportsbooks offer more accurate lines today relative to the past, Bouchard uses

two main regression frameworks to test whether absolute error in probability estimation ($E_{i,t}$) or standard deviation of this error ($\sigma_{i,t}$) can be predicted based on the year the line was offered and the "line group" (or which of 29 subintervals of $[0, 1]$ in which (p_f, q_f) fell):

$$\begin{cases} E_{i,t} = \beta_0 + \beta_1 Year_t + \sum_{i=1}^{29} \gamma_i I_i + \epsilon_{i,t} \\ \sigma_{i,t} = \beta_0 + \beta_1 Year_t + \sum_{i=1}^{29} \gamma_i I_i + \epsilon_{i,t} \end{cases} \tag{18}$$

where $E_{i,t}$ is calculated as the absolute difference between the true outcome and the fair implied probability. $Year_t$ is an indicator variable for whether the given line was offered in year t and I_i is an indicator variable for whether the given line was in the ith line group.

The paper finds that absolute probability prediction error is lower in later years, but not statistically significantly so, which is consistent with findings shown in Section V (no major reduction in sportsbook loss function scores over time). However, the results of the second regression demonstrate that the standard deviation of absolute errors has declined over time, with the most drastic improvement coming in the late 1990s and early 2000s. In Bouchard's words, the major shift in market dynamics since the late 1970s appears to be characterized by "improvements in prediction precision rather than prediction accuracy"

To assess whether sportsbook predictions improve over the course of the season (presumably as more games reveal more information about true team skill), Bouchard uses the following regression framework:

$$E_{i,t} = \beta_0 + \beta_1 N_{i,t} + \sum_{i=1}^{42} \gamma_t I_t + \epsilon_{i,t} \tag{19}$$

where $N_{i,t}$ equals the total number of games played before the ith week of year t and I_t is an indicator variable for whether the given line was offered in year t.

Interestingly, the results of this regression analysis yield an estimate for β_1 of -0.0000289, which is statistically significant at the 1% level, seemingly implying that sportsbook accuracy improves throughout the season. This finding holds even when additional terms are added to the regression to control for the effects of the start of the NFL season, which is sometimes hypothesized to increase the informedness of the average MLB moneyline bettor (as casual bettors move to the larger NFL betting market). This finding potentially has implications for an MLB moneyline strategy; perhaps one should decrease participation as the season progresses or perhaps this effect is offset by improvements in one's own model over the course of the season.

Another fascinating section of Bouchard's paper explores whether sportsbooks intentionally bias their lines as a marketing tactic to maximize betting volume, a phenomenona which would violate the above assumption that p_f and q_f always represent a book's true beliefs on a given game, but which would likely create additional opportunities for a systematic betting strategy. To assess this possibility, Bouchard introduces another regression framework, this time calculated over a six year period for which both opening and closing lines are available:

$$E_{i,t} = \beta_0 + \beta_1 \Delta_{i,t} + \beta_2 \mu_{i,t} + \beta_3 (\mu_{i,t} * \Delta_{i,t}) + \delta_0 F_{i,t} + \sum_{i=1}^{6} \gamma_t I_t + \epsilon_{i,t} \tag{20}$$

where $\Delta_{i,t}$ equals the magnitude of average percentage change of year t games in the ith line group (here separated by decile), $\mu_{i,t}$ is an indicator variable for whether the line moved towards the team favored to win (i.e., if the favorite team became more favored), and $(\mu_{i,t} * \Delta_{i,t})$ is the interaction variable.

While Bouchard's detailed results and discussion are likely beyond the scope of this paper, he concludes that the start of the NFL season coincides with a larger error in opening lines, potentially suggesting that sportsbooks may intentionally bias opening lines to attract NFL bettors. More strikingly, the paper finds that larger line movements are associated with larger prediction errors. As a result, one might consider using the difference between opening and closing lines on a game as a predictive feature. Overall, Bouchard's work provides an interesting application of traditional market efficiency concepts in the MLB moneyline space with potential implications for game prediction and strategy timing.

B.3 Discussion

While the papers explored above are useful for measuring how efficient a given sportsbook or model is, one can surmise that, based on the simplicity of the pricing problem, perfect efficiency can never be achieved. Given the assumption that $(\tilde{p}, \tilde{q})$ are unknown and unknowable, the best a sportsbook can hope for is for (p_f, q_f) to predict $(\tilde{p}, \tilde{q})$ with minimal deviation cross-sectionally. To the extent that one can produce a model which outputs probabilities with less deviation, it is always theoretically possible to beat the sportsbook. In other words, the upper bound on efficiency is always correctly picking the winner of a game and assigning that outcome a 100% probability ex-ante. But this level of prescience is impossible. Given the "no oracles" assumption that the true probabilities are unknowable, perfect efficiency is never possible; thus, marginally more efficient models are always possible, limited only by the quality of available features and of the models used to find a function from feature space to $(\hat{p}, \hat{q})$.

II Data

A Historical Game Data

All historical game data were downloaded from retrosheet.com. The data accessed include the winning and losing team, each team's starting lineup, starting pitchers, managers, umpires, ballpark and attendance data, and other miscellaneous data points such as whether the game was part of a doubleheader. I decided to download game logs back to 1976 to get a large data set (98,762 games).

B Historical Player Statistics

All historical player statistics were downloaded from baseball-reference.com. Among other data, this site maintains an extensive database of annual batting, pitching, and fielding statistics. Downloaded data includes standard batting and pitching, baserunning, situational statistics, Sabermetrics, and ratio statistics. In constructing the design matrix, a simplifying assumption of ignoring bullpen and replacement position players' statistics was made. Precisely which pinch hitters, pinch runners, and relief pitchers will appear in a given game is not known prior to the game start. However, a distribution of possible replacement players is approximated by including some historical team statistics, rather than individual player statistics. For instance, entries for 2019 New York Yankees games include 2018 New York Yankees team relief pitching statistics as accessible features, rather than the individual pitching statistics of each available reliever.

C Historical Odds Data

Historical odds data from 2009-2019 were obtained from oddswarehouse.com. Odds Warehouse sources its odds data primarily from Pinnacle, a European sportsbook known to have very accurate lines because of a unique business model. Specifically, Pinnacle enforces very low bet limits on opening lines and very high limits on closing lines compared to an average sportsbook in order to more fully incorporate market opinions and is known to cater more towards professional sports bettors than are other sportsbooks. In addition, Pinnacle offers considerably lower margin than the average sportsbook (about 1.9% on average). When odds data from Pinnacle are unavailable, the data are taken either from BookMaker or BetOnline, other established offshore sportsbooks. Odds data from 1977-2006 were sourced from Computer Sports World. Overall, the aggregate odds data correctly predict the winning team 57.6% of the time (calculated by which team has higher fair implied odds). For comparison, the home team wins 53.8% of the time throughout the entire sample, and so, this figure can be loosely regarded as a lower bound for any model or sportsbook prediction accuracy.

III Portfolio Construction

The methods used to obtain $\hat{p}$ and $\hat{q}$ will be discussed in later sections, but it is worth exploring how to best use these probability estimates before proceeding. In Section I.A.3, I explored how to decide whether or not to bet on either team in a game. The next step is prudently selecting portfolio weights using well-calibrated probability estimates, $\hat{p}$ and $\hat{q}$, in order to maximally exploit mispriced games while avoiding severe drawdowns associated with such risky assets. To do this, I use the Kelly criterion.

A Kelly Criterion

In 1956, J.L. Kelly Jr. described the Kelly criterion, which provides a "staking amount" as a percentage of one's total bankroll by maximizing one-period expected log wealth (Kelly, 1956). Remarkably, under the assumption of infinite and identical trials, it has been demonstrated that myopically optimizing over one period in this way is equivalent to maximizing one's expected geometric growth rate over an infinite number of periods. While the assumption of infinite identical wagers is certainly violated in the baseball moneyline market, the Kelly criterion is quite robust to this particular violation and is widely used in gambling, albeit with modifications such as fractional Kelly, which will be subsequently explored.

Without loss of generality, consider a situation in which one's model has determined that a sportsbook has sufficiently underestimated the home team's chances of victory: $\hat{p} > p_h$. Formally, the Kelly criterion selects the betting proportion, x^*, which maximizes one-period expected log wealth:

$$x^* = \underset{x}{\text{argmax}} \quad \hat{p}log[1 + x * (m_h - 1)] + \hat{q}log(1 - x) \quad \text{s.t. } x \in [0, 1] \tag{21}$$

Solving the above optimization problem yields the Kelly criterion betting proportion:

$$x^* = max\left\{\frac{\hat{p} * m_h - 1}{m_h - 1}, 0\right\} \tag{22}$$

Or to more closely resemble Kelly's original notation, letting $b_h = m_h - 1$ be the "net odds", or net multiplier received on betting on the home team:

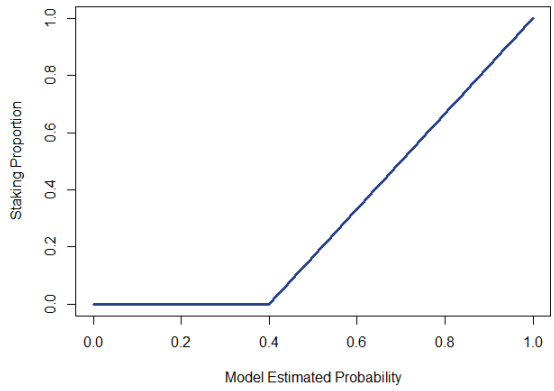
$$x^* = max\left\{\frac{\hat{p} * b_h - \hat{q}}{b_h}, 0\right\} \tag{23}$$

For example, suppose that in the above example, the sportsbook's estimate of the Red Sox's probability of victory, $p_h = .5327$, is too low. The model predicts that the Red Sox actually have a 56% chance of victory: $\hat{p} = .56$. In this case, the Kelly criterion would suggest one bets $x^* = 5.842\%$ of the bankroll on the

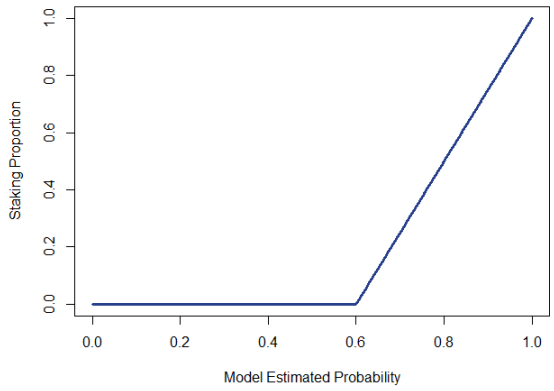
Red Sox. Intuitively, one should conceptualize the Kelly criterion as increasing in $\hat{p}$ such that one's expected growth rate is maximized and decreasing in p_h down to a minimum of zero, i.e., refusing to bet on games in which the bettor does not have a (perceived) positive expected return.

To visualize how the Kelly criterion staking proportion changes as a function of one's "edge," or degree of perceived mispricing, fix p_h, without loss of generality, and plot x^* as a function of $\hat{p}$. Once $\hat{p} > p_h$, the staking proportion is a linear function of $\hat{p}$. More intuitively, the Kelly optimal staking proportion always represents the proportion of the distance from p_h to $\hat{p}$ to the distance from p_h to 1:

$$x^*(\hat{p}, p_h) = max\left\{\frac{\hat{p} - p_h}{1 - p_h}, 0\right\} \tag{24}$$



(a) $x^*(\hat{p})$ when $p_h = 0.4$



(b) $x^*(\hat{p})$ when $p_h = 0.6$

B Kelly Criterion for Multiple Simultaneous Bets

The Kelly criterion above can be analogously extended to the multivariate case. Again without loss of generality, consider a situation in which one's model has underestimated the home team' probabilities of victory in 2 simultaneous and independent games (if the away teams' chances are underestimated, just change the sign of the relevant x terms and switch where the b coefficients are). Letting $\vec{x^*} = [x_1^*, x_2^*]^T$ and still maximizing one-period expected log wealth gives the following program:

$$\vec{x^*} = \underset{x_1, x_2}{\text{argmax}}\ \hat{p}_1\hat{p}_2 log(1 + b_1x_1 + b_2x_2) + \hat{p}_1\hat{q}_2 log(1 + b_1x_1 - x_2) + \hat{q}_1\hat{p}_2 log(1 - x_1 + b_2x_2) + \hat{q}_1\hat{q}_2 log(1 - x_1 - x_2)$$

$$\text{s.t. } x_i \in [0, 1],\ \forall i \in \{1, 2\},\ \sum_{i=1}^{k} x_i \leq 1$$

where b_1 and b_2 are analogous to b_h above, for both of the bets. One can extend this framework to k games, but even writing the objective function becomes cumbersome for $k > 2$ (it will have 2^k terms, each representing one possible combination of outcomes). In addition, finding the first-order conditions (and hence a closed-form solution for $\vec{x^*}$) in the multivariate case is quite laborious. Luckily, solving the problem in the multivariate case numerically is very easy using Excel solver, for instance. The difference between computing $\vec{x^*}$ analytically or numerically is minuscule and practically irrelevant.

Interestingly, when a sportsbook simulatenously underestimates multiple teams' chances of victory, any parlay (combination bet) offered by the sportsbook will also be underpriced if the sportsbook's pricing is consistent. Of course, if this is the case, the Kelly criterion would also suggest betting some positive amount on this parlay. In the above case of two games ($\hat{p_1} > p_{h1}$ and $\hat{p_2} > p_{h2}$) implies $\hat{p_1} * \hat{p_2} = \hat{p_{1,2}} > p_{h1,h2} = p_{h1} * p_{h2}$. In other words, to take into account available parlays, one would want to include $b_{1,2}x_{1,2}$ in the first term of the above objective function and $-x_{1,2}$ in all other terms of the objective function, as this bet only pays off if both teams win. Of course, considering parlay bets further complicates the above objective function, particularly when the number of games is high. Since the optimal staking size for these bets will be low and decreasing in the number of games, one possible constraint is limiting the "parlay order," or number of games included in any parlay, to a low number (perhaps 2 or 3). For the remainder of this paper, I will ignore parlays as they are not explicitly offered in the odds data and simply assuming $p_{h1,h2} = p_{h1} * p_{h2}$ (i.e., consistent pricing) may not reflect reality. In addition, some sportsbooks do not offer parlays. In practice, maximally exploiting mispriced games will require consideration of parlay bets.

C Adjustments for Probability Estimation Error

A crucial assumption of the Kelly criterion which is more consequentially violated in reality is that the bettor can perfectly estimate $\tilde{p}$. In reality, the true probabilities are unknowable, and even with a perfectly unbiased model, $\hat{p}$ is subject to estimation error. Unfortunately, the Kelly criterion is less robust to violations of this assumption. Thus, it is critical to ensure one's probability estimate is highly precise, or to modify the staking amount in response to the model's imprecision. Several probability estimation error risks arise even if the game is truly mispriced that can adversely affect one's expected growth rate:

Error	Consequence
$\hat{p} < p_h < \tilde{p}$	Missed opportunity to take a positive expected return bet
$p_h < \hat{p} < \tilde{p}$	Staking proportion was too low
$p_h < \tilde{p} < \hat{p}$	Staking proportion was too high

Table 3: Probability estimation error risks and consequences

C.1 Discretionary Approaches

Intuitively, the third error is worse than the second error. Assuming symmetric probability estimation error, betting too much on a single game (itself an asset with very high variance of returns) is more detrimental than betting too little on a single game. Since the staking amount given by the Kelly Criterion gives the growth optimal portfolio weight assuming a perfect probability estimate, when a bettor's probability estimate is too high, the bettor will be taking a positive expected return bet but with a lower expected growth rate because of the higher variance. Conversely, when the bettor has underestimated $\tilde{p}$ by an equal amount, his expected growth rate is decreased by a roughly equal amount owing to decreased expected return, as variance has actually decreased in this case. Most popular modifications to the Kelly criterion heed this intuition and take a more conservative approach than "raw Kelly."

Three popular modifications include "thresholding," "fractional Kelly," and establishing a maximum stake per game.

Thresholding modifies the base Kelly criterion by using the base criterion only if $\hat{p}$ exceeds p_h by a predetermined threshold. Intuitively, if probability estimation error is a consideration, cases in which $\hat{p}$ is only slightly larger than p_h would yield positive x^* values under the Kelly criterion, but run a higher risk of truly being negative expected return bets compared to cases in which $\hat{p} >> p_h$, assuming roughly equal estimation errors for each game.

The second approach, "fractional Kelly," seeks to immunize the bettor against surpassing the growth optimal portfolio weight. Rather than seeking to decrease the probability of taking negative return bets, fractional Kelly simply multiplies the base Kelly criterion staking percentage by a constant between 0 and 1. As a result, the bettor takes the same bets, but bets a smaller percentage on each game. In this sense, fractional Kelly reduces the risk of sub-optimally high variance by decreasing the average expected growth rate of all bets, assuming $\hat{p}$ is an unbiased estimator of $\tilde{p}$.

Lastly, establishing a maximum staking size serves to protect the bettor against the severe consequences of having a large, incorrect bet. As will become clear in the backtesting section, volatility reduction is key to achieving a consistently profitable strategy. As a result, not allowing a model to "put it all on black" is a very sensible modification, particularly when probability estimation error is high.

Various combinations of the above approaches will be utilized in Section V. To summarize:

Kelly Modification Parameters	
Thresholding Parameter (a)	Only bet if $(\hat{p} - a) > p_h$
Fractional Parameter (f)	$x^*_{frac} = fx^*,\ f \in (0, 1)$
Maximum Parameter (m)	$x^*_{max} = min\{m, x^*\}$

Table 4: Summary of Kelly Modification Parameters

I combine these three modifications for use together in the following way:

$$x^*_{mod} = \begin{cases} min\{m, fx^*\},\ (\hat{p} - a) > p_h \\ \\ 0, \text{ otherwise} \end{cases} \tag{25}$$

C.2 Robust Optimization

Admittedly, the modifications presented above are somewhat unscientific. While they provide useful and easily digestible safeguards against violations of the Kelly criterion's assumptions, it is not clear that these methods do so optimally. A more comprehensive approach involves a full characterization of the distribution of $\hat{p}(\vec{x_i})$, where $\vec{x_i}$ is the relevant row in the design matrix.

John Mulvey and Robert Vanderbei's paper *Robust Optimization of Large-Scale Systems* provides a framework which can be applied to maximizing one-period expected log-wealth but with a randomly perturbed $\hat{p}(x)$ (Mulvey and Vanderbei, 1995). Applying the Kelly criterion with a random objective function in this way can ensure that x^* is "model robust."

Intuitively, Robust Optimization simply extends any linear optimization program over a set of S scenarios each with different values for the control variables (probabilities of winning, for the purposes of this paper), $\Omega = \{1, 2, 3, ...S\}$, each occurring with probability p_s. The technique then minimizes some aggregate objective function, $\sigma(\hat{p}_1, ...\hat{p}_S)$ of the probabilities over the S scenarios. In the simplest formulation of this technique, one can define $\sigma(*)$ equal to the weighted average (negative, since I am now minimizing) Kelly criterion staking amount over the scenario set, which can be shown to be equivalent to taking a weighted-average of multiple model outputted probabilities (weighted by $[p_1, ..., p_s]^T$) first and then applying the Kelly criterion. This equivalence provides further theoretical support for the use of ensemble approaches (explored in Sections V and VI).

Aside from simply averaging the probabilities outputted by multiple predictive models, other potential applications of Robust Optimization to this problem include imposing an assumed error distribution around a single probability estimate and integrating over this distribution of values for the "true probability" to determine a continuous scenario set or using more risk-averse aggregate objective functions, such as mean-variance utility or some other custom utility function.

Because the joint distribution governing the relationship between $\hat{p}$ and x is quite complex for more complicated predictive models, and for simplicity, I have decided to apply the more discretionary, "post-optimization" modifications described above for the purposes of this paper. However, this framework provides a potentially fruitful direction for future exploration.

D Building Daily Portfolios

Once one has decided upon a staking algorithm (either raw Kelly criterion or some variation), he can construct $\vec{x^*}$ for each day in the testing set and plot the path of the bettor's wealth over time. Given an available betting universe of k games on day t (which may vary but is on the order of 10-15) and a starting wealth of W_0, the bettor's wealth is a random process which evolves as follows:

$$W_t = W_{t-1}(1 - \sum_{i=1}^{k} x_i^* + \sum_{i=1}^{k} \mathbb{1}_{\{y_i = g_i\}} m_i x_i^*), \ \forall t \in \{1, 2, ...\} \tag{26}$$

where y_i indicates whether the home or away team won and g_i indicates whether the bet was taken on the home or away team:

$$g_i = \begin{cases} 1, \text{ bet on home team} \\ 0, \text{ bet on away team} \end{cases} \tag{27}$$

$$y_i = \begin{cases} 1, \text{ home team won} \\ 0, \text{ away team won} \end{cases} \tag{28}$$

and m_i is the payoff multiplier associated with the correct side of the bet,

$$m_i = \begin{cases} m_{h,i}, \text{ bet on home team} \\ m_{a,i}, \text{ bet on away team} \end{cases} \tag{29}$$

Intuitively, the bettor ends the prior day with wealth W_{t-1} (the first term), places his bets for the day (he must pay the second term to do so), and finally realizes returns on none, some, or all of his bets (the third term). Having a positive return on a given day requires the third term to exceed the second term in absolute value. This can occur either by correctly predicting many outcomes (having the indicator set to 1 often), correctly predicting some underdog winners (having large m_i for games correctly predicted), or a combination of both. In addition, the importance of the i^{th} game is weighted by x_i^*, clearly illustrating the importance of implementing a sensible staking strategy. Rearranging the wealth evolution process allows one to calculate daily net return:

$$R_t = \frac{W_t}{W_{t-1}} - 1 = -\sum_{i=1}^{k} x_i^* + \sum_{i=1}^{k} \mathbb{1}_{\{y_i = g_i\}} m_i x_i^*, \ \forall t \in \{1, 2, ...\} \tag{30}$$

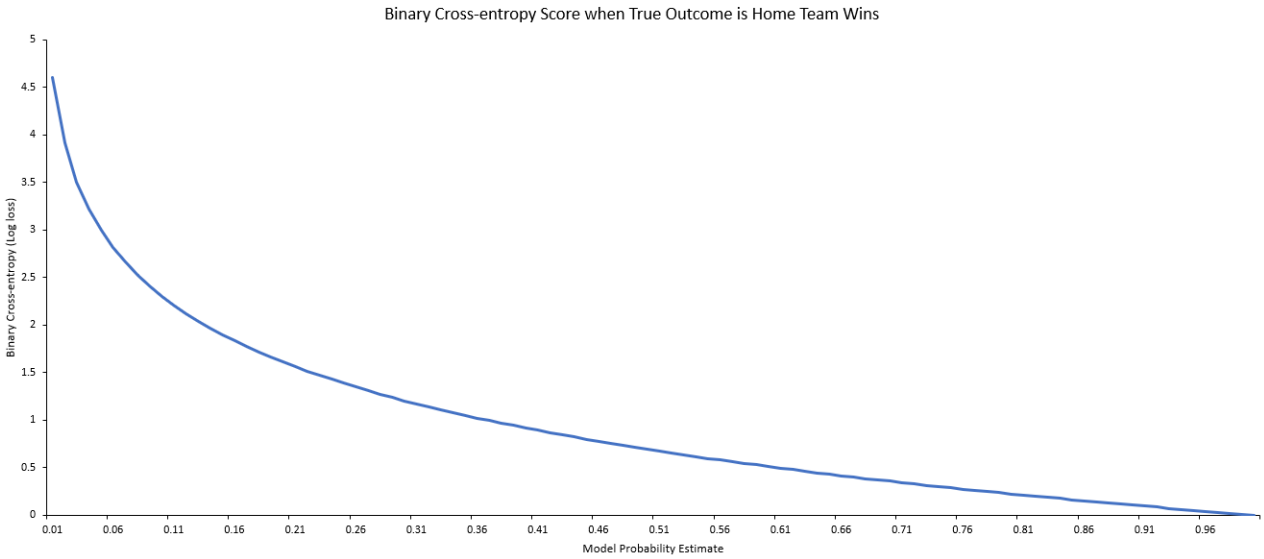
IV Model Calibration

I now move to actually obtaining the probability estimates needed as inputs to the Kelly criterion and its variants. Throughout the remainder of the paper, I use two main criteria to assess how well a model predicts the outcome of games: binary cross-entropy and calibration plots. Binary cross-entropy gives a measure of a probability estimate's "distance" to the true outcome. For example, if the home team wins ($y_i = 1$), a probability estimate ($\hat{p}$) of 0.8 results in a lower (better) score than a probability estimate of 0.7, which in turn earns a much better score than a probability estimate of 0.4. When averaged across many games, binary cross-entropy allows one to assess the average proximity from $\hat{p}$ (or $\hat{q}$) and the true outcome (y_i). Practically speaking, any strictly proper scoring rule could provide a similar gauge of estimate proximity (such as mean-squared error or quadratic loss). Selecting one particular scoring rule over another corresponds to defining a particular loss distribution for errant probability estimates. However, the basic idea remains the same for all reasonable loss functions: I seek to minimize some measure of distance between estimate and realization on average; I want to incentivize well-calibrated probability estimates. Binary cross-entropy, the traditional loss function used in binary classification, is defined as follows:

$$H(\vec{y}, \vec{\hat{p}}, \vec{\hat{q}}) = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(\hat{p}_i(\vec{x_i})) + (1 - y_i)\log(\hat{q}_i(\vec{x_i})) \tag{31}$$

Using a measure of distance to gauge model performance is crucial; optimizing on raw accuracy, for instance, may lead to estimates accumulating around 0 and 1, or other pathological outcomes. In fact, a high degree of accuracy may not even be necessary in an excellent model. As the Kelly criterion makes clear, one might find positive expected return bets in betting on the supposed underdog team (if $\tilde{p} > 0.5 > \hat{p} > p_h$, for instance). In this scenario, the model's belief that the away team is the probable victor is actually incorrect but by having a probability estimate closer to reality than the sportsbook's estimate, the bettor can still take a positive expected return bet. In this sense, accuracy as a performance metric is too simplistic to evaluate the quality of estimates in an intelligent way; all it "knows" is whether the estimate is on the correct side of 50%. While creating a moneyline betting model is a binary classification problem at its core, the actual binary classification (the last step) is unimportant relative to obtaining realistic probability estimates since the goal is not simply to determine which team is more likely to win, but rather whether either team, whether the favorite or the underdog, is being underestimated.

To visualize binary cross-entropy, consider (without loss of generality) the case in which one knows ex-post that the home team won game i ($y_i = 1$). In this case, the binary cross-entropy score for game i is simply $-\log(\hat{p}_i(\vec{x_i}))$. The loss becomes more-than-proportionally severe as $\hat{p}$ moves further from the true outcome. The graph is exactly reversed when the away team wins:



A probability calibration plot complements binary cross-entropy by graphically assessing distance between $\hat{p}$ (the expected or "e" frequency) and the frequency of true outcomes (the observed or "o" frequency) in the corresponding region of the histogram. For this paper, I use the R package *givitiR* to create calibration plots. This package separates the "e" axis into 200 subintervals of equal length:

$$[0, 0.005], (0.005, 0.01], ..., (0.995, 1]$$

and plots the actual observed frequency of events "o" within each subinterval.

Better calibration corresponds to a calibration curve closer to the 45-degree line. A calibration curve under the bisector in a specific region corresponds to the model overestimating the likelihood of events in that region. Conversely, a calibration curve over the bisector in a specific region corresponds to the model underestimating the likelihood of events in that subinterval. In general, having regions under the bisector is much more problematic than having regions above the bisector for the same reasons itemized in Section III.C.1. Namely, being below the bisector corresponds to overestimating the probability of events, which will prompt the Kelly criterion to take suboptimally large bets. While areas above the bisector are also problematic, they lead to Kelly taking suboptimally small bets, reducing expected return within this subinterval of $[0, 1]$ but also reducing the overall importance of this region of relatively higher binary cross-entropy scores, a sort of self-correction mechanism not present when one's estimates are below the bisector. Note that the p-value quoted alongside each calibration plot is that of the Hosmer-Lemeshow statistical test, a test of goodness-of-fit for logistic regression models. For well-calibrated models, one may fail to reject the null hypothesis of this test, but it is not an absolutely necessary prerequisite for consideration. A model that is well-calibrated overall may still have deviations from the 45-degree line which are globally

inconsequentially, but locally noteworthy; this type of calibration plot will lead to extremely low p-values but should not lead to ruling out the model.

A The Design Matrix

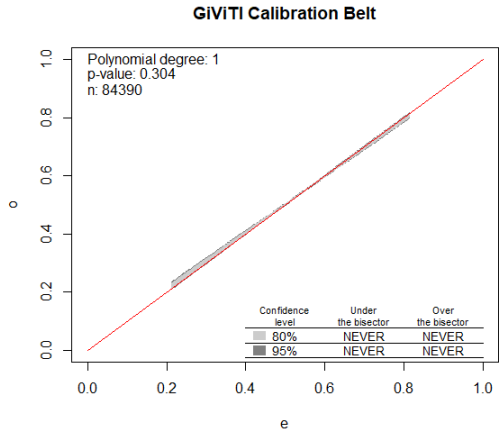
The design matrix used in the models presented is constructed mostly from the aforementioned historical player data acquired from baseball-reference.com. However, some features are taken from the retrosheet game data itself (manager and umpire identifiers, stadium characteristics, etc.) and some features come from other sources (historical weather data). For any given game, historical player statistics are used from the previous season, along with some cumulative career totals. In all, the design matrix contains 98,762 entries, each corresponding to a regular season MLB game from 1976-2019. The entries are randomly divided into an 80/20 training and testing split. There are a total of 676 features per game prior to any feature mapping or elimination of redundant or obfuscatory features. All results shown in Sections IV and V use this design matrix. The real world results presented in Section X uses both this design matrix and also a polynomial transformation of a subset of these features designed to more effectively capture certain feature interactions (2346 features in total after the transformation).

The feature set is primarily composed of single-season level player statistics from the prior season. For example, a game in 2018 will be predicted largely by batting and pitching statistics for each player in the starting lineup and the starting pitcher from 2017. To fill missing data points, I make a simplifying assumption that players missing data are generally worse than average players. For each position in the lineup over each year, I calculate the distribution of each rate statistic and assign missing players the mean less one standard deviation. It is crucial to calculate a distribution for each player in the lineup because players lower in the batting order typically are less skilled batters. While this is an imperfect solution, it appeared to at least be more consistent than extrapolating from minor league statistics or otherwise more finely differentiating between players, although it could be interesting to see whether more creative solutions for filling missing data could improve data quality (and model performance). Perhaps a clustering algorithm could be applied to define a similarity measure between players and missing data could be filled with that of the most similar player(s).

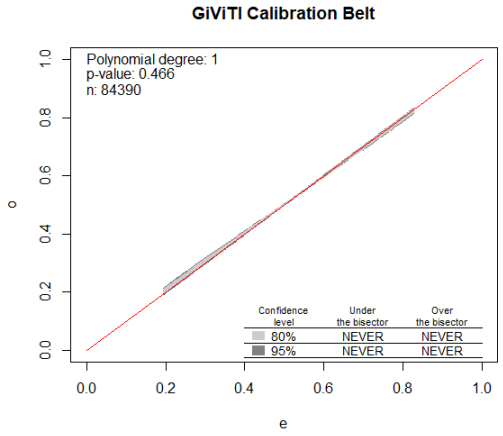
B Historical Odds Data Calibration

To assess the calibration of the subsequent models presented, I first assess that of the implied odds using the aggregate level odds data to establish a baseline calibration level needed for profitability. Below are the calibration plots for the opening and closing fair implied probabilities respectively, calculated on all games

with available odds data:



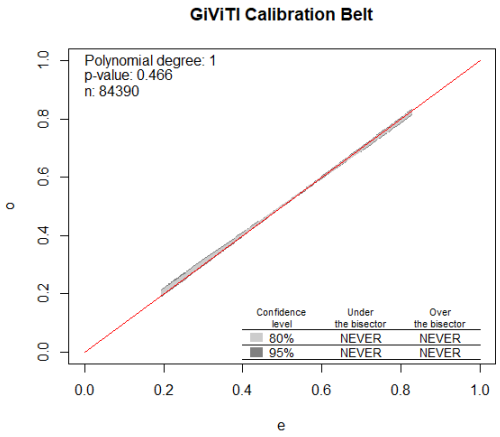
(a) Opening Line Fair Odds: $H(p_f, q_f) = 0.6777$



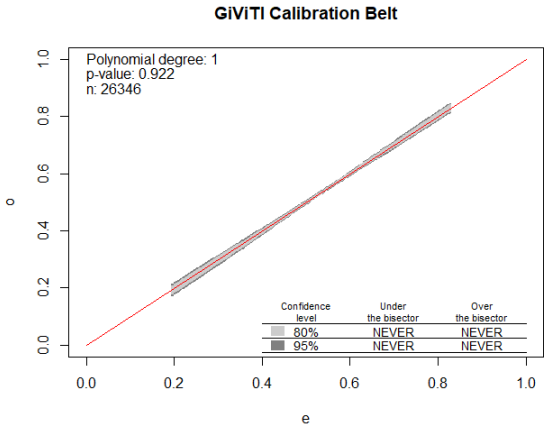
(b) Closing Line Fair Odds: $H(p_f, q_f) = 0.6772$

In general, the sportsbooks' probability estimates are well-calibrated. The historical odds do not tend to overestimate or underestimate in any specific regions and the binary cross-entropy score is quite low. I also observe that the closing line is slightly more efficient (has a slightly lower binary cross-entropy score). While these plots and scores provide an assessment of the accuracy of sportsbooks' fair odds calibration, they do not fully describe the level of accuracy required to be more accurate on average. Because of the effects of margin described earlier, to be more accurate on average when allowing the sum of sportsbooks' probability estimates to exceed 100%, the relevant metric is binary cross-entropy calculated by using the (unfair) implied win probability corresponding to the correct outcome. In other words, in calculating binary cross-entropy, one should use p_h and q_a, rather than p_f and q_f. Making this adjustment yields opening and closing binary cross-entropy scores of 0.6575 and 0.6583, respectively. One can think of this roughly 0.02 decrease in binary cross-entropy as the additional accuracy required on average because of margin.

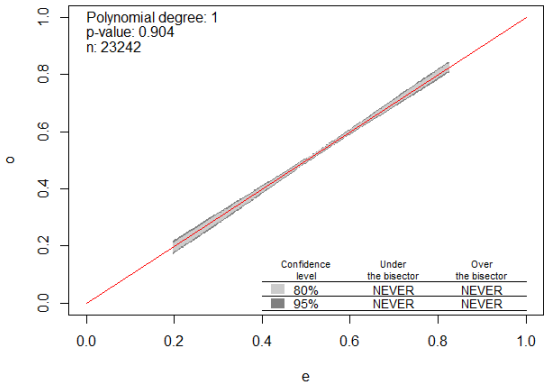
Analyzing the closing-line calibrations of the sportsbooks individually, when such data is available (2009-2019), gives similar results:



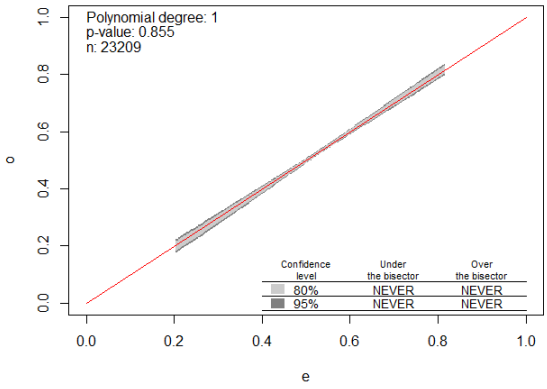
(a) Aggregate Fair Odds: $H(p_f, q_f) = 0.6744$



(b) Pinnacle Fair Odds: $H(p_f, q_f) = 0.6745$



(c) BookMaker Fair Odds: $H(p_f, q_f) = 0.6744$



(d) BetOnline Fair Odds: $H(p_f, q_f) = 0.6745$

The cross-entropy scores and calibration plots of the fair closing lines are almost identical between the three sportsbooks. BookMaker and BetOnline charge somewhat higher margin compared to Pinnacle (2.98% and 2.10% respectively, compared to 1.92%). Thus, calculating their cross-entropy scores with the actual (unfair) implied odds yields greater score improvement, not because of more accurate implied beliefs, but rather, because of higher transaction costs. Using (p_h, q_a) rather than (p_f, q_f) for each calculation gives binary cross-entropy scores of 0.6557, 0.6451, and 0.6537 for Pinnacle, BookMaker, and BetOnline, respectively.

C Formalizing the Objective of Predictive Models

I assume a function $f : \mathbb{R}^k \to [0, 1]$ from the feature space to $\tilde{p}$ exists and I seek to approximate f by analyzing the relationship between y_{train} and X_{train}. That is, I build a function $\hat{p}_{train} = \hat{f}(X_{train})$, evaluating the quality of prospective functions not simply by binary accuracy, but by the quality of calibration over all of

$[0, 1]$. I then calculate $\hat{p}_{test} = \hat{f}(X_{test})$, $\hat{q}_{test} = 1 - \hat{p}_{test}$ and compute binary cross-entropy (and construct the calibration plot) between $(\hat{p}_{test}, \hat{q}_{test})$ and y_{test}. These test set results are presented for selected models in Section V.

D Baseline Model

Here I present the equivalent binary cross-entropy score and calibration plot for a basic logistic regression model. Unfortunately, logistic regression has a number of theoretical issues here; most notably, many of the features in the dataset are highly correlated. In addition, nonlinear and interaction effects between the features may be useful in estimating a team's chances of victory, both of which are ignored in this model. The model performs considerably more poorly than even the fair implied odds calibration above, but provides a useful minimum threshold against which to compare subsequent models.

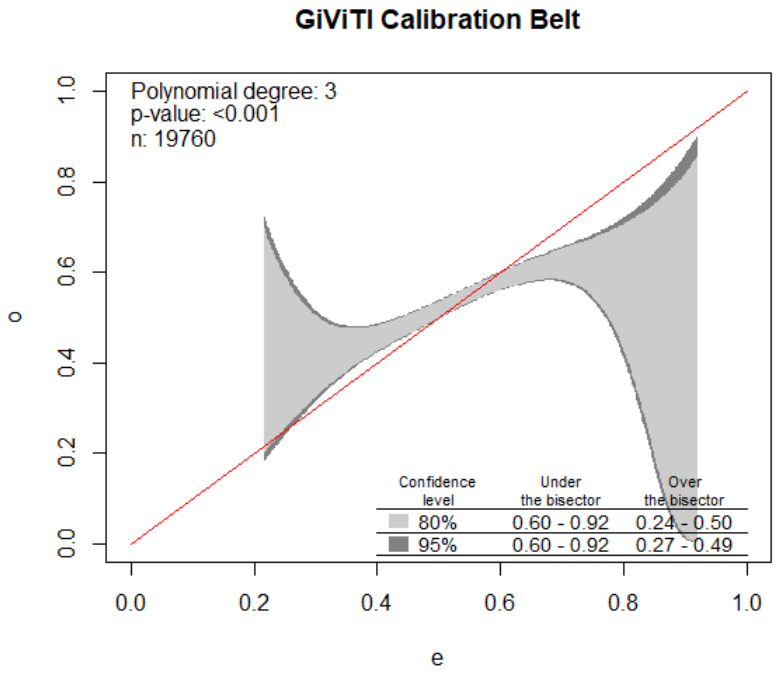


Figure 4: Logistic Regression: $H(\hat{p}, \hat{q}) = 0.6889$

The binary cross-entropy score is considerably higher than that of the odds data and that the model is generally poorly calibrated: it actually underestimates the probability of relatively unlikely events and overestimates the probability of relatively likely events! In addition, the range of empirically observed

probabilities for a given probability estimate output from this learner is extremely wide, particularly for large $\hat{p}$.

E Backtesting the Baseline Model

Of course, simply achieving a lower average binary cross-entropy score than the calibrated odds data is not equivalent to constructing a profitable betting strategy. Instead, because one can choose whether or not to bet on any particular game, achieving a lower overall score is not even strictly necessary. Rather, because any strategy will only bet on some subset of the games available and staking amounts will vary in proportion to perceived mispricing, a profitable betting strategy only truly requires a lower weighted binary cross-entropy score, approximately weighted by dollars wagered, on a subset of games that the bettor is free to choose. To more fairly assess profitability, the following procedure will be implemented.

I define a betting strategy as the combination of a predictive model (logistic regression, neural network, etc.) and a staking algorithm (Kelly criterion, fractional Kelly, etc.) To test the profitability of a betting strategy, the predictions on the test set are fed into the staking algorithm, giving staking amounts for each game in the test set. I then produce a graph depicting the wealth path over all games in the test set, calculate the average return and standard deviation of returns over 15 game intervals (the number of games available for betting on a normal day). In addition, I note the percentage of total games on which a bet is placed to analyze the model's perception of the frequency of mispriced games. I will assume, for simplicity, that the bettor starts with $1000. For example, here are these results for the baseline logistic regression model using the Kelly criterion staking method without modification:

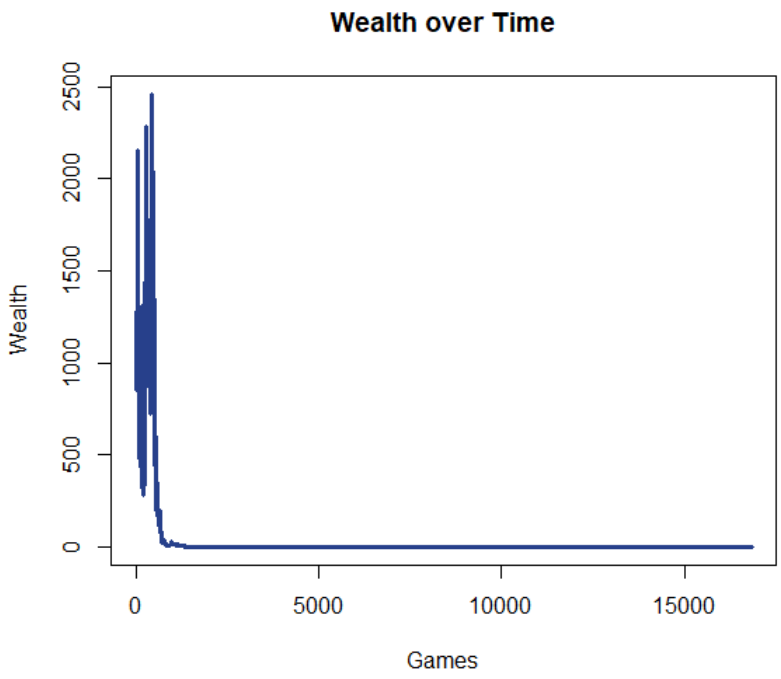


Figure 5: Baseline Logistic Regression, Raw Kelly Wealth Path

The daily net returns of this strategy are positive in expectation, but are also extremely volatile:

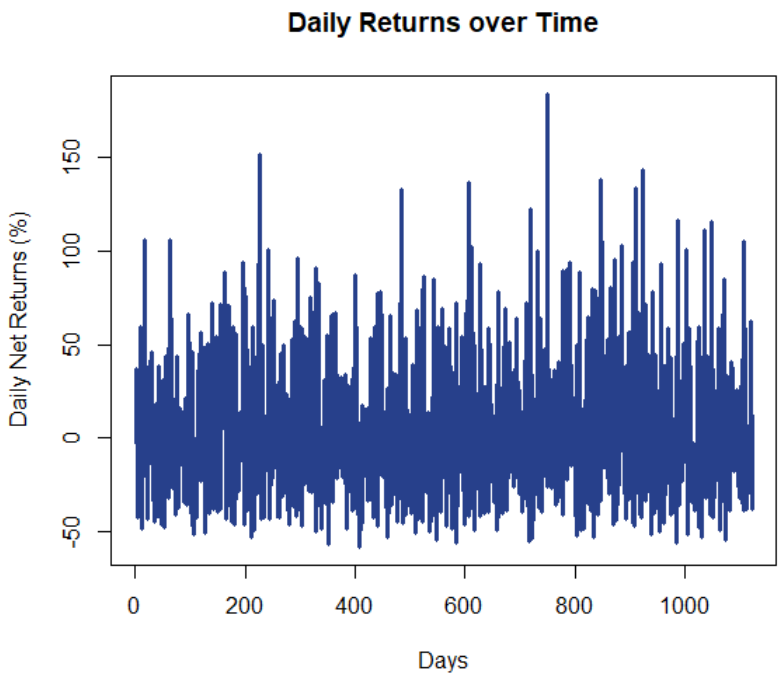


Figure 6: Logistic Regression, Raw Kelly: Daily Returns

After a short period of great performance, the bettor quickly goes effectively bankrupt. Note that actually getting to zero wealth is impossible when using the Kelly criterion as the bettor is only ever betting a fraction of his wealth (assuming arbitrarily small bet sizes are allowed and $\hat{p} \neq 1, \hat{q} \neq 1$). A few extreme results are apparent. Firstly, because the model is so poorly calibrated, it vehemently disagrees with the implied odds (which are at least close to $(\tilde{p}, \tilde{q})$) nearly all the time. As a result, the Kelly criterion's response to the outputted $\hat{p}$ vector is to bet (often very large amounts) on nearly all available games. Betting on many games is not itself necessarily disastrous, but confidently betting the majority of one's portfolio multiple times per week is not a recipe for success. In addition, even over short stretches, using a nominally conservative approach (the Kelly criterion) can lead to wild swings in wealth if $\hat{p}$ is poorly calibrated. This strategy astoundingly gives an average (arithmetic) daily return of 1.55%, but with extreme volatility. To put this volatility in perspective, the wealth path over the first 105 games (roughly the first week) is shown below:

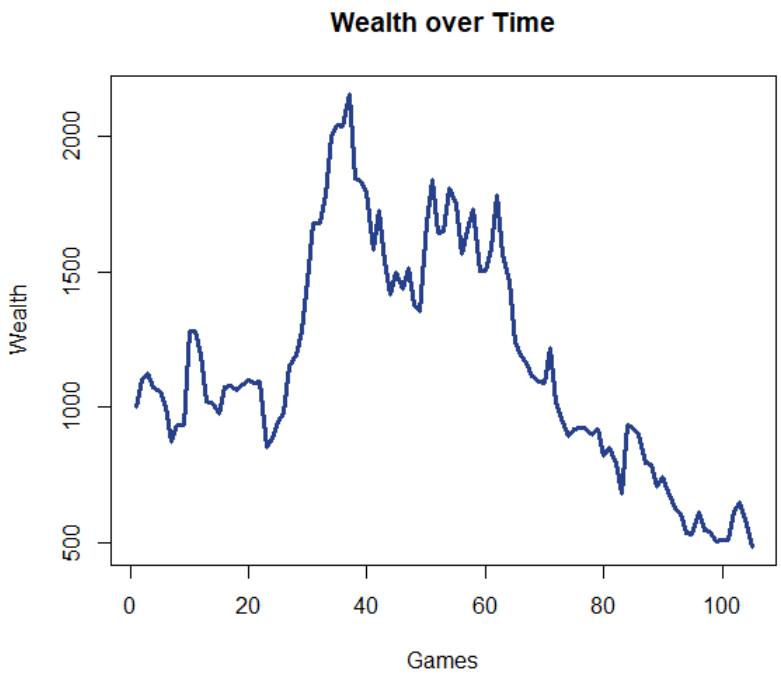


Figure 7: Logistic Regression, Raw Kelly: First Week's Wealth Path

Within a week, the bettor went from his starting $1000 up to over $2000 at his peak, to below $500, finally ending at $484.76. Clearly this degree of volatility is unacceptably high and will need to be tempered in any useful betting strategy. Responding to the highly deviant probability estimates outputted by this model, the Kelly criterion perceives an astonishing degree of mispricing and takes very large bets in response. In fact, the Kelly criterion suggests betting on over 90% of all games and more than 10% of the bettor's bankroll on 24.9% of all games!

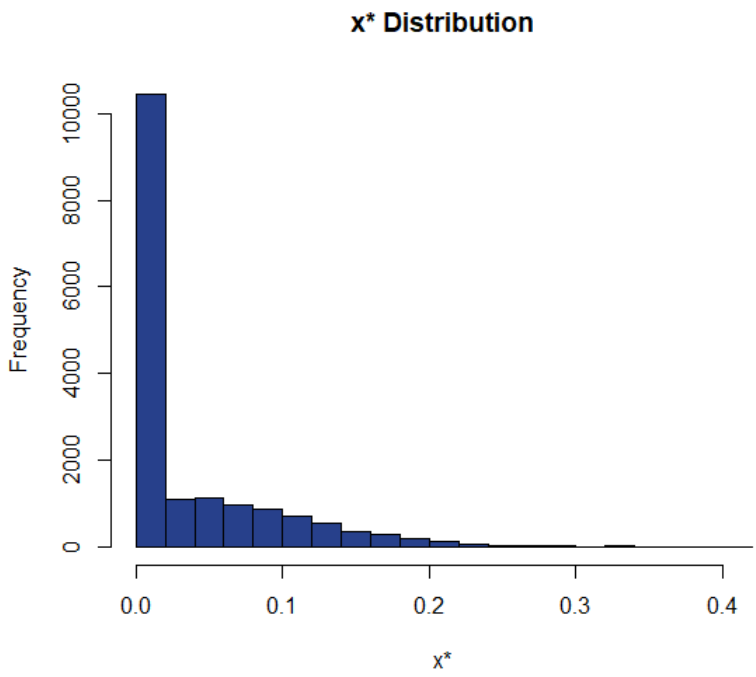


Figure 8: Logistic Regression, Raw Kelly: x^* Distribution

It is somewhat surprising that a naive approach such as this can yield a strategy with positive daily expected returns. However, the extremely high volatility of returns essentially dooms this strategy and all but guarantees bankruptcy in the long run, even when modifying using an extremely low fractional Kelly multiplier. The ratio of volatility to average return is simply too great to overcome. In addition, the 1.55% average arithmetic return figure is highly misleading even before considering the destructive effect of volatility drag. Taking a look at the histogram of returns, it is clear that the distribution is extremely positively skewed and 57.2% of all daily returns are negative. In fact, on 25% of days, the bettor has losses of 25% or more!

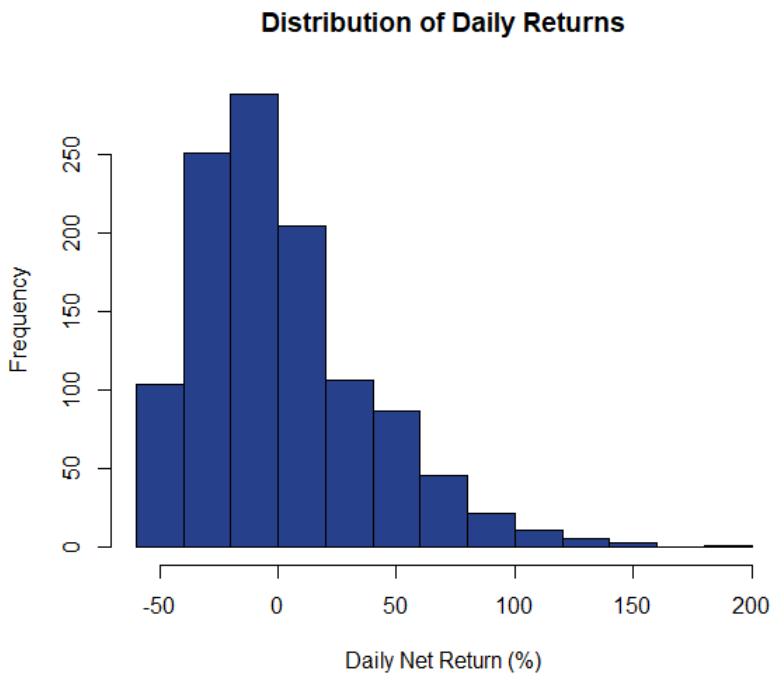


Figure 9: Logistic Regression, Raw Kelly: Distribution of Daily Returns

V Building Better Models

I now shift focus to addressing the shortcomings of the baseline betting strategy (logistic regression with raw Kelly) by achieving lower binary cross-entropy scores and by regulating the Kelly criterion's worst impulses. In this section, I explore how some additional pre-processing steps assist in better understanding the dataset and improving performance. All models presented shown below are better performers than the baseline model. As mentioned earlier, the historical odds data have an accuracy of 57.6%. As shown in Section VII, past research has been able to achieve out-of-sample binary accuracy just over 60%.

A Pre-Processing & Feature Importance

A.1 Principal Components Analysis

Here I explore a classic pre-processing step, principal component analysis (PCA), in order to determine whether a defined factor structure exists and whether further dimensionality reduction would be useful. Interestingly, while the first few principal components account for noticeably more total variance than the rest, the data did not appear to exhibit much of a useful factor structure. In fact, the cumulative variance explained by the first j principal components was larger than j/n generally, but not by as much as one might expect. In other words, the plot of explained variance as a function of number of principal components included is nearly linear:

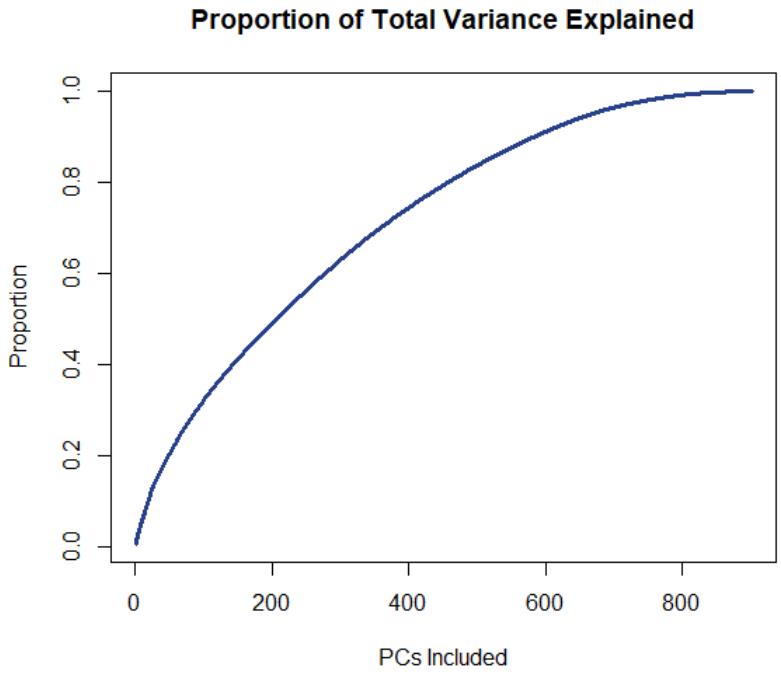


Figure 10: Cumulative Percentage of Total Variance Explained by PCs

Interestingly, the four PCs do provide a much more than proportional explanation of variance within the data:

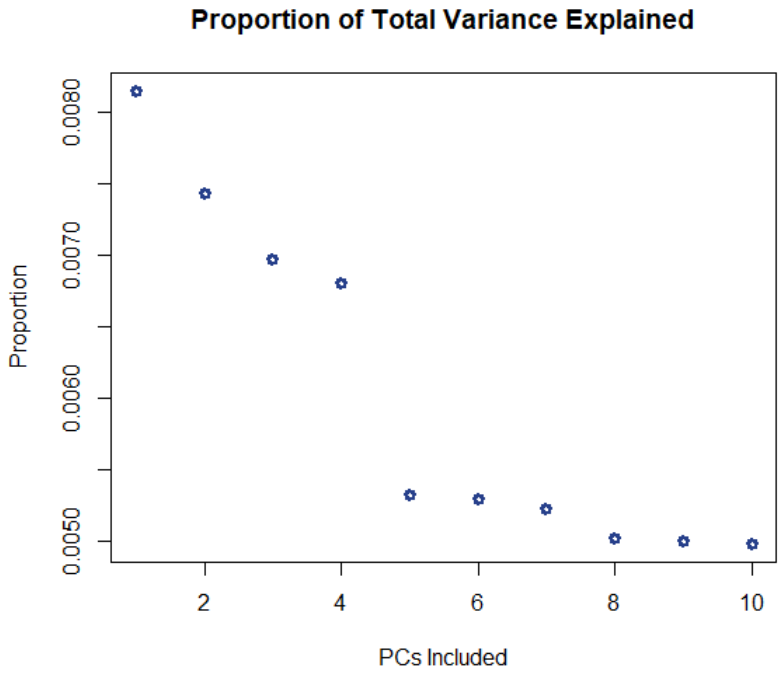


Figure 11: Proportion of Total Variance Explained by first 10 PCs

Taking a deeper look a the composition of these first four PCs, I find that the result is not incredibly interesting. The top 15 variables in terms of contribution to the first PC are all regularization statistics one would expect to vary together (AIR). Specifically AIR for a given player in a given year seeks to provide a normalization coefficient for a player's performance based on league and park factors. Because league and park factors are very heavily correlated between players in the same lineup in the same game (because they are on the same team, and hence play in the same parks and league), one would expect this variable to explain a great deal of variance, but not necessarily to be predictive. A similar phenomenon is observed for the second PC. Once I move to the third PC, a more diverse and potentially predictive set of variables emerges. The largest values in the third PC include Runs Created, RBIs, Wins above Replacement, and Total Bases, among others.

Overall, because the cumulative variance explained increases so slowly in the number of PCs, PCA did not appear to be a particularly fruitful way to re-organize the data or reduce dimensionality. For completeness, a logistic regression was run using the first 100 PCs obtained above as the predictors. However, this led to some uninspiring results at the probability estimation step:

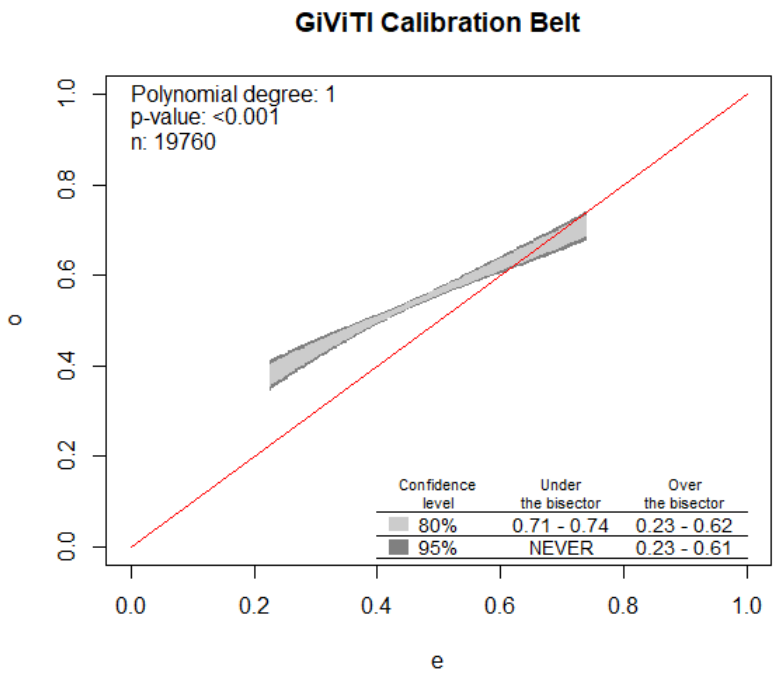


Figure 12: PCA-based Log Reg Calibration: $H(\hat{p}, \hat{q}) = 0.7004$

A.2 xgb Feature Importance

Later in this section, among other learners tested, xgboost will be analyzed. In addition to providing relatively well-calibrated probability estimated compared to other learners, xgboost also outputs a feature importance vector. The top six features in terms of "gain" are the Starting Pitchers' strikeout rates and RE24s (a sabermetric which measures run expectancy), and FIP rates (a variation on ERA which seeks to isolate pitcher performance by focusing only events for which the pitcher is fully responsible and ignoring plays on which the ball is put in play).

B Standardized Staking Strategies

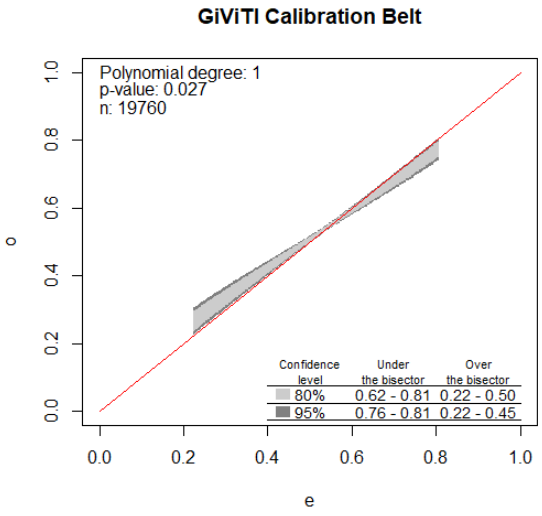
To standardize return distribution results for comparison, I use two staking strategies: the raw Kelly criterion and a more selective strategy. These strategies have the following modification parameters:

	a	f	m
Raw Kelly	0.00	1.00	1.00
Selective	0.10	0.70	0.20

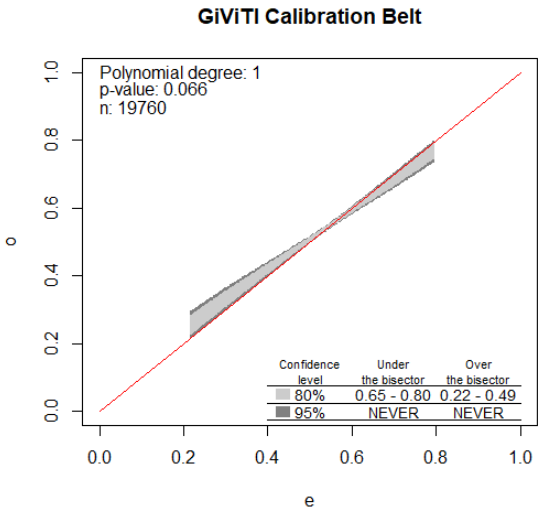
Table 5: Definitions of Staking Strategies

C Penalized Regression Models

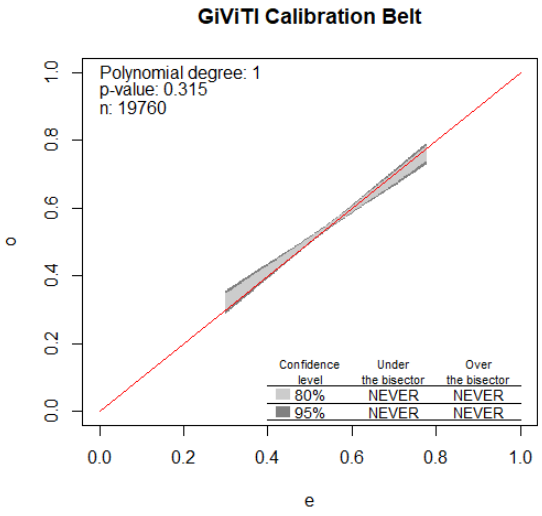
To address the theoretical shortcomings of the logistic regression model presented above, I first try logistic regression models with penalty terms for complexity using the R package *glmnet* before moving on to more complex models. Specifically, I consider LASSO, Ridge, ElasticNet (even weighting between L1 and L2 penalizations), and SCAD penalty terms. Presented below are the final cross-entropy score for each model and calibration plots of the final model, along with expected return and volatility figures. Note that *glmnet* trains via a descent-based algorithm using a quadratic approximation to log-likelihood, which is approximately equivalent to descent using binary cross-entropy. The SCAD penalization is done using the R package *ncvreg*.



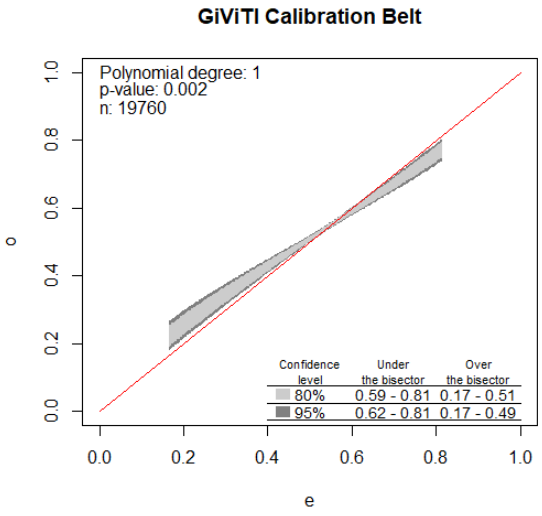
(a) Lasso Calibration: $H(\hat{p}, \hat{q}) = 0.6845$



(b) Ridge Calibration: $H(\hat{p}, \hat{q}) = 0.6845$



(c) ElasticNet Calibration: $H(p_f, q_f) = 0.6843$



(d) SCAD Calibration: $H(p_f, q_f) = 0.6849$

Despite reasonably good calibration plots, each of these techniques gives an out-of-sample binary cross-entropy score high enough that long term profitability is not possible. As shown in Table 6 below, a few of the models are able to achieve positive arithmetic returns but none are able to do significantly enough to overcome their own volatility; they all go bankrupt in the end.

Here are the return and volatility statistics for each model once they have been endowed with the above staking strategies to form an overall betting strategy:

	LASSO	Ridge	ElasticNet	SCAD
Raw Kelly				
Daily Expected Return (Arithmetic)	-0.17%	0.03%	-0.18%	-0.12%
Daily Expected Return (Geometric)	-6.47%	-6.37%	-6.49%	-6.51%
Max Daily Return	151.06%	163.83%	174.76%	156.18%
Max Daily Loss (Min Daily Return)	-61.66%	-62.43%	-61.28%	-63.29%
Daily Standard Deviation	37.28%	37.75%	37.41%	37.59%
Skewness	37.28%	108.26%	106.75%	105.56%
Kurtosis	116.69%	138.14%	134.48%	126.61%
Percent of Games Bet On	87.81%	87.53%	87.42%	87.72%
Max Drawdown	99.99%	99.99%	99.99%	99.99%
Selective				
Daily Expected Return (Arithmetic)	-0.77%	-0.81%	-0.74%	-0.57%
Daily Expected Return (Geometric)	-2.52%	-2.52%	-2.39%	-2.48%
Max Daily Return	85.13%	85.26%	81.33%	86.68%
Max Daily Loss (Min Daily Return)	-50.41%	-50.91%	-48.92%	-51.48%
Daily Standard Deviation	18.97%	18.75%	18.52%	19.90%
Skewness	70.65%	71.28%	78.28%	74.02%
Kurtosis	134.42%	132.63%	135.38%	127.06%
Percent of Games Bet On	7.98%	7.77%	7.57%	8.37%
Max Drawdown	99.99%	99.99%	99.99%	99.99%

Table 6: Return and Volatility Statistics: Penalized Logistic Regression Models

Looking at the backtested wealth paths of some of the above models, one sees faint glimpses of profitability owing to each strategy's high daily risk. For example, the SCAD model with Selective Staking amasses over 600% net returns at its highest point. However, over the entire testing set these models fail to remain profitable and are all effectively bankrupt after just a few seasons. However, compared to the baseline model, these penalized regression models have tighter calibrations, lower loss scores, and less monotonic wealth paths, a reassuring sign that long-term profitability is possible with further refinement.

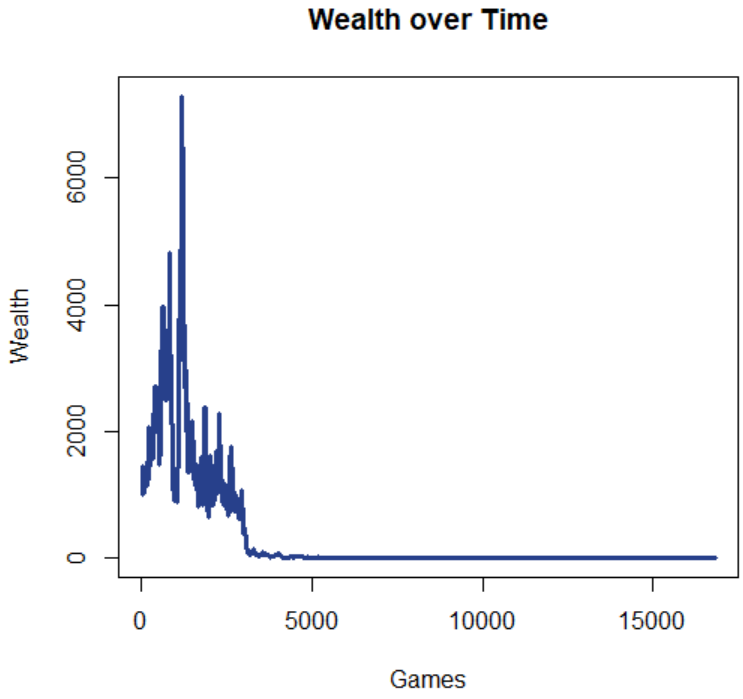
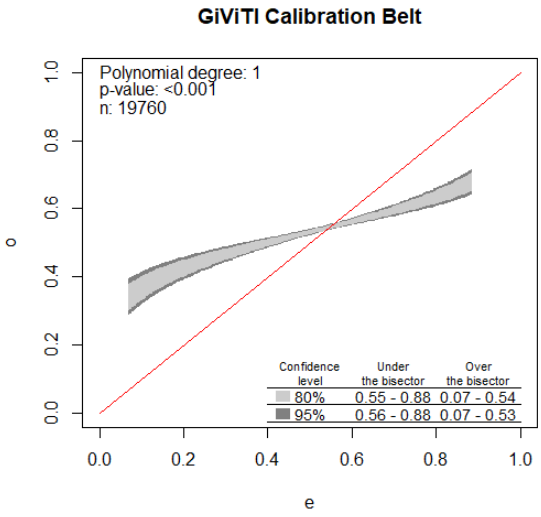


Figure 14: SCAD Wealth Path (Selective Staking)

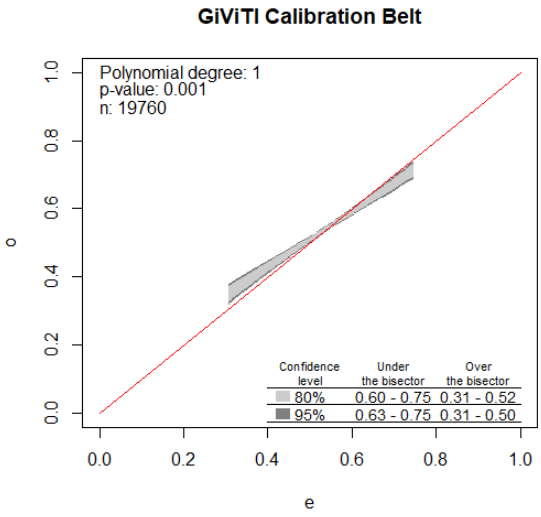
D Trees & Random Forests

For the boosted trees in this section, I used the R package *xgboost* and for the random forest models in this section I used the R package *randomForest*. The performance of random forest models was generally worse than the simpler penalized regression models presented in Section V and, in fact, it was surprisingly difficult to find a random forest model that could achieve good binary cross-entropy scores even on the training set. I present my best performing random forest model below, with accompanying return and volatility statistics. The boosted classifiers presented in this section are the most accurate of all individual models and have the

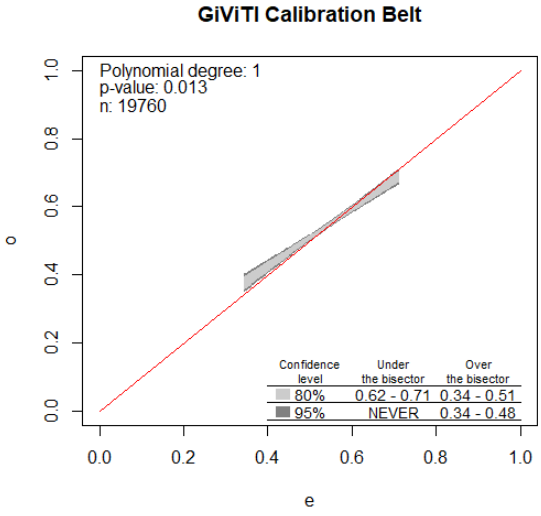
lowest binary cross-entropy scores when parameters are tuned correctly. The parenthetical parameters refer to the values for (*max.depth*, *nrounds*, and *min_child_weight*) given as arguments to *xgboost* respectively.



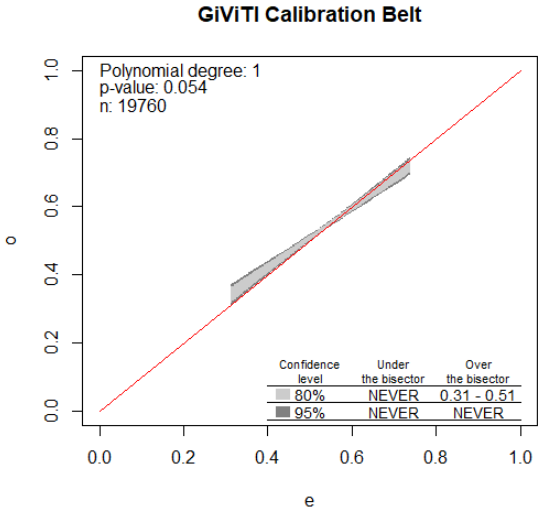
(a) Random Forest Calibration: $H(\hat{p}, \hat{q}) = 0.6973$



(b) Xgboost 1 Calibration: $H(\hat{p}, \hat{q}) = 0.6841$



(c) Xgboost 2 Calibration: $H(p_f, q_f) = 0.6841$



(d) Xgboost 3 Calibration: $H(p_f, q_f) = 0.6838$

	Random Forest	xg(3,100,1000)	xg(10,30,1000)	xg(2,100,1000)
Raw Kelly				
Daily Expected Return (Arithmetic)	-1.92%	-0.44%	-0.48%	-0.29%
Daily Expected Return (Geometric)	-7.41%	-6.40%	-6.56%	-6.43%
Max Daily Return	186.25%	212.97%	161.84%	177.10%
Max Daily Loss (Min Daily Return)	-64.82%	-61.10%	-63.05%	-64.39%
Daily Standard Deviation	34.00%	36.00%	36.39%	36.68%
Skewness	95.19%	103.50%	105.37%	103.52%
Kurtosis	162.81%	169.73%	173.70%	136.16%
Percent of Games Bet On	91.37%	87.18%	87.71%	86.54%
Max Drawdown	99.99%	99.99%	99.99%	99.99%
Selective				
Daily Expected Return (Arithmetic)	-0.96%	-0.15%	-0.37%	0.03%
Daily Expected Return (Geometric)	-7.00%	-1.82%	-2.07%	-1.63%
Max Daily Return	193.71%	93.14%	110.55%	94.57%
Max Daily Loss (Min Daily Return)	-67.23%	-53.28%	-57.06%	-48.26%
Daily Standard Deviation	36.45%	18.83%	19.03%	18.76%
Skewness	117.82%	96.41%	101.29%	95.99%
Kurtosis	231.96%	231.09%	255.02%	208.61%
Percent of Games Bet On	22.53%	7.56%	7.52%	7.02%
Max Drawdown	99.99%	99.99%	99.99%	99.99%

Table 7: Return and Volatility Statistics: Random Forest and xgBoost

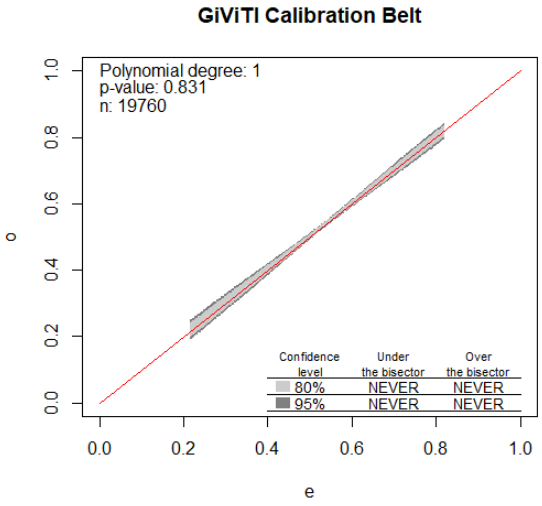
E Other Models

In addition to the machine learning models presented here, I tried a number of other models throughout the research process but was unable to achieve out-of-sample loss functions scores that surpassed even the baseline logisic regression model consistently. Among the models tested were neural networks, support vector machines (SVMs) with both linear and non-linear kernels, the naive Bayes classifier, other boosting packages (*JOUSBoost* and *LightGBM*), and Extreme Learning Machines (ELMs). Of course, the efficacy of these models is largely a function of hyperparameter tuning and the nature of the design matrix so success in applying these models profitably certainly should not be ruled out. In particular, support vector machines appear capable of producing well-calibrated probability estimates in the literature, under certain circumstances and with particular modifications such as Platt scaling, a topic explored in Section VII. Beyond simply applying more refined supervised learning models, there also may be considerable opportunity in pre-processing the data further, either manually or with unsupervised learning techniques beyond PCA.

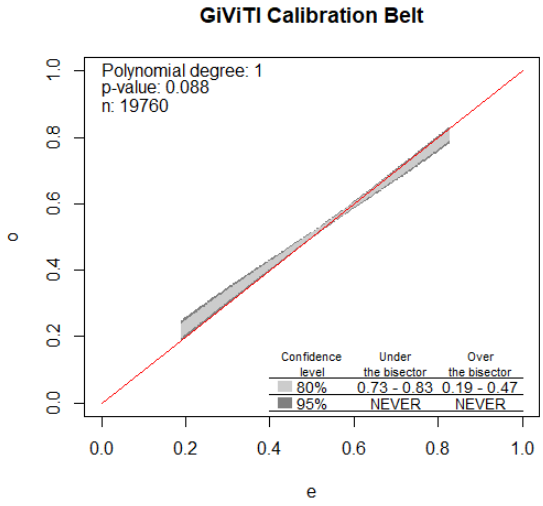
F Parameter Tuning and Cross-Validation

Thus far, the models presented have exhibited relatively disappointing return distributions. While some have achieved positive arithmetic average returns over the test set, the general result has been eventual bankruptcy. A possible explanation for this comes simply from suboptimal parameter calibration. In this section, I explore notable performance improvements through cross-validation based model selection. In particular, xgboost models appear to enjoy the greatest benefit from cross-validation-based model selection.

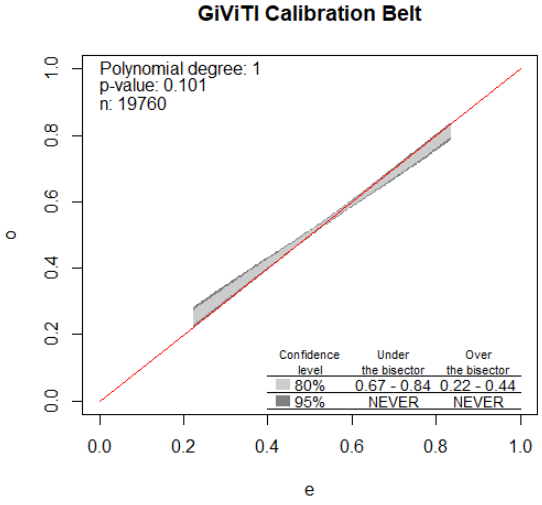
In order to find the optimal parameters for xgboost, the training set is 10-fold cross-validated in order to determine optimal values for *nrounds*, *eta*, *min_child_weight*, and other miscellaneous parameters. "Learning" the best parameters in this way yields substantial performance improvement. Shown below are the calibration plots and backtested returns for two strategies with parameters set via 10-fold cross-validation with *max_depth* set to 2, 3, and 4 respectively and the results achieved when averaging the three models:



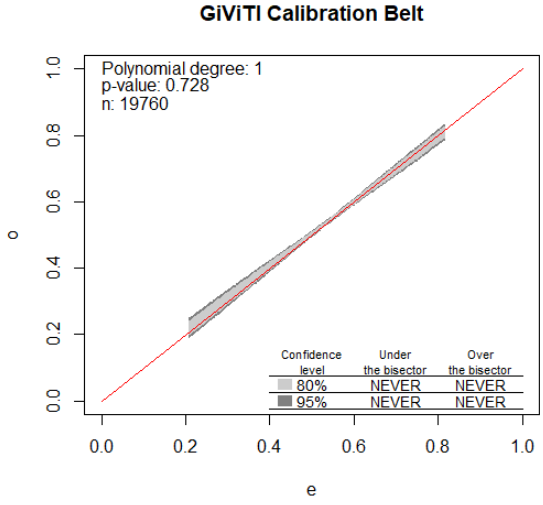
(a) Cross-Validated Xgboost(2): $H(\hat{p}, \hat{q}) = 0.6785$



(b) Cross-Validated Xgboost(3): $H(\hat{p}, \hat{q}) = 0.6792$



(c) Cross-Validated Xgboost(4): $H(\hat{p}, \hat{q}) = 0.6803$



(d) Simple Average Model: $H(\hat{p}, \hat{q}) = 0.6789$

	xg(md=2)	xg(md=3)	xg(md=4)	Ensemble
Raw Kelly				
Daily Expected Return (Arithmetic)	0.74%	2.42%	-0.03%	0.75%
Daily Expected Return (Geometric)	0.69%	1.05%	-0.25%	0.66%
Max Daily Return	15.65%	76.47%	27.14%	15.25%
Max Daily Loss (Min Daily Return)	-11.24%	-50.37%	-24.85%	-14.21%
Daily Standard Deviation	3.31%	17.00%	6.65%	4.07%
Skewness	21.31%	56.46%	23.48%	13.60%
Kurtosis	73.03%	84.09%	74.95%	60.47%
Percent of Games Bet On	35.57%	45.24%	48.00%	39.82%
Max Drawdown	40.00%	92.95%	98.52%	36.29%

Table 8: Return and Volatility Statistics: Cross-Validated Xgboost and Ensemble

VI Discussion of Modeling Results

Even slightly reducing the binary cross-entropy score through parameter selection via cross-validation yields highly profitable strategies. It is worth noting that higher returns are not the only benefit of better-calibrated probability estimates. Even before introducing a more sophisticated staking algorithm than raw Kelly criterion, I find a major volatility reduction compared to the previous models, simply as a consequence of being closer to the sportsbook's fair implied odds. Because the sportsbook's odds are generally quite well-calibrated, well-calibrated model probabilities will likely be highly correlated with the sportsbook fair odds, leading to lower staking sizes on average, and therefore, less volatility. In fact, this reasoning suggests that blending one's model estimates with the fair implied odds of a skilled sportsbook does much of the work of Kelly criterion modifications implicitly, a topic which will be further explored in Section X.

As an additional benefit, averaging these three models together allows for greater diversification in the final probability estimate through combining less-than-perfectly-correlated estimates, resulting in a strategy with more desirable volatility and drawdown characteristics. In general, I found the most effective probabilistic classifiers were composed of multiple well-calibrated, but not perfectly correlated models. The motivation for this approach comes from the insight that some models may perform better or worse depending on a number of characteristics about specific games. For example, just because a particular model provides very accurate probability estimates for toss up games (say $0.45 < \tilde{p} < 0.55$), there is no reason to necessarily expect this "expertise" to generalize to other subintervals of $[0, 1]$. Ensemble methods in their simplest form can consist of simply averaging the probability estimates from a number of models or by taking a weighted average based on some measure of confidence in each model. Alternatively, one can further partition the full dataset into training and testing sets for the original models and a final holdout test set for a meta-learner, which will learn to what degree to trust each base model in different situations, based on their performance on the original test set and some metadata about the games. For the purposes of this paper, I opted for the simple averaging technique.

Section V.F demonstrates that it is possible to construct betting strategies with binary cross-entropy scores approaching those of sportsbook lines given a sufficiently rich dataset (in terms of predictive power) and when using systematic techniques for model selection such as cross-validation. In Section X, I show that by combining multiple such models, one can arrive at a final model, which when combined with a sensible staking strategy, can prove immensely profitable in the real world.

In the remaining sections, I briefly explore two other possible modelling paradigms (rules-based techniques and an alternative conception of the loss function) before comparing the results presented here to the existing literature and some real world results.

VII Comparison to "Beating the MLB Moneyline"

The results presented in Section V.F represent a noticeable improvement over the existing literature. In their paper, "Beating the MLB Moneyline" Leland Chen and Andrew He attempt to construct profitable moneyline betting strategies using similar models as in this paper but with a very different set of features (Chen and He, 2010). Namely, Chen and He explicitly assume that the best predictive features are team statistics over the very recent past (on the order of the past 5 to 10 games). Over 2006-2009, they report the following binary accuracy scores (cross-entropy scores are not included):

Model	Classification Accuracy
Logistic Regression	59.25%
Linear SVM	56.69%
Polynomial SVM	60.05%

Table 9: Binary Accuracy: Chen & He

While the models applied are interesting and the accuracy scores are promising, I am quite skeptical about the feature set used in this paper. Team performance over a baseball season exhibits a high degree of mean reversion. This is precisely the rationale for playing such a large quantity of games: to ultimately determine the best teams by eliminating noise. It seems quite likely that team statistics over a short time period are essentially noisy versions of longer term trends, potentially hindering model performance substantially. Furthermore, by using player statistics, one guarantees that the historical statistics used as inputs in the relevant row of the design matrix actually derive from the players present in the game. Conversely, using team statistics implicitly assumes that players will contribute in the same proportions as in the recent past (in terms of playing time). It also ignores the possibility of trades, injuries, and any other changes to team composition over a season.

Chen and He also apply other models used in this paper but do not report their accuracy scores. However, using a random forest model and their best thresholding strategy, they report a total return of 4.48% over the period 2000-2009, equating to a 0.44% annual geometric return. Note that they only bet on the final third of each season, using the first two thirds as the training set for that season. Interestingly, Chen and He decided to use a thresholding modification similar to the one described earlier (i.e., only bet if the model's probability estimate exceeds the implied odds by a pre-specified threshold) but appear to use a fixed staking strategy. This staking strategy seems ill-advised compared to the Kelly criterion or some way of increasing betting proportion as a function of $(\hat{p} - p_h)$. It seems highly likely that staking in this way was suboptimal.

VIII Rules-based Strategies

As an alternative strategy, I analyze the performance metrics of selected rules-based investment strategies. Because these strategies do not involve estimating $(\hat{p}, \hat{q})$, I always bet 2.5% of the total bankroll for simplicity. The four rules-based strategies tested are as follows:

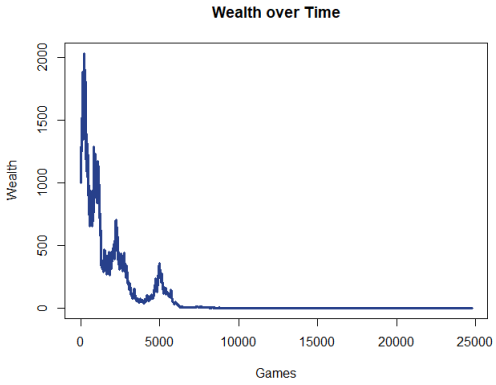
1. Always bet on the underdog. That is, bet on the home team if $p_f < q_f$ and bet on the away team if $q_f < p_f$. If $p_f = q_f$, do not bet.

2. Always bet on the home team.

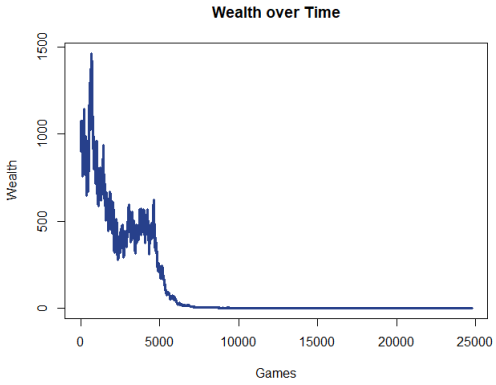
3. Always bet on the stronger starting pitcher. For this strategy, I select the team whose starting pitcher has a lower $t-1$ FIP (Fielding-independent pitching)

4. Always bet on "reverse line movement." For this strategy, I bet on the side of the game for which the sportsbook has reduced its implied odds. In other words, if the closing line p_h is less than the opening line p_h, I bet on the home team, and vice versa for the away team. This can be thought of as a contrarian betting strategy, taking the opposite view of the public's opinion (which is assumed to be the force creating the line movement). If the opening and closing implied probabilities are the same, do not bet.

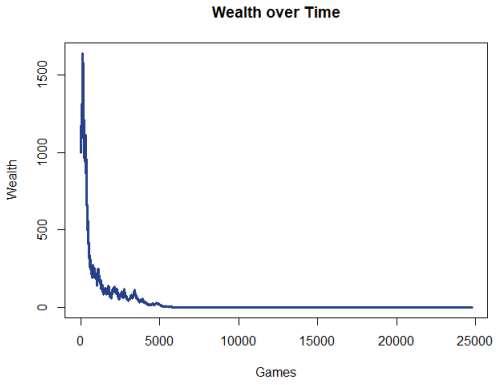
Here are the wealth paths for each strategy:



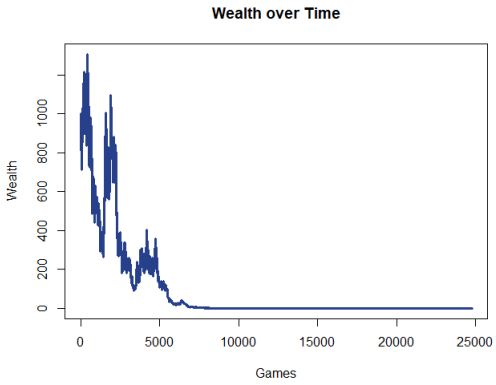
(a) Always Bet on the Underdog Wealth Path



(b) Always Bet on the Home Team Wealth Path



(c) Always Bet on the Stronger Starting Pitcher Wealth Path



(d) Always Bet on Reverse Line Movement Wealth Path

All of these oft-advertised strategies lead to eventual bankruptcy. The reverse line movement strategy is the only of the four with positive expected returns, and even still, is ultimately rendered unusable by its ratio of volatility to expected return:

	Underdog	Home Team	Pitcher	RLM
Daily Expected Return	-0.66%	-0.05%	-1.80%	0.32%
Daily Standard Deviation	10.31%	8.82%	8.97%	9.48%
Sharpe Ratio	-0.86	-0.08	-2.69	0.45

Table 10: Return and Volatility Statistics: Rules-based Strategies

A more sophisticated variant on rules-based methods involves codifying certain characteristics of each game (such as whether or not the team is home, differential between starting pitcher FIP, etc.) into formal ranked scores akin to quant alpha modelling and investing based on some normalized average of these scores. The main benefit (and arguably, the main drawback) of this approach comes from sidestepping the need to predict probabilities directly. Rather, such an approach would rely on identifying factors that are directionally associated with mispricing and determining staking amounts as a function of the game's ranking

on a number of these factors. For example, perhaps "underdog with strong pitcher," "accurate umpire," or "bad weather" could be defined as factors, with higher bets placed on games ranking highly on an average of these factors.

While there is some promise in such a strategy, it appears to suffer from a few critical drawbacks. Most notably, by relying simply on factor ranking for determining staking sizes, the strategy explicitly assumes a constant level of mispricing across games each day. In other words, if sportsbooks were to suddenly become more accurate on average, the strategy would be wagering too much on average and vice versa if sportsbooks became less accurate (assuming one has a good estimate of cross-sectional mispricing to begin with). Overall, the very simple nature of the pricing problem seems to lend itself most directly to investing based on some comparison of probabilities (with Kelly criterion or otherwise). Because prices and probabilities are equivalent in this asset class, it appears that excluding probabilities from one's analysis is a fatal modelling flaw. Throughout my research, I looked into a few potential factor-based strategies but was not able to find any which were profitable in the long-term. However, many such potential strategies exist, of which I tried only a small subset, and so there is certainly room for further exploration in this direction.

IX Alternative Loss Function

As a final aside, I consider an alternative to the training paradigm used in the above sections. As discussed in Section III.C.1, errantly high probability estimates are considerably more detrimental to a strategy's return profile than errantly low probability estimates. This is due to the nonlinearity introduced in the step following the probability estimation task, the investment decision. An errantly high probability estimate leads to an overly optimistic estimate of expected return under the Kelly criterion and thus, over-investment (additional risk). Conversely, an errantly low probability estimate leads to an overly pessimistic estimate of expected return, leading to under-investment or not betting at all. Of course, this outcome is also suboptimal owing to opportunity cost but is not nearly as dangerous as betting a large amount on a truly negative expected return game, for example.

Despite this important nonlinearity, the training procedure used to arrive at probability estimates above penalizes errantly high and errantly low probability estimates equally. Binary cross-entropy loss outputs the same loss when $|\hat{p}_i - y_i| = c$, for any c, regardless of whether $(\hat{p}_i - y_i) = c$ or $(\hat{p}_i - y_i) = -c$ despite the difference in harm to the strategy's return profile. While a symmetric loss function such as binary cross-entropy incentivizes the models to provide unbiased probability estimates, unbiased probability estimates may still lead to suboptimal investment decisions due to probability estimation error.

In addressing this issue, an alternative to the staking modification parameters proposed above and Robust Optimization (Sections III.C.1 and III.C.2 respectively) is compressing the probability estimation step and investment decision step into one step. Rather than using a traditional loss function (such as binary cross-entropy) which is agnostic to the final use case, this section explores whether applying a custom loss function allows one to reach more optimal staking proportions more directly.

A Return-based Loss Function

In deciding which custom loss function to use, I already have a solid candidate in the daily net return equation presented in Section III.D:

$$R_t = \frac{W_t}{W_{t-1}} - 1 = -\sum_{i=1}^{k} x_i^* + \sum_{i=1}^{k} \mathbb{1}_{\{y_i = g_i\}} m_i x_i^*, \ \forall t \in \{1, 2, ...\} \tag{32}$$

Simplifying the return function to apply to only a single game (i) yields:

$$R_i(y_i, x_i^*) = -x_i^* + \mathbb{1}_{\{y_i = g_i\}} m_i x_i^* \tag{33}$$

Rather than training by iteratively minimizing $H(\vec{y}, \vec{\hat{p}}, \vec{\hat{q}})$ (binary cross-entropy as a function of the model's

probability estimates and the actual outcome), one can bypass the probability estimation step entirely and minimize

$$-(\vec{1} + R(\vec{y}, \boldsymbol{X}^*)) = -\big(1 + \prod_{i=1}^{N} R_i(y_i, x_i^*)\big) \tag{34}$$

(negative gross portfolio return compounded over the training set). The first column of $\boldsymbol{X}^*$ represents the staking proportion on the away side of the bet and the second column represents the staking proportion on the home side ($\boldsymbol{X}^* \in \mathbb{R}^{N*2}$). In general, if ${\boldsymbol{X}^*}_{i,1} > 0$, then ${\boldsymbol{X}^*}_{i,2} = 0$ and vice versa (i.e., bets will not be placed on both sides of the same game).

Since the model is directly learning x_i^* for each game, estimating $(\hat{p}, \hat{q})$ directly is not required and one need not apply the Kelly criterion; the model is now incentivized to output sensible investment decisions directly rather than unbiased probability estimates, and thus is sensitive to the final use-case. Presumably, this allows the model to account for the nonlinear payoffs of each bet and adjust staking decisions based on learned pitfalls, rather than requiring the user to manually and arbitrarily force different staking levels through the modification parameters used in the standard staking strategies above.

B Empirical Findings

To test the above hypothesis, I trained a series of Keras neural networks using this custom loss function with different hyperparameters on the original training set and obtained proposed staking proportions. Since I no longer obtain $(\hat{p}, \hat{q})$ as an output of the model, I cannot produce calibration plots or calculate binary cross-entropy. Instead, I can only directly assess model performance through the numerical value of the return-based loss function or through backtesting. Unfortunately, throughout my research using this alternative loss function, I was never able to achieve profitability as measured by the loss function. In other words, no model I used was able to minimize the loss function below -1 out-of-sample. In other words, I never achieved $R(\vec{y}, \boldsymbol{X}^*) > 0$ averaged over the testing set. While I was unable to achieve profitability using this learning paradigm, considerable research opportunities appear to exist here. The prospect of learning the investment decision, rather than probabilities, counteracts any issues arising from probability estimation error and is fundamentally more elegant. However, doing so requires managing the challenges of training with a non-differentiable custom loss function or applying a more apt reinforcement learning algorithm.

X Real World Results (2021 MLB Season)

As the true final robustness check for the ideas in this paper, I decided to apply the concepts introduced thus far to generate probability estimates for true out-of-sample games over the first three months of the 2021 MLB season (April - June). I built a predictive model using the conceptual framework introduced in Sections III through V, ending with a model constructed as a simple average of four different xgboost models, two of which were trained on the full feature set and two of which were trained on a degree 2 polynomial transform of a subset of the dataset. From here, I blended 2-to-1 with the given fair implied probabilities from Pinnacle. Given the statistically unrepresentative nature of the 2020 MLB season (only 60 games rather than 162 and no fans due to the Covid-19 pandemic), I decided to primarily use single season stats from two years prior during model training for the 2021 season and simply omit any 2020 statistics from the training data entirely.

A Daily Probability Estimation

To prepare for the season, I used two datasets, $(y_{standard}, X_{standard})$ and $(y_{poly(2)}, X_{poly(2)})$ to train four functions $f_i : \mathbb{R}^{k_i} \to [0,1]$ for $i \in \{1,2,3,4\}$ with $k_i = 676$ for $i \in \{1,2\}$ and $k_i = 2346$ for $i \in \{3,4\}$, now on the entire dataset rather than just the training set. Throughout the season, I used each new day's games to construct the true out-of-sample versions of the input data: $X^{new}_{standard}$ and $X^{new}_{poly(2)}$.

Applying the four models trained above, I produced daily probability estimates for each of the day's j games, $(\hat{p}^j_i, \hat{q}^j_i)$ for $i \in \{1,2,3,4\}$, $j \leq 15$. Simply averaging the estimates led to my raw probability estimate for each game:

$$(\hat{p}^j = \frac{1}{4} * \sum_{i=1}^{4} \hat{p}^j_i, \hat{q}^j = 1 - \hat{p}^j) \tag{35}$$

These probability estimates, when combined 2-to-1 with Pinnacle's fair implied probabilities yielded the final probability estimates used for each day's games:

$$(\hat{p}^j_{final} = \frac{2}{3} * \hat{p}^j + \frac{1}{3} * \hat{p}^j_{f,Pinnacle}, \hat{q}^j_{final} = 1 - \hat{p}^j_{final}) \tag{36}$$

The decision to blend the ensemble model's probability estimates with Pinnacle's fair implied probabilities served two major purposes. Firstly, this decision acted as an additional risk management mechanism overlaid on top of the staking parameters introduced in Section III.C.1, subduing the negative effects of individual ensemble model probability estimates occasionally deviating significantly from sportsbook estimates. Secondly, and more critically, since Pinnacle generally has more accurate probability estimates than other sportsbooks,

the blending in of its estimates allowed the final strategy the model to capture the inter-sportsbook pricing differential in addition to the outperformance associated with its own probability estimates.

B Strategy Staking Decisions

Given daily probability estimates for each available game, the next step was to arrive at final wager amounts for each game, as a percentage of the total portfolio. To do this, I directly applied the framework established in Sections III.B and III.C. $\vec{x^*}$ was calculated by extending the optimization program in Section III.B to the applicable number of games each day and solving directly. From here, I used the following staking modification parameters to arrive at the modified daily staking vector, x^*_{mod}:

a	f	m
0.01	0.80	0.15

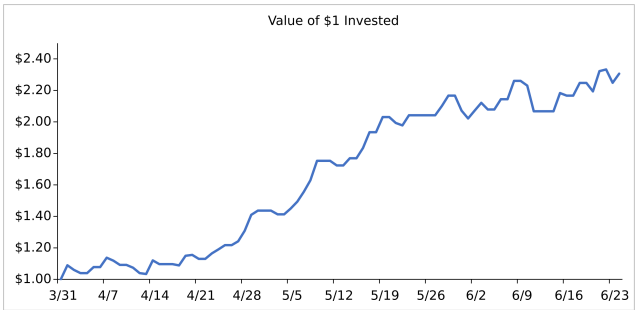
Table 11: 2021 Model Staking Parameters

$$x^*_{mod} = \begin{cases} min\{0.15, 0.8x^*\}, \ (p_{\hat{final}} - 0.01) > p_h \\ 0, \text{ otherwise} \end{cases} \tag{37}$$

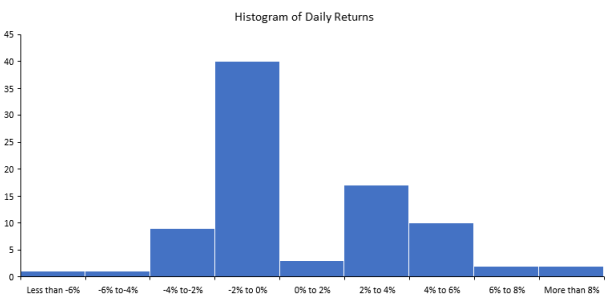
Intuitively, the first check is whether any games appear to be mispriced by at least 1% (on either side, home or away). If so, the model bets 80% of the stake suggested by raw Kelly, up to a maximum of 15% of the total portfolio. In practice, this 15% ceiling was never even approached, with a maximum bet size of just over 5%. After some final discretionary modifications to x^*_{mod}, namely excluding games in which either starting pitcher did not have sufficient history in MLB and excluding all 7 inning doubleheaders (a regrettable rule change introduced at the start of the 2020 MLB season), I arrive at final daily investment decisions: $x^*_{final} \in \mathbb{R}^{n_j * 2}$

C Strategy Performance

From April 1 through June 24, 2021, out of a total of 946 "eligible games," I bet on 99 games (10.47% of available opportunities) using the strategy described above, winning 53 of them (53.54%). Over this period, the strategy achieved a net return of 130.66%, broken down into returns of 43.68%, 44.20%, and 11.33% in April, May, and June, respectively. Despite a slowdown in performance in June, the strategy performed consistently well over the period, with average (arithmetic) daily returns of 1.03% and a daily standard deviation of 3.01%.



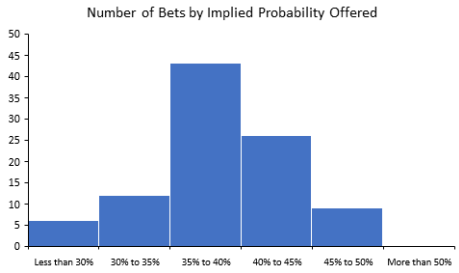
(a) Evolution of $1 Invested (April 1 - June 24)



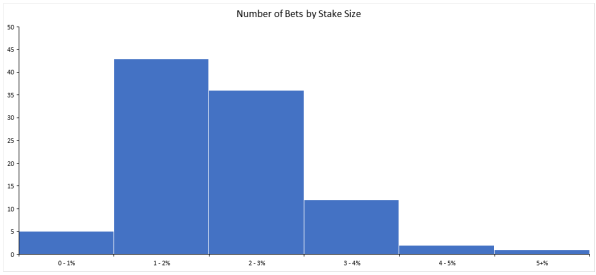
(b) Histogram of Daily Returns (April 1 - June 24)

C.1 Average Bet Characteristics

The strategy heavily favored underdogs, never betting on a team favored to win. In fact, the stake size weighted average odds for a bet taken was +164, corresponding to a sportbook implied probability of just 37.87%, or equivalently, a payout multiplier of 2.64. 69 of the 99 bets had a sportsbook implied probability (offered line) between 35% and 45%. Most bets were individually quite small, with only 15 bets representing larger than 3% of the bankroll at any given time and an average bet size of just 2.18% of the total portfolio.



(a) Number of Bets by Sportsbook Implied Probability



(b) Number of Bets by Stake Size

Throughout the season, I bet using two New Jersey sportsbooks, always placing each bet with the book offering the more favorable odds (lower implied probability).

C.2 Return Attribution

Given the large deviation between the average sportsbook implied probability (37.87%) and the strategy's hit rate (53.54%), it is natural to wonder how much of the return captured by this differential is attributable to different sources. Here I decompose the strategy's returns into true model skill, probability estimation error, and raw luck.

At first glance, consider that the stake-size weighted average model probability estimate ($\hat{p}$ or $\hat{q}$ depending on the side bet on) was only 40.10%; just looking at averages, the model bet on games with a 37.87% sportsbook implied probability with an expectation of winning 40.10% of these games. In reality, the strategy

won 53.54% of these games, intuitively indicating that a decent portion of the strategy's returns over this period are attributable to good luck alone, with some of the returns attributable to model skill. Another way to see this is to consider these numbers in the context of a single bet; as mentioned above, a 37.87% implied probability corresponds to a payout multiplier of 2.64. If a bettor is able to take bets with such a payout repeatedly and wins 40.10% of them (exactly as his model predicts), his expected net return per bet is $(40.10\% * 2.64 + 59.90\% * 0) - 1 = 5.86\%$. However, if his model again predicts the bets should pay out 40.10% of the time but they empirically end up paying out 53.54% of the time, his model is much less accurate but he has a noticeably higher expected net return: $(53.54\% * 2.64 + 46.46\% * 0) - 1 = 41.35\%$, with the differential attributable to good luck.

A Monte Carlo simulation of the games the strategy bet on confirms this intuition. In order to better isolate luck and skill over this stretch, I simulate 10000 versions of the 99 games bet on over the season using the same staking amounts but setting the probability of winning each game equal to the probability predicted by model. In this sense, this simulation illustrates possible wealth evolution paths in a world in which the model's probability estimates $(\hat{p}_{final}, \hat{q}_{final})$ equal the true (oracle) probabilities $(\tilde{p}, \tilde{q})$. For ease of viewing, I show just 100 of these possible paths below.

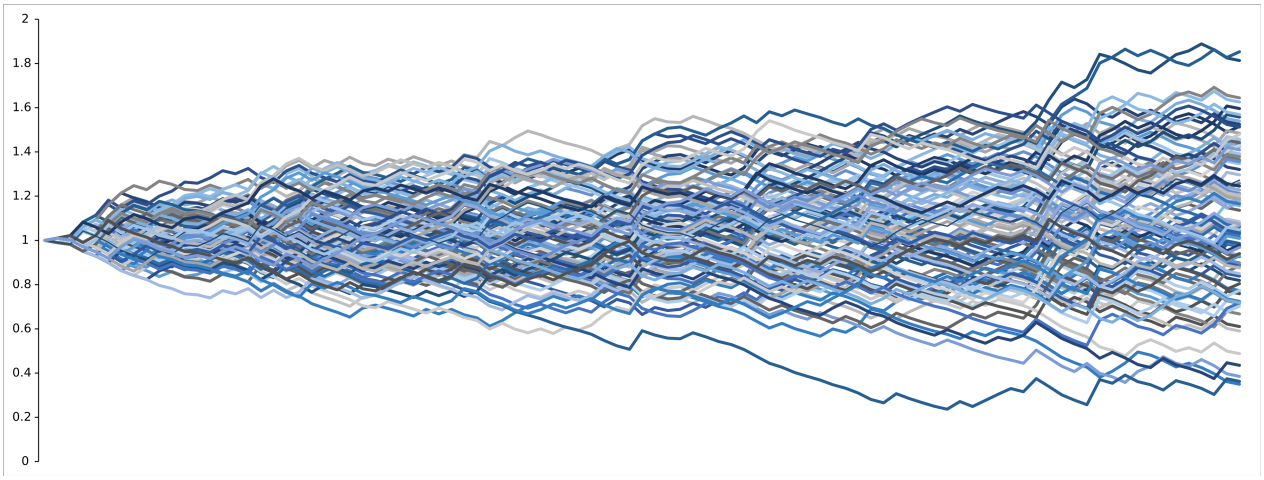


Figure 20: 100 Example Monte Carlo Simulation with Model Probabilities

Over the 10000 simulations, the average ending value of $1 invested is $1.11 (11% return) with a standard deviation of returns of 29%, compared to the $2.31 (131% return) observed above. Thus, in the hypothetical world in which the model's probabilities are perfectly calibrated, the returns actually observed correspond to a +4.14 standard deviation event and so one can be quite confident that the model probabilities are not incredibly well-calibrated.

XI Conclusion

The research presented in this paper seeks to answer two key questions. Firstly, it explores whether statistical/machine learning techniques can be applied to past game and player data to produce more accurate estimations of a given team's probability of winning a given MLB game relative to sportsbook implied probabilities. Secondly, this research seeks to explore how well-calibrated probability estimates can be used to construct daily portfolios profitable enough to overcome relatively large implicit transaction costs present in the MLB moneyline market.

In general, answering the first question is non-trivial; the research presented in Sections IV and V mostly highlights many ways to fall short of this task. I found that naive application of a number of popular probabilistic classifiers is not sufficient to exceed sportsbooks' calibration; simple logistic regression, penalized logistic regression, neural networks, boosted trees, support vector machines, and random forest models, among others tested, all fail to achieve lower binary cross-entropy loss function scores than sportsbook implied probabilities on average. However, this research also shows that model performance can be significantly improved using cross-validation to optimize parameters and ensemble methods to intelligently combine insights from different models. In applying these techniques, one can produce probabilities with similar binary cross-entropy scores to sportsbook implied probabilities.

In addressing the second question, I developed betting strategies capable of earning positive returns in backtest space. What remained unclear at this step was the correct allocation of credit to competing explanations for this result. To what extent did the backtested models constitute novel predictive analytics and to what extent did the available historical odds data understate the quality of sportsbooks' estimates in reality? The real world results presented in Section X seek to answer this question. While a good deal of luck appears to have contributed to the 131% return figure, sportsbook prediction accuracy and binary cross-entropy generally appeared to be in line with the historical sportsbook data used in backtesting.

While it is possible to achieve impressive returns through application of the principles in this paper, the scalablity of such an investment strategy is probably limited; perhaps exclusion/bet size limits from sportsbooks are prohibitively obstructive or perhaps any substantial level of success will eventually lead to market impact; any competent sportsbook is sure to notice a consistently profitable bettor. While the dataset in this paper (and previous literature) shows only slight improvement in sportsbook calibration over time, perhaps sportsbooks are beginning to implement these sorts of predictive models themselves and will soon move within a safe range in which margin makes them impossible to beat consistently.

References

Matthew Bouchard. "Information and Market Efficiency: Evidence From the Major League Baseball Betting Market". In: *Bachelor's thesis, Harvard College* (2019). DOI: https://nrs.harvard.edu/URN-3:HUL.INSTREPOS:37364609.

Markus Brunnermeier. "Notes for the Course FIN 501: Asset Pricing I". In: (2014).

Leland Chen and Andrew He. "Beating the MLB Moneyline". In: (2010).

J.L. Kelly. "A New Interpretation of Information Rate". In: *The Bell System Technical Journal* 35.4 (1956), pp. 917–926. DOI: 10.1002/j.1538-7305.1956.tb03809.x.

John M. Mulvey and Robert J. Vanderbei. "Robust Optimization of Large-scale Systems". In: *Operations Research* 43.2 (1995), pp. 264–281. DOI: http://links.jstor.org/sici?sici=0030-364X%28199503%2F04%2943%3A2%3C264%3AROOLS%3E2.0.CO%3B2-H.

David B. Rosen. "How Good Were Those Probability Predictions? The Expected Recommendation Loss (ERL) Scoring Rule". In: *Maximum Entropy and Bayesian Methods* ().

Linda M. Woodland and Bill M. Woodland. "Market Efficiency and the Favorite-Longshot Bias: The Baseball Betting Market". In: *Journal of Finance* 49.1 (1994), pp. 269–279. DOI: https://econpapers.repec.org/RePEc:bla:jfinan:v:49:y:1994:i:1:p:269-79.

Acknowledgements

The research presented in this paper came to fruition largely due to the exceptional encouragement of my Master's thesis supervisor, Caio Ibsen Rodrigues de Almeida. Upon my initial proposal of the project, Professor Ibsen Rodrigues de Almeida recognized it as an unconventional, but potentially fruitful application of portfolio optimization, risk management, and machine learning techniques. Despite a merely tangential relationship between baseball betting and other established areas of quantitative finance at surface level, Professor Ibsen Rodrigues de Almeida eagerly endorsed the project and provided invaluable guidance throughout the research process. Even after my graduation in June 2020, Professor Ibsen Rodrigues de Almeida offered his continual support and guidance as I refined the original work and applied the paper's findings in practice. His genuine excitement about my research (despite no previous knowledge of baseball!) is representative of the catalytic impact his involvement has had on the research produced by many of classmates. I am incredibly grateful for this dedication to mentorship, a gift which incalculably improved the quality of this paper.