Beating the Book with Machine Learning

Team: Ethan Ma, Eric Wang, Chung Un Lee (Richard). **Project Mentor TA:** Frank Liu

1) Abstract

The sports betting market has been growing substantially in recent years. To capture this opportunity, we are exploring the potential to profit from sports betting on the NBA. A key challenge in reliably beating a sportsbook is the necessity of having predictions that differ from those of the sportsbook itself. To address this, our first significant contribution involves gathering up-to-date data from the most recent complete NBA session (2022-23). This dataset includes comprehensive team and player statistics, betting lines, and match outcomes which are used to develop our predictive models. Our second major contribution is the development of a new loss function specifically designed to prioritize deviations from the sportsbook's predictions. Our evaluations indicate that the sportsbook's predictions are often accurate, underscoring the need for an approach to achieve a competitive edge in sports betting. Ultimately, for the games our model felt most confident that the home team would win, we achieved a value approaching ~35% where our model was correct while the sportsbook was wrong on smaller sample sizes of predictions and ~30% on larger sample sizes.

2) Introduction

Recent concerns about the growing popularity of sports betting underscore the importance of approaching this booming industry with caution. While it is a widely held belief that consistently outperforming bookmakers is nearly impossible and that engaging in sports betting typically results in a net loss, our project aims to explore this challenge critically. We have developed a robust dataset and predictive model focused on NBA games, utilizing innovative loss functions designed to diverge from traditional bookmakers' predictions. This approach not only aims to potentially improve prediction accuracy but also to provide valuable insights into the interactions between different predictive models and the impact of novel loss functions on these interactions. Through this research, we hope to contribute to a more informed understanding of sports betting, potentially mitigating its gamification and fostering a more analytical view of its outcomes.

Inputs: Our dataset includes team and player statistics, betting lines from bookmakers, and actual match outcomes for the 2022-23 season. Player and team statistics encompass key performance metrics such as points, assists, steals, and blocks. Match outcomes are denoted by wins or losses, and we obtained the betting lines from Rotowire.com.
Outputs: Our objective was to predict match outcomes through logistic regression and random forest classifiers using the input dataset so that our models beat the predictions made by the book. We achieved reasonable performance with both baseline models and models with adjusted loss functions. The most important features identified were wins (W), field goals attempted (FGA), field goals made (FGM), steals (STL), points (PTS), defensive rebounds (DREB), free throws attempted (FTA), turnovers (TOV), and blocks (BLK).

3) Background

A. Exploiting sports betting market using machine learning
   a. Link:https://www.researchgate.net/publication/331218530_Exploiting_sports-betting_market_using_machine_learning
   b. Unlike previous studies that focused solely on maximizing predictive accuracy, this paper introduces three main innovations. First is the decorrelation from Bookmaker's Model. The paper emphasizes the importance of reducing the model's correlation with the bookmaker's predictions, as aligning too closely with these predictions can lead to losses due to inherent market margins. Second is the use of Convolutional Neural Networks. For the first time, CNNs are applied to match outcome prediction, leveraging extensive player-related statistics. Third is the application of modern portfolio theory. They utilize elements from modern portfolio theory to optimally balance the trade-off between profit expectation and variance in bet distribution.
   c. We were particularly drawn to their modification of the loss function to punish samples that were too closely correlated with the given lines, and we have designed our own loss functions that prioritize deviations from the sportsbook's predictions.
   d. The paper applied CNNs for prediction. However, we opted to use logistic regression and random forest classifier due to the limited number of games played per season. CNNs require a significantly larger dataset for accurate predictions, and incorporating historical data would be inadequate because teams and players change each season.
B. https://github.com/NBA-Betting/NBA_Betting?tab=readme-ov-file#mldl-modeling
   a. This is very similar to our project's goals, however, we want to implement more up-to-date data, as well as more specific recommendations in relation to game lines. Because the numerous factors that can impact a team's performance can change drastically with time, the findings and results of previous sports-betting-based research projects on older data may not be particularly useful without the incorporation of new data.

4) Summary of Our Contributions

Our contributions lay in two primary areas:

1. Contributions to data: Because existing publicly available datasets such as those on Kaggle are rarely updated frequently enough to capture recent match and player data, which is often the data most critical for making accurate predictions, our project aimed to augment existing datasets by scraping up-to-date data. In addition to the player and team statistics data that many previous works have utilized, we included historical match outcomes (win/loss) and betting line data from a sports betting website.
2. Contributions to models: The sportsbook models of today are extremely accurate and hard to beat. Any given competing model will likely fail to be superior enough to make a profit, or at best correlate strongly with the sportsbook's models. When the costs associated with betting margins are factored in, simply guessing game outcomes does not seem to be a viable way to make money. As a result, rather than simply emulating existing books, we decided to develop a model to tackle the inaccurate home line set by the bookmaker, which is recorded by RotoWire. By modifying the loss function to

penalize predictions that align with the book (to prioritize deviations from the sportsbook's predictions), we saw improvements in how many games we correctly predicted over the sportsbooks.

5) Detailed Description of Contributions

5.1 Methods

Data Contribution (Contribution 1)

In terms of gathering data, research suggests that the choice of features is more important than the choice of the model itself [Zimmermann, Moorthy, and Shi; 2013]. Knowing this, we aimed to provide an up-to-date dataset with features backed by research (listed above, and the choice of features listed as well in the Hubacek, Sourek, and Zelenzy paper). To do this, data from the 2022-23 (the most recent completed season) was scraped, and then the results were cleaned, combined, and labeled to create new .csv files that were ML-ready.

- Data Collection Mechanism
    - We conducted extensive desktop research to find resources that are relevant to our project. Finding player and team statistics data was straightforward as we were able to obtain it from the official NBA website. We then scraped all the match outcomes of the 2022-23 season (which is the most recent complete season) from Google and the betting line data from a sports betting website, rotowire.com.
    - We mainly utilized Selenium to scrape data from various sources.
- Quick view of the final dataset

	label	home_line	W	PTS	FGM	FGA	TPA	TPM	FTM	FTA	OREB	DREB	REB	AST	TOV	STL	BLK	BLKA	PF	PFD
0	1	-9.5	1	-1.0	0.5	1.6	-2.3	-1.2	-0.9	0.8	1.4	1.9	3.3	-0.9	1.2	0.0	-0.3	0.0	-0.8	-0.1
1	0	-2.0	1	-1.0	0.5	1.6	-2.3	-1.2	-0.9	0.8	1.4	1.9	3.3	-0.9	1.2	0.0	-0.3	0.0	-0.8	-0.1
2	0	-4.5	4	1.7	1.9	6.6	7.7	2.2	-4.4	-2.7	2.4	5.3	7.7	0.6	0.9	-1.3	0.2	-0.7	-2.4	-0.6
3	1	-5.0	4	1.7	1.9	6.6	7.7	2.2	-4.4	-2.7	2.4	5.3	7.7	0.6	0.9	-1.3	0.2	-0.7	-2.4	-0.6
4	1	-12.5	5	1.1	-0.9	4.0	9.1	3.0	-0.2	0.0	1.0	4.6	5.6	-3.1	0.1	-1.1	0.4	-0.3	-0.6	-0.5

    - This is our final dataframe obtained by combining player and team statistics, match outcomes, and betting line datasets.
    - Label indicates win (1) or loss (0), and home_line indicates the betting line set by the bookmakers. Other features are the performance statistics of the home team and players.
- Data Distribution
    - See Appendix A for visualizations of data distribution.
    - Other than the binary feature 'label', all features generally exhibit normal distribution without significant skew.
- Data Preparation for ML
    - Data imputation was not needed as we did not have any null values
    - We inner-joined player, team, and match outcome datasets on home_team and away_team columns. We obtained the combined performance statistics by subtracting the average statistics of the away team from the average statistics of the home team to merge the datasets. Finally, we joined the merged dataset with

the betting line dataset on tipoff, season, and home_team to ensure the win/loss labels were correctly assigned.

- Baseline Models to test ML-readiness
  - Logistic Regression
    - Training accuracy of Logistic Classifier: 62.3%
    - Testing accuracy of Logistic Classifier: 60.4%
  - Random Forest Classifer
    - Training accuracy of Random Forest Classifier: 68.4%
    - Testing accuracy of Random Forest Classifier: 59.4%

Model Contribution (Contribution 2)

To provide separation from the sportsbook predicted outcomes, the original paper [Hubáček, Ondřej & Šír, Gustav & Železný, Filip] proposed a couple of methods. The first was to do a weighted sampling of data. For example, sampling a particular data point multiple times (or equivalently assigning it a higher weight) depending on the bookmaker's odds, so games with higher pay-outs are emphasized more. The second, which is the one we decided to contribute to, was to modify the loss function.

Loss Function Development:

Beginning with an open source implementation of logistic regression, we wanted to redefine the loss for our logistic regression in order to best suit the goals of our model. Since we were ultimately dealing with a classification problem by predicting the winner of any given match, we opted for cross entropy loss as a more suitable loss function (as opposed to the square loss used in the guiding paper). By using the home-line, we simply classified negative values as the sportsbook predicting a home win, and otherwise a home loss.

The idea behind our penalty term was simple: we want to penalize our prediction each time it lines up with the sportsbook. Thus, a straightforward evaluation metric was to take p_o (our prediction) and p_c (the sportsbook prediction), and calculate the term ~xor(p_o,p_c)=and((p_o,p_c)+~or((p_o,p_c). A hyperparameter c could then be used to scale the magnitude of the penalty term. Thus, the full loss function is written below and is similar to our work in Homework 5:

$$L = \frac{1}{n}\sum_{i=1}^{n}[y_i log(p_o) + (1-y_i)log(1-p_o)] + \frac{c}{n}\sum_{i=1}^{n}[max(0, p_o + p_c - 1) + max(0, 1 - p_o - p_c)]$$

Where $p_o = \sigma(\theta^T x_i)$

The next issue was applying this to our model, specifically through gradient descent. Fortunately, each of the terms was essentially ReLU (in terms of the math), and thus the gradient of the inner term of the summation was straightforward to take:

$$[p_o(1-p_o)x_i] * a - [p_o(1-p_o)x_i] * b$$

Where a is an indicator if $p_o + p_c + 1 \geq 0$ and b is an indicator if $1 - p_o - p_c \geq 0$

5.2 Experiments and Results

Data Augmentation (Contribution 1):

Our team was able to create ML-ready data, below is what part of a row of data looks like in our results:

	home_team	away_team	W	PTS	FGM	FGA	TPA	TPM	FTM	FTA
2	MIL	BOS	1	-1.0	0.5	1.6000000000000100	-2.3000000000000000	-1.2000000000000000	-0.8999999999999990	0.7999999999999970

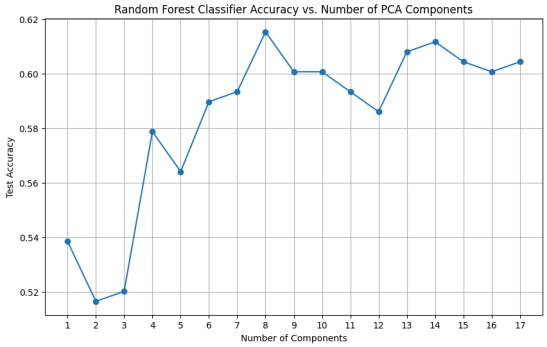
This data was scraped from the official NBA website via pages on player stats and comprehensive match histories. The scraping itself was done via selenium. Since each game has 2 teams, data was labeled 1 if the home-team won, and 0 if the away-team won. Each final feature was also the difference between the home-team statistics and the away-team statistics.

Baseline Models:
To see how ML-ready our data was, we wanted to create baseline models that we could use to not only evaluate our data, but also the feasibility of our chosen features. The results are listed in the introduction.

PCA/Feature Analysis:

```
Top Features by Importance:
OREB: 0.1216
PTS: 0.0868
FGA: 0.0831
TPA: 0.0816
FTM: 0.0718
TPM: 0.0564
FGM: 0.0561
REB: 0.0512
BLK: 0.0506
FTA: 0.0484
```

In the information above it's important to note that these are the top features ranked by importance after the W(wins) feature was removed, which had by far the largest ranked importance by an order of magnitude. This is understandable since teams which have won previously are likely to win subsequently, but it does raise some questions about the ability of our model to generalize.



Here, it seems like 8 components resulted in the largest accuracy gains, though overall accuracy remained fairly unimpressive around 62% with a baseline random forest classifier.

Performance Evaluation (Contribution 2, Appendix B: Figures 21-24):
Evaluation is inherently tricky since the sportsbook is usually right (about ~68% of the time for the 2022-23 season). As a result, simply evaluating accuracy like in traditional models won't be the best metric. Recall that we are penalizing similarity to the sportsbooks, meaning many times we are penalizing the outcome as being correct. Thus tuning the penalty hyperparameter c becomes incredibly important since having the penalty term dominate may lead to unexpected behavior. We settled on the following evaluation method.
For our most confident home-win predictions (predictions with the highest raw score generated by the logistic regression) we can measure how many times we correctly predict the label while the sportsbook is wrong. Thus, by taking the top X most certain predictions of our model we can expect to beat the sportsbook a certain number of times. However, the evaluation magnitude, or

the number of predictions from our model that we actually utilize is also critical, since if we use too few predictions we derive little benefit, while using too many predictions results in decreasing accuracy. In addition to Graph 21 of Appendix B which simply shows the raw number of predictions on the y-axis, we thus also need to examine the ratio of differing correct predictions to the total evaluation magnitude (Graph 22 of Appendix B). Seemingly, the optimal value for our hyperparameter was around 2, predicting correctly over the sports book at a rate of about 50% on smaller sample sizes, and ~35% on larger sample sizes on all data. When considering only test data (Graphs 22, 23 of Appendix B), values hovered around 30% - which shows some ability to generalize.

6) Compute/Other Resources Used

Because the implementation of logistic regression that we implemented for our homework assignment produced divide-by-zero errors that resulted in predicted accuracy values that either exploded to values approaching one or 0 with the sigmoid applied, we referenced a medium article (cited in works referenced section below) on logistic regression to try to resolve the issue, replacing their cross-entropy loss based implementation with our own custom penalty function.

7) Conclusions

Outcomes: The model we created was able to successfully outperform simple baseline models like random forest and vanilla logistic regression through the implementation of a novel penalty term that specifically aimed to interact with and respond to the conventional predictions of a sportsbook. This certainly emphasizes the power that penalty terms can have over guiding a model toward specific or complex behaviors that conventional models may have difficulty capturing.

In Hindsight: It was very interesting to see how seemingly ‘reasonable’ the predictions of our model were. The ratio of ‘correct’ predictions of our model to ‘incorrect’ predictions of the sports book settled in at around ~0.35 once considering a decently sized amount of data. While this number seems to be a great indicator of general success, we believe that there could have been better ways to actually measure monetary gain. The main challenge we had in this area was collecting ‘odds’ data (i.e. +/-125), and we did not want to speculatively translate home lines to probabilities to ‘odds’ data. We also believe it would have been interesting to test this on the actual live NBA season, but the timing with the regular season ending before our model was complete did not allow us to do so.

Ethical Considerations, and Impact: One of the first conclusions that we reached while working on this project was the fact that winning against the sportsbook is essentially impossible. Simply put, any predictive model that you put together is unlikely to be more accurate than the sportsbook, and any model that is will still bleed money due to betting margins. The model we chose was specifically designed to make predictions against the sportsbook in an effort to exploit underdog picks, and while its performance was reasonable with the last season’s data, it is hard to predict how well it could do in a live season. With the prevalence of sports betting, hopefully, our research will convince the public that the service should only be viewed as a source of entertainment and not an actual option for making a profit despite advertising and messaging from sports betting platforms.

Broader Dissemination Information:

Your report title and the list of team members will be published on the class website. Would you also like your pdf report to be published?
YES

If your answer to the above question is yes, are there any other links to github / youtube / blog post / project website that you would like to publish alongside the report? If so, list them here.

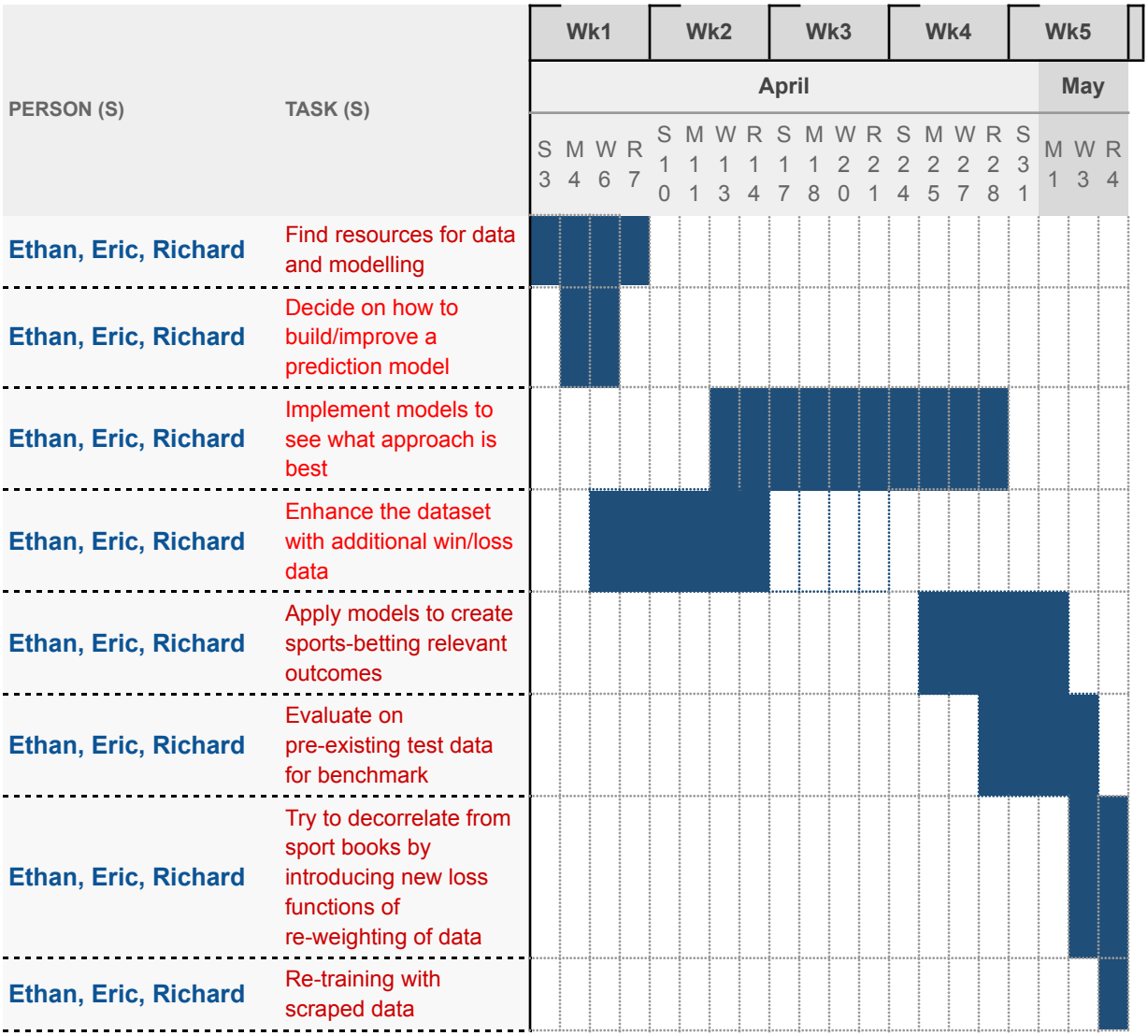
- **Dataset Generation Notebook:** Contains code necessary for cleaning, parsing, and augmenting our training and test data**.**

  https://colab.research.google.com/drive/1WyUTxmJNG31D1gzq4YUmS-weBPwkaYGv?usp=sharing

- **Model Evaluation Notebook:** Contains the actual code for initializing, training, testing, and evaluating our novel model.

  https://colab.research.google.com/drive/1b_QD-NCn8ar701dCNckUEb3qBATclJmD?usp=sharing

(Exempted from page limit) **Work Report: This may look like your GANTT chart from the midway report, with more completed steps now. Okay to modify.** (Mark completed steps in green, as shown here. For convenience, you may split into two charts, one till Nov 8, and another for after Nov 8, placed one below the other.)



(Exempted from page limit) Attach your midway report here, as a series of screenshots from Gradescope, starting with a screenshot of your main evaluation tab, and then screenshots of each page, including pdf comments. This is similar to how you were required to attach screenshots of the proposal in your midway report.

F

Frank Liu <liufrank@seas.upenn.edu> Sat, Apr 27, 3:05 AM

to Ethan, Chung, me

Hey guys,

Hopefully the final is going well. Since we met regularly, I know you guys are on track. Here are some of my comments:

1) It is not like you are incorrect, but random forests generally should perform better than logistic regression. Maybe play around with hyperparameters for random forests a bit, like the number of trees and others.
2) Very good work and clear structure. I can follow easily. Nice modification to address the comment last time.
3) I don't have other comments. Look forward to your final deliverables!

Let me know if you have any questions

Best,
Frank

•••

Beating the Book with Machine Learning

Team: Ethan Ma, Eric Wang, Chung Un Lee (Richard). **Project Mentor TA:** Frank Liu

1) Introduction

Inputs

- home_team, away_team, W, PTS, FGA, TPA, FTM, OREB, DREB, REB, AST, TOFV, STL, BLK, BLKA, PF, PFD, season, date, home_team, away_team, label

Outputs

- Learnability of logistic regression and random forest classifier using the input dataset
  - Logistic Regression
    - Training accuracy of Logistic Classifier: 62.3%
    - Testing accuracy of Logistic Classifier: 60.4%
  - Random Forest Classifer
    - Training accuracy of Random Forest Classifer: 68.4%
    - Testing accuracy of Random Forest Classifer: 59.4%
  - Top Features by Importance
    - OREB, PTS, FGA, TPA, FTM, TPM, FGM, REB, BLK, FTA
- New dataset
  - Based on the learnings, we generated a final dataset that combines input data
  - Features: home_team, away_team, W, PTS, FGA, TPA, FTM, OREB, DREB, REB, AST, TOFV, STL, BLK, BLKA, PF, PFD, season, date, home_team, away_team, label

Data:

- 2022-23 NBA regular season players data
  - Features: Rank, Player name, Team, GP, MIN, PTS, FGM, FGA, FG%, 3PM, 3PA, 3P%, FTM, FTA, FT%, OREB, DREB, REB, AST, STL, BLK, TOV, EFF
- 2022-23 NBA regular season teams data
  - Features: Rank, Team, GP, number of wins, number of loss, win %, MIN, PTS, FGM, FG%, 3PM, 3PA, 3P%, FTM, FTA, FT%, OREB, DREB, REB, AST, TOV, STL, BLK, BLKA, PF, PFD, +/-
- 2022-23 NBA regular season match win/loss data
  - Features: season type, status, date, home team score, home team, away team score, away team, winner, loser

Evaluation: The evaluation of success for our project is divided into two parts: evaluation of our model and evaluation of our contribution to existing data. For the former, our overarching goal is to get to the point where we can measure the profitability of our predictive model compared to the open betting market as our measure of success. However, because this may require some difficult integration with betting APIs, a simpler and more achievable metric for evaluation would simply be to measure the accuracy/profitability of our models on large historical sports-betting data sets when compared to the actual betting lines for those games.For evaluating the success of our data augmentation, we can train a baseline model on a historical dataset, and compare accuracy when using the same model trained on the same data set with our contributions of more modern data.

Motivation: There's been some recent concern about the increased prevalence of sports betting, which is now a hugely popular industry. I think that the most likely finding from our project will be the fact that trying to beat the book is inherently impossible, and should never be approached with any other mindset than the knowledge that engaging with betting sites is a guaranteed net loss. However, I do think that if we are successful in creating some novel loss functions, we can

reveal some interesting findings about the ways in which different models interact, and how loss functions be adjusted to efficiently and positively interact with the behaviors of other models. Hopefully this project will lessen the 'gamification' that sports betting has recently been taking on.

2) How We Have Addressed Feedback From the Proposal Evaluations

We received feedback that our TEP definitions are not comprehensive. To address this issue, we have created more detailed TEP definitions with the exact step of our method. To achieve our task of beating the book with machine learning, we first gathered data that contribute to the outcome of NBA matches. As the current season is still on-going, we decided to focus on previous season's data. This includes performance data of players and teams, and the win/loss outcome of each match. We then pre-processed the data to be used for classification models (logistic regression and random forest) and tested the learnability of the data. In the process, we identified the top features by importance and the accuracy of the models. Moving forward, we plan to enhance the dataset by adding match outcomes of all previous seasons and improve our model through PCA or hyperparameter tuning. We will then evaluate the performance by comparing the outcome with the book.

To address the fact that our contributions section was too vague, we added a more concrete and well-defined alternative method for evaluating our models on an existing data set in addition to the stretch goal of implementing a pipeline to actual make bets using our model and tracking the profitability as a possible metric. We also added some more detail on the potential contributions made on the data side.

We changed our model approach to focus on refining potential loss functions in order to efficiently make surprise profits against the likely prediction of the sportsbook. As a result, our models themselves are primarily simpler classifiers like logistic regression and decision tree methods like random forest.

3) Prior Work We are Closely Building From

A. Exploiting sports betting market using machine learning
   a. Link:https://www.researchgate.net/publication/331218530_Exploiting_sports-betting_market_using_machine_learning
   b. This is the main prior work that we are building from. The authors, Hubacek, Sourek, and Zelenzy, proposed that traditional machine learning methods in sports betting were ineffective, since at best they would correlate with the book maker's odds, making so that overtime the better would lose money proportional to the margin. Thus, they had come up with various ways to decorrelate with the bookmaker, such as introducing new loss functions that punished samples that were too closely correlated with the given lines.
B. https://github.com/NBA-Betting/NBA_Betting?tab=readme-ov-file#mldl-modeling
   a. This is very similar to our project's goals, however we want to implement more up-to-date data, as well as more specific recommendations in relation to game lines.

4) What We are Contributing

Our contributions lay in two primary areas:

1. Contributions to data: Because existing publicly available datasets such as those on kaggle are rarely updated frequently enough to capture recent match and player data, which is often the data most critical for making accurate predictions, our project aimed to augment existing datasets by scraping up-to-date data.

2. Model development: The sportsbook models of today are extremely accurate, and any given competing model will likely fail to be superior enough to make a profit when the costs associated with betting margins are factored in. As a result, our model can't simply emulate existing books.

5) Detailed Description of Each Proposed Contribution, Progress Towards It, and Any Difficulties Encountered So Far

5.1 Methods

While we have yet to make significant progress in the development of a complete novel loss function, most of our initial progress has revolved around curating a sufficiently large and feature-rich data-set with which we can augment existing data which often lacks recent results. This is particularly critical because outdated data is rarely useful in accurately predicting outcomes in the world of sports because of the large amount of churn experienced by professional sports teams. Having established some working baseline models in a colab notebook, and conducted analysis on our PCA to determine our most critical features and an ideal number of components, we're currently working on testing modifications to error functions and evaluating their performance on our augmented data set.

In terms of gathering data, research suggests that the choice of features is more important than the choice of model itself [Zimmermann, Moorthy, and Shi; 2013]. Knowing this, we wanted to provide an update-do-date dataset with features backed by research (listed above, and choice of features listed as well in the Hubacek, Sourek, and Zelenzy paper). To do this, data from the 2023 and 2024 (still on-going) seasons were scraped, and then the results were cleaned, combined, and labeled to create new .csv files that are ML-ready.

5.2 Experiments and Results

Data Augmentation:

Our team was able to create ML-ready data, below is what part of a row of data looks like in our results:

	home_team	away_team	W	PTS	FGM	FGA	TPA	TPM	FTM	FTA
2	MIL	BOS	1	-1.0	0.5	1.6000000000000100	-2.3000000000000000	-1.2000000000000000	-0.8999999999999990	0.7999999999999970

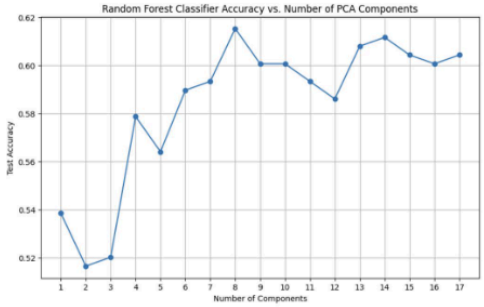
This data was scraped from the official NBA website via pages on player stats and comprehensive match histories. The scraping itself was done via selenium. Since each game has 2 teams, data was labeled 1 if the home-team won, and 0 if the away-team won. Each final feature was also the difference between the home-team statistics and the away-team statistics.

Baseline Models:
To see how ML-ready our data was, we wanted to create baseline models that we could use to not only evaluate our data, but also the feasibility of our chosen features. The results are listed in the introduction.
PCA/Feature Analysis:

```
Top Features by Importance:
OREB: 0.1216
PTS: 0.0868
```

```
FGA: 0.0831
TPA: 0.0816
FTM: 0.0718
TPM: 0.0564
FGM: 0.0561
REB: 0.0512
BLK: 0.0506
FTA: 0.0484
```

In the information above it's important to note that these are the top features ranked by importance after the W(wins) feature was removed, which had by far the largest ranked importance by an order of magnitude. This is understandable since teams which have won previously are likely to win subsequently, but it does raise some questions about the ability of our model to generalize.



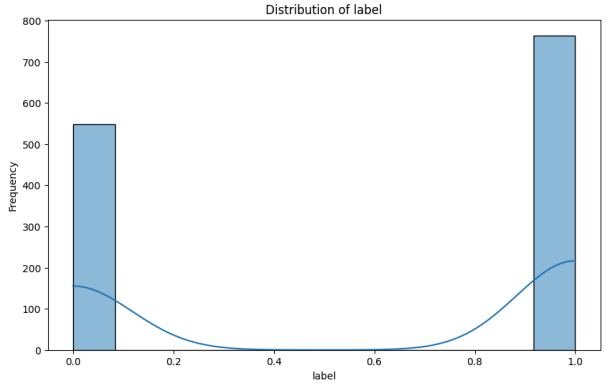
Here, it seems like 8 components resulted in the largest accuracy gains, though overall accuracy remained fairly unimpressive around 62% with a baseline random forest classifier.

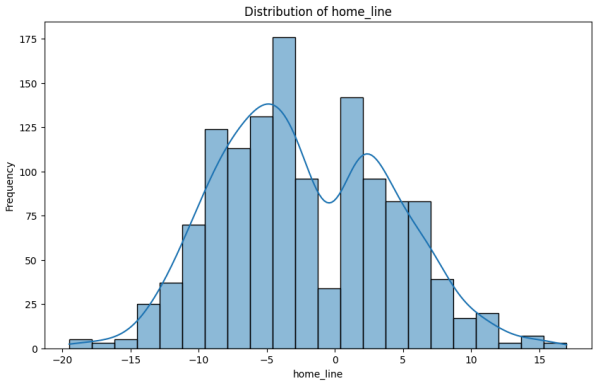
6) Risk Mitigation Plan

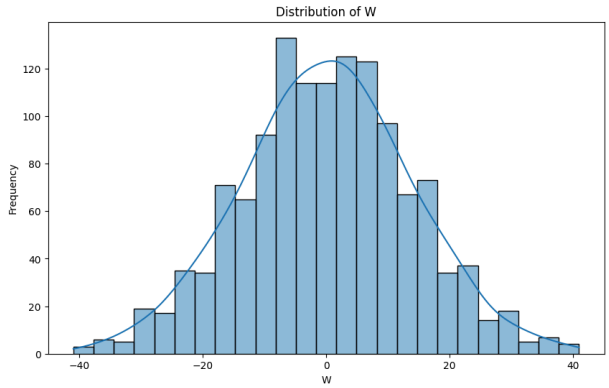
From this point onwards, most of our effort will be towards taking our baseline model and transforming it into a useful tool for placing bets. The baseline model itself just predicts the probability of winning games given a set of features/team-based statistics. From here, we want to apply more information on the game lines (+/- scores) to our model to create relevant statistics that can be used for placing bets. For example, using the given game lines to more heavily favor the games where the line was wrong. In order to mitigate risk, we will add features incrementally. First we will apply more in-depth statistics in our model (taking into account more player-centric data, adding more years of game data to our training set, etc.) Then, we will try to apply our in-class Logistic Regression implementation to allow for customization of the loss function (i.e. increasing loss for games too closely correlated with the book maker's line). Finally, we will simulate actually betting on games. At any point, as long as we have finished the first point of this iterative process, we will have some minimal working product. In the case that our model is not good at predicting games or does not give good results, we can still draw conclusions. Some conclusions can be, what are relevant features to pay attention to while placing sports bets? Is it a good idea at all to bet on game lines? What extra features would have been helpful? Etc.

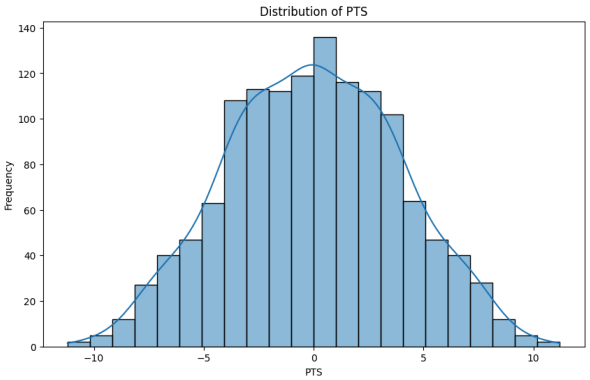
Appendix

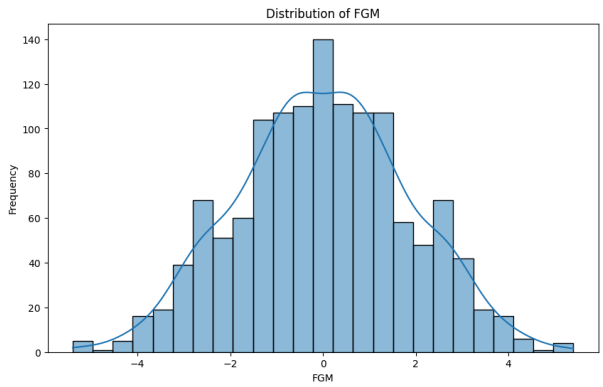
Appendix A:

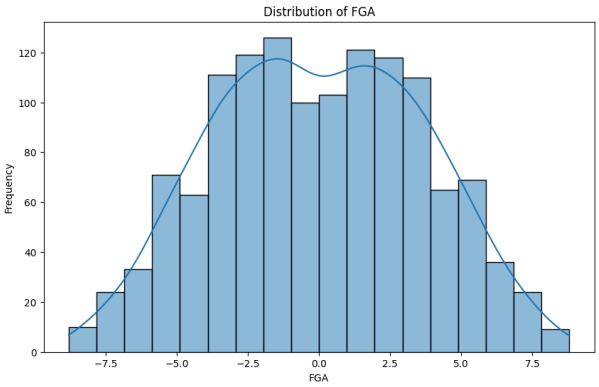


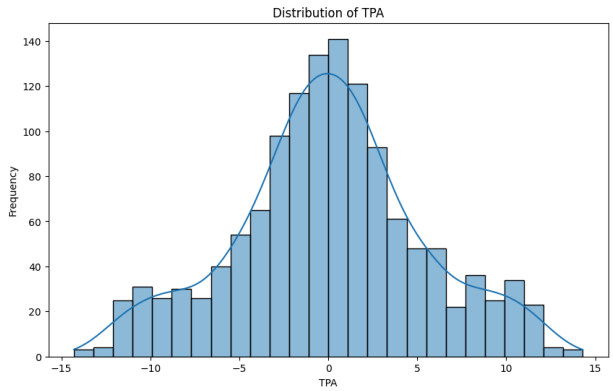


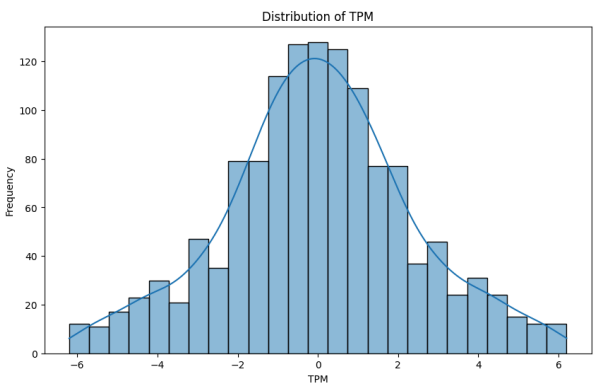


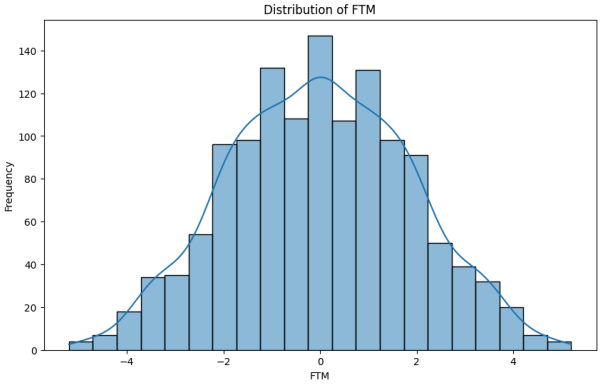


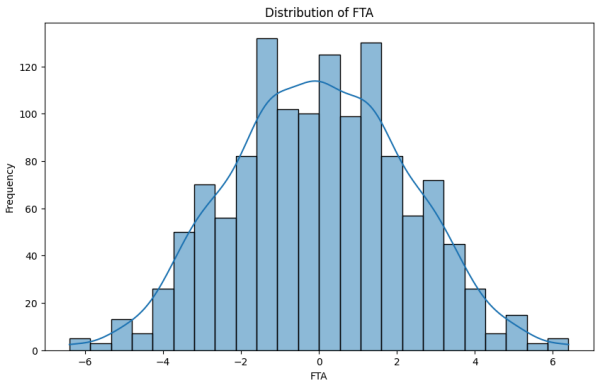


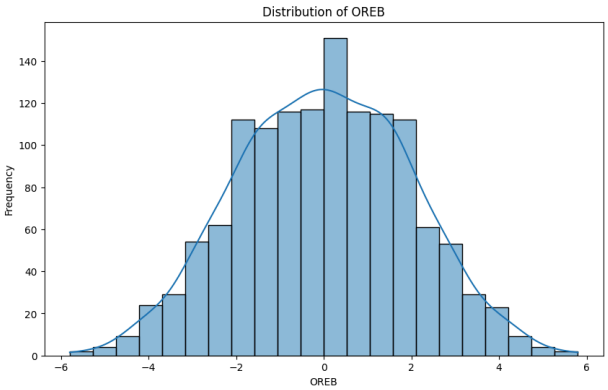


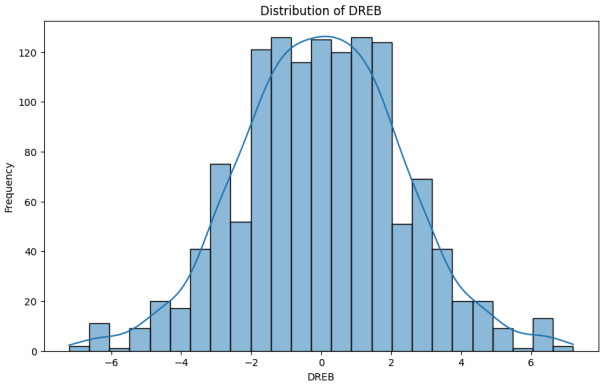


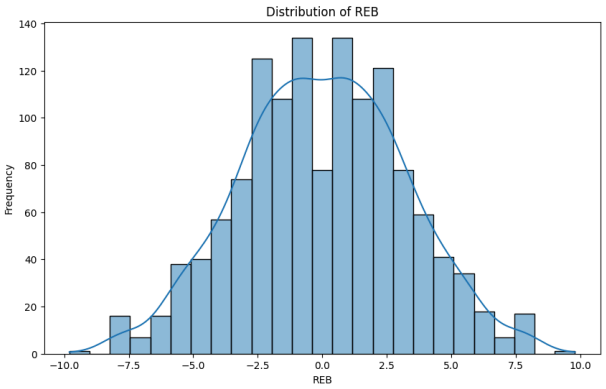


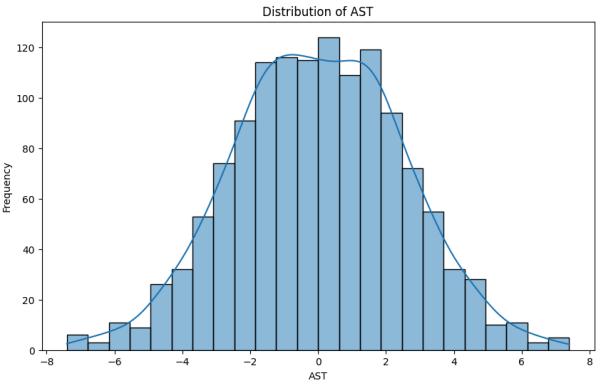

Distribution of FTM
Frequency
FTM

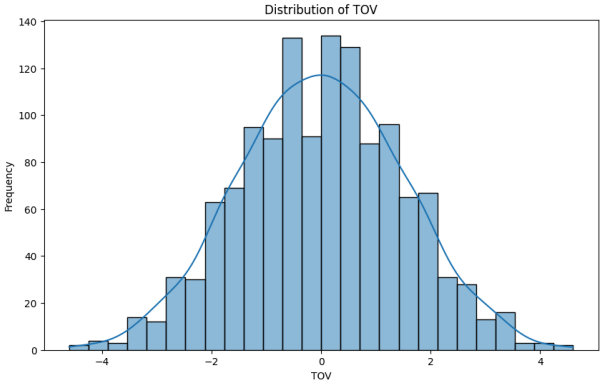

Distribution of FTA
Frequency
FTA

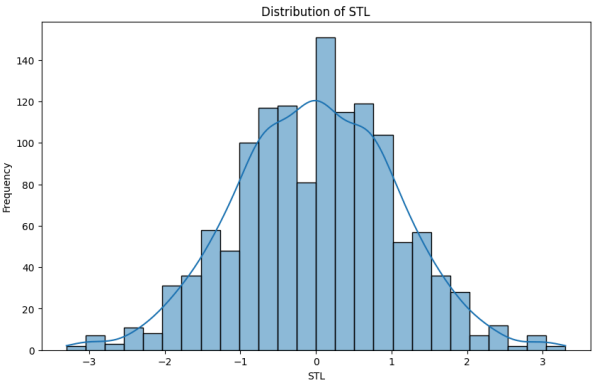

Distribution of OREB
Frequency
OREB

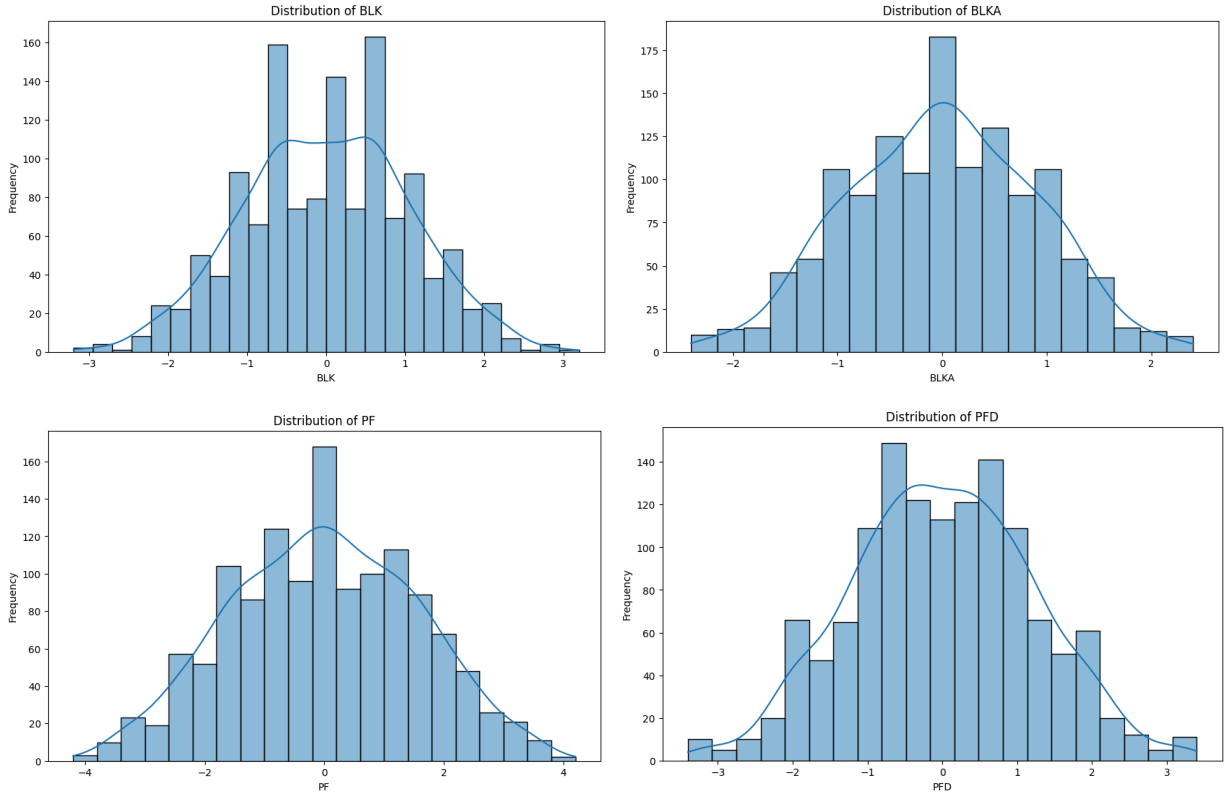

Distribution of DREB
Frequency
DREB


Distribution of REB
Frequency
REB


Distribution of AST
Frequency
AST


Distribution of TOV
Frequency
TOV


Distribution of STL
Frequency
STL



Figures 1-20: Distribution of Final Dataset

Appendix B:

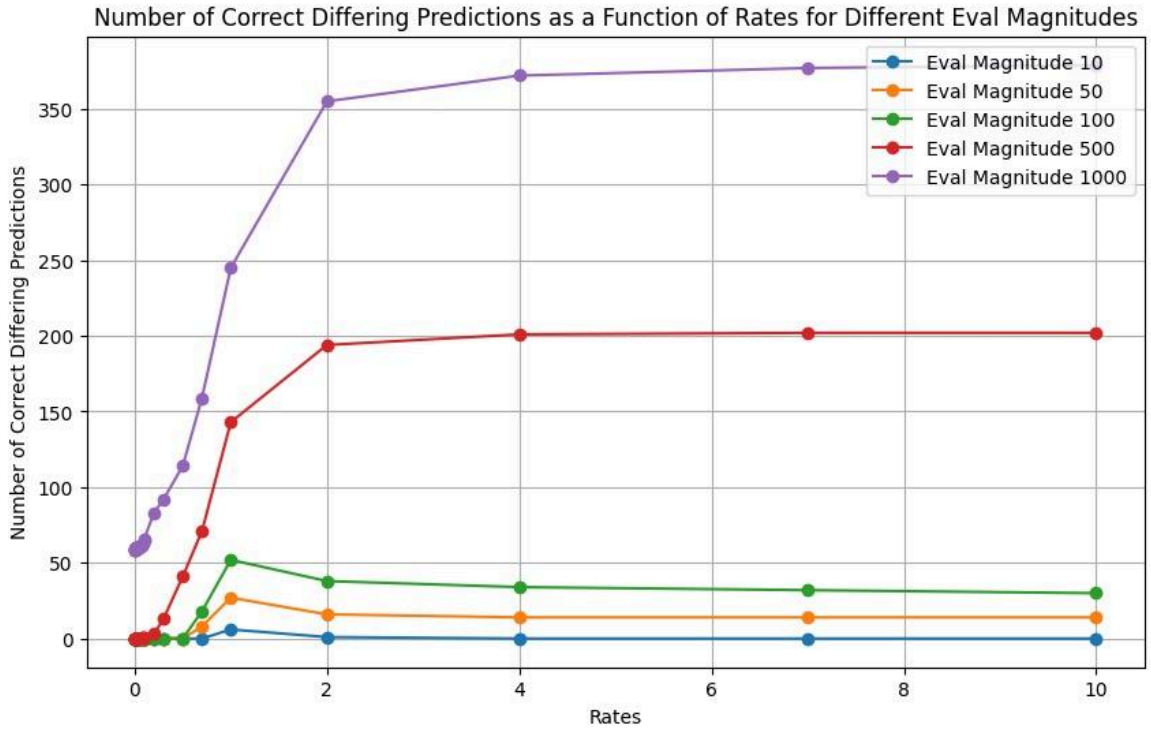


Figure 21

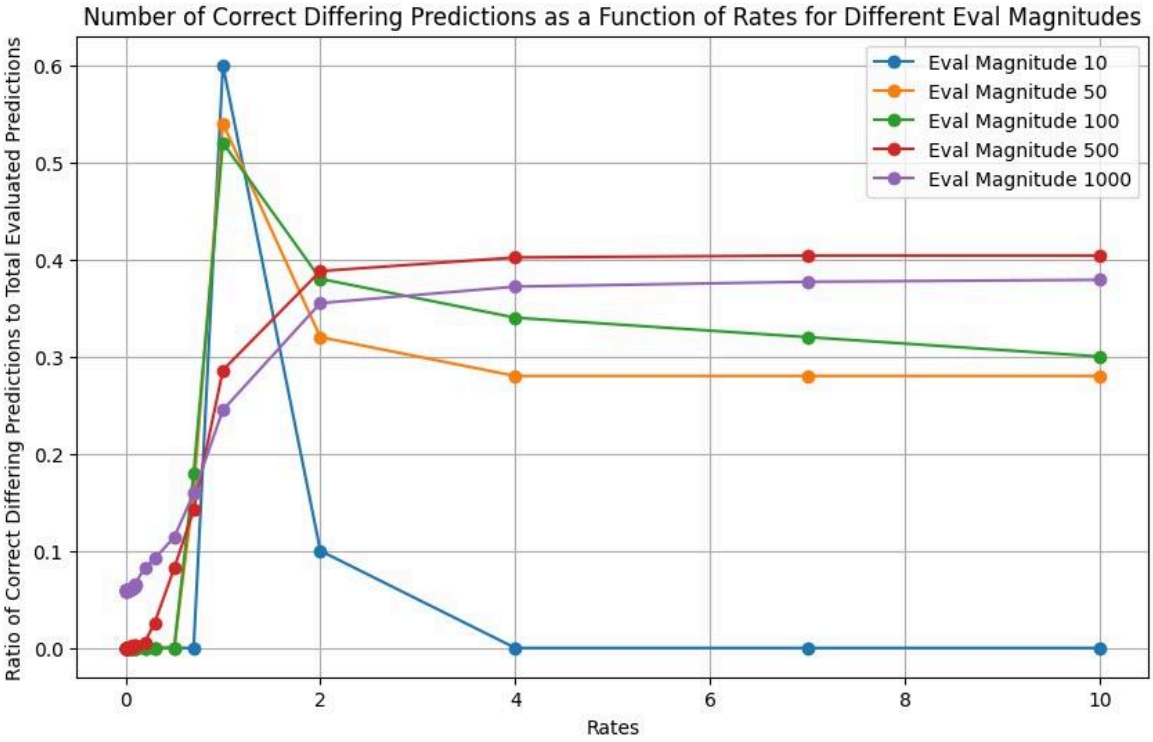


Figure 22

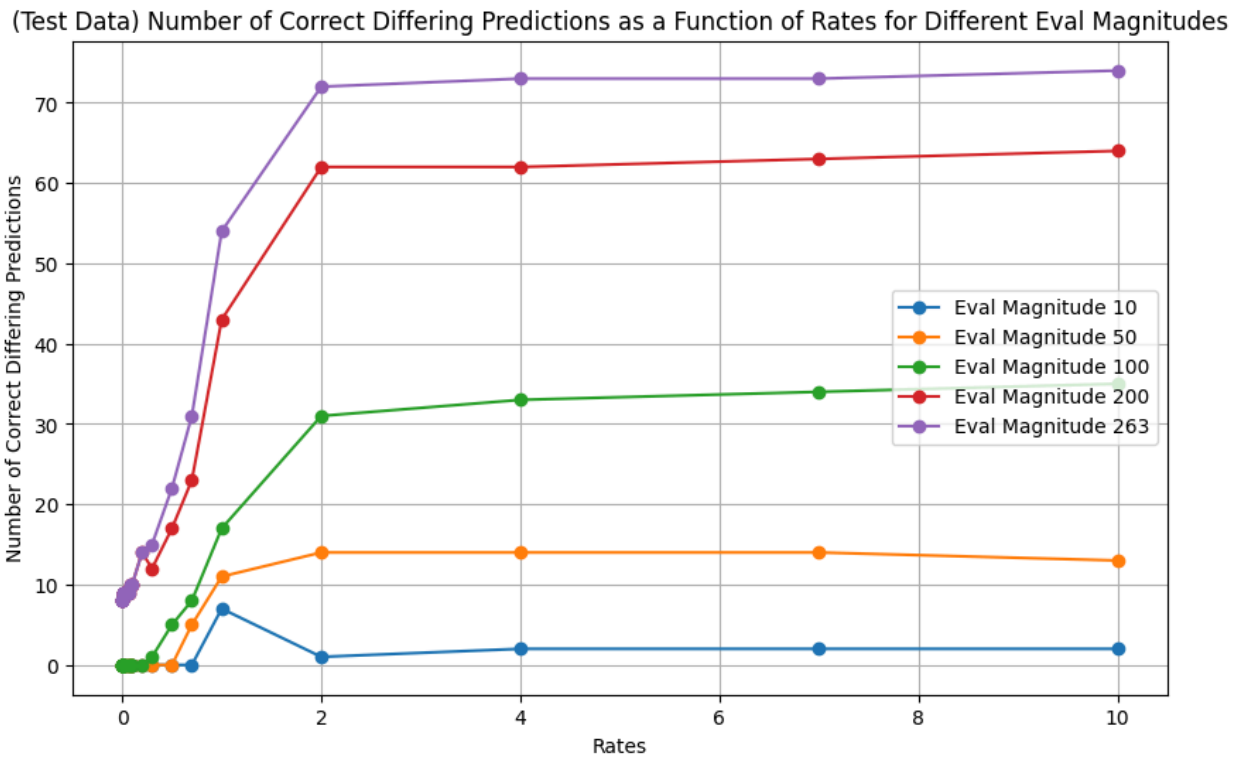


Figure 23

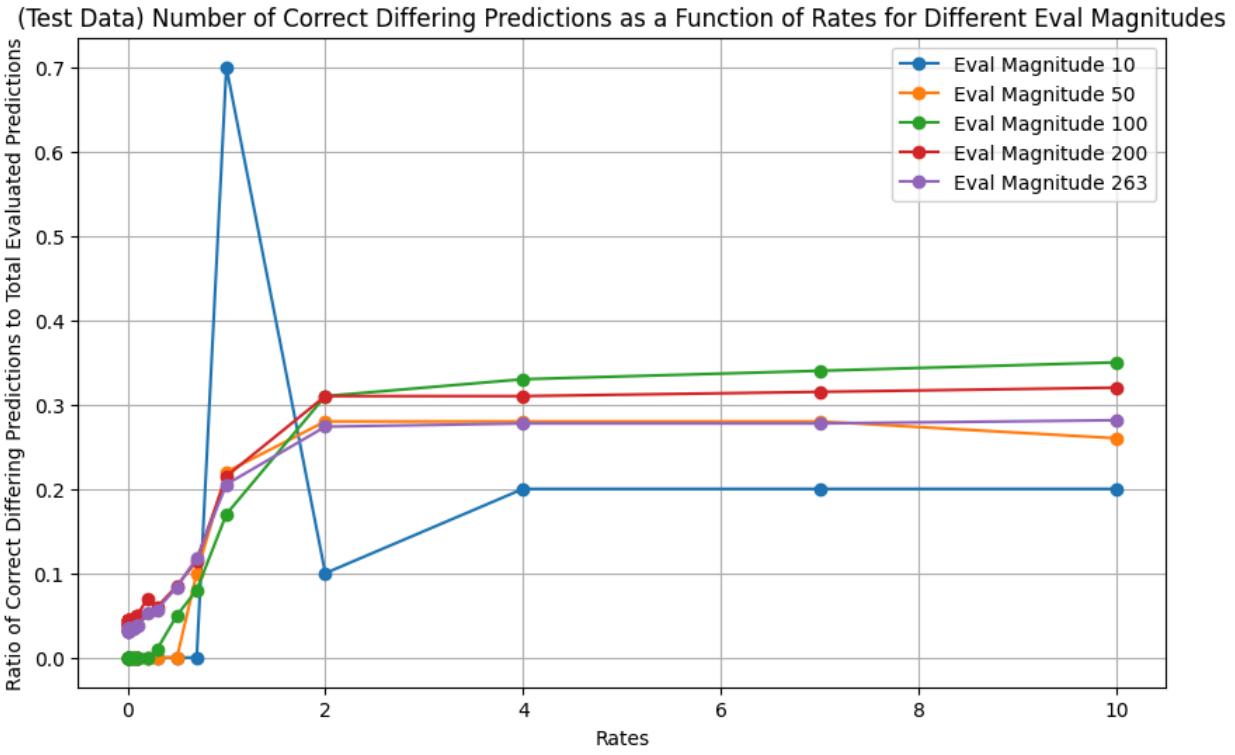


Figure 24

(Exempted from page limit) Other Prior Work / References (apart from Sec 3) that are cited in the text:

1. Koushik. “Logistic Regression from Scratch.” *Medium*, Medium, 28 Aug. 2023, medium.com/@koushikkushal95/logistic-regression-from-scratch-dfb8527a4226
2. *Zimmermann, A., Moorthy, S., & Shi, Z. (2013). Predicting college basketball match outcomes using machine learning techniques: some results and lessons learned. arXiv preprint arXiv:1310.3607 , .*