# Data Quality Analysis in Sports Betting Odds

Group 7: Avi Bewtra, RJ Krimetz, Sarvar Khamidov

## Introduction

Since sports betting was legalized four years ago, Americans have wagered more than $125 Billion USD, with annual betting participation increasing rapidly. On the surface sports betting odds appear to move like any time series financial data. However research has found that there is a notable elasticity in sports bettors who consistently cluster bets behind the favorite despite unfavorable movement in the odds. This has led oddsmakers to boost profits by setting odds off of the ‘market clearing price’ more commonly observed in other financial instruments. Accordingly, preliminary backtests reveal that there are betting strategies capable of systematically exploiting these intentionally ‘inaccurate’ odds set by most bookmakers.

Detailed sports data is freely available on the internet, but historical odds data is surprisingly difficult to find, with no APIs offering historical betting odds and game statistics in a usable format. In this project we aim to collect data for a variety of sources containing both historical odds and team statistics. Then we aim to perturb copies of our dataset such that its quality deteriorates in the few key metrics selected. With these datasets—the baseline and the perturbed—we aim to compare the quality of model prediction we are able to achieve. In other words, in the domain of sports betting, we hope to discover which key data quality metrics are most important. These results provide clarity on what data quality metrics are most useful in looking for systematic mispricing in the sports betting market. To accomplish this we are going to look to predict the odds of NBA games on a series of perturbed datasets to assess which data quality metric is most influential in our predictions.

The team was able to calculate different data perturbations for the sports betting dataset. The row uniformity performed identical to the model. Adding noise significantly degraded the model performance but boots-strap sampled features increased it. After dropping features or adding dummy features to our data we observed that performance drastically declined.

## Related Work

Most relevant to our research is a data quality piece written by the previous DS 440 class. Citing last year’s data quality project report, data quality metrics of interest include: accuracy, column to row ratio, conformity, distribution coverage, file size, timeliness, and row uniqueness. The group did comprehensive studies on these seven metrics, and their relevance to analysis. In academic research, *Model Cards for Model Reporting* by M. Mitchell et al., was an excellent insight into model cards, which help compare models after training to assess quality. *The Dataset Nutrition Label: A Framework To Drive Higher Data Quality Standards*, by Holland et. al. was an excellent coverage of the state of the art in dataset health; their measurement depends on an investigation dataset metadata, provenance, variables, statistics, pair plots, probabilistic models, and ground truth correlations.

On the modeling of sports betting odds, it has been shown that sports book makers do not try to more accurately predict who will win a game, instead they focus on predicting what percentage of betters will choose the favorite irrespective of the odds (Levitt 2004). In other words they are setting the odds of a bet based on the elasticity of sports bettors, and in theory, their behavior is predictable if you know the volume of betters on each side of the trade. This means that favorite teams beat their spreads more than fifty percent of the time, however if a higher percentage of bettors put their money on the favorite the system is still profitable for the bookmaker. Levitt describes how the price of a bet can be predicted based upon three variables: *p*, the estimated probability of the frontrunner winning; *f*,the fraction of total dollars bet on the frontrunner; and *v*,the percent commission taken by the bookkeeper. However, the problem in calculating these three variables—*p*, *f*, and *v*—is a lack of data. This implies that odds are systematically inaccurate. The book keepers aren’t concerned with setting a fair price for the bet. They are concerned with maximizing their profit in the following expectation:



This formula calculates the expected profit by summing the bookkeepers expected earnings and subtracting the expected costs. In order to maximize the expected profit, the bookkeeper would need to use the bettings odds (or prices); because *f* is a function of the bettings odds—meaning that as the odds changes, the percentage of bettors on each side does too—the expectation shown above becomes a function of the betting odds.

The other prominent body of published work focuses on predicting winners and losers irrespective of the spread. Notable feature of most of the prior work relevant to our exploration of data quality is that not many researchers use API/live feed data. This limits the application of their research and raises feasibility concerns about replication and implementation built on un-recreatable datasets, further increasing the motive to look into data quality metrics.

On other notable takeaways, Neural Networks are the most common Machine Learning approaches used in this field. Most previous research looks to solve a binary classification problem (Win/Loss), however the few that look to predict odds directly appear to be more accurate. Player statistics have been used as features across almost all research however none. The most commonly cited work has looked to predict odds on player specific lines.

## Datasets

The team was able to compile a comprehensive datasets from a variety of sources to build a model for predicting the odds.This data set is built from the following sources described below. However, the data that was used for the project’s model was Kaggle dataset:

[The Kaggle Dataset ‘NBA Historical Stats and Betting Data’](https://colab.research.google.com/drive/1EKZXzPZMpxa2W4R6dyXCtuo6lDd2NlqN?usp=sharing) contains player data and team data across 15,000 games, along with betting data—including money line odds and betting spreads across 10 sports books. We will be able to use this data to measure the distribution of team-odds across seasons and different sportsbooks. Based on initial investigation, the data is observably clean, complete, and accurate. However the data isn’t perfectly timely; it only captures years up to 2018, meaning it’s a few years old. Also, this dataset is static—not a live API, so it cannot be easily updated. This data includes game data—teams, location, each team's win-loss record, points per quarter, assists, rebounds, and turnovers—over 6 seasons. For each game, it also includes the odds for money-line bets, point-spread bets, and over-under bets for six different casinos. Our code showing samples of the dataset is linked below in references.

The team was able to pull, clean and analyze other datasets from various sources. We have created a web scraping tool to compile data from Team Rankings website, as well made scripts to pull the data from 2 APIs. All those data were cleaned and formed into the proper format for the model. But they were not used in the project since the datasets were not enough for the final model. The project required historical data of odds and stats for the game, but these sources did not provide them fully.

1. [The Odds API](https://colab.research.google.com/drive/1rIHf0832eVcr61oJZ6yE4yOmSKikAkZq?usp=sharing) has real time odds for several bookmakers including US (DraftKings, FanDuel, BetMGM, Bovada, MyBookie.ag, Intertops), UK (Unibet, William Hill, Ladbrokes, Betfair, Bet Victor, Paddy Power), EU and Australia. It provides data for several different sports such as Basketball, Football, Soccer, Ice Hockey, etc. It also includes different types of bet types such as head-to-head, point spread, totals, outright, etc. Sports odds data is no older than a few seconds to a few minutes. The data is constantly being updated for the upcoming games.
2. The group was able to get more data from the [Team Rankings website](https://colab.research.google.com/drive/1x-QYt09evA3rErTDd5F9C-6uHygqTJeR?usp=sharing). The source has historical statistics for teams and for individual players in the NBA for the last 20 years. The team's stats include various types of data, such as points per game, shooting percentage, rebounds, blocks, steals, fouls, assists and many others for each game week. The individual player statistics also have points, assists, rebounds, steals, blocks and other. The sample of web scraping of this data into proper format is in the link below:
3. Finally, for the NBA season starting on October 18th the group has access to the [APISports NBA pipeline.](https://colab.research.google.com/drive/12bpwh7HZcFL9BS290bGiUJD-l-tI-2mF?usp=sharing) This API provides a variety of data tables from team statistics to odds breakdowns for all bookmakers. However this dataset does not provide any historical data, but the live feed can be used going forward to expand the current dataset for forward looking predictions. The specific bets included in the data schema include 3Way Result, Home/Away, Over/Under/Highest Scoring Half, Double Chance, Over/Under (quarterly, and entire game), Odd/Even, etc. Finally, the team stats include the league, team, and relevant game stats for both home/away benches.

## 

## Methods

#### *Dataset Perturbations*

We plan to attack specific quality metrics of our dataset, then compare it to the original. We will train models to predict odds based on each dataset and compare the accuracy metrics. To ensure a thorough test, we will train an ensemble of models, making sure to account for which types of models might perform better on certain datasets.

We aim to perturb our data quality metrics:

1. \*\*Row Uniformity: Take out a percentage of the data and replace the removed rows with random duplicates
2. \*\*Timeliness: Predict a years/games odds based on data from ***two*** years ago, not the year prior—or something to this effect.
3. \*\*Feature Statistics:
   1. Create bootstrapped synthetic datasets using the statistics of each feature: mean, variance, min, max, etc; or use another probabilistic model.
   2. Create bootstrapped synthetic datasets using ***changed*** statistics of each feature: mean, variance, min, max, etc; or use another probabilistic model.
4. Distribution Coverage: Remove certain portions of the dataset targeting certain portions of the distributions. i.e. What happens when the dataset is imbalanced?
5. \*\*Drop features
6. \*\*Add dummy features

\*\* Indicates currently implemented perturbations

#### *Basic Sports Betting Model*

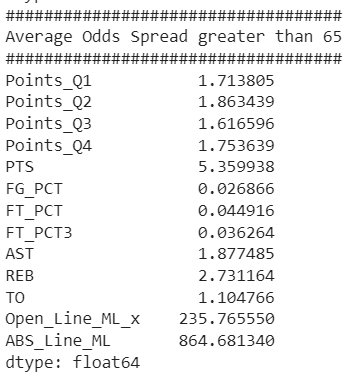
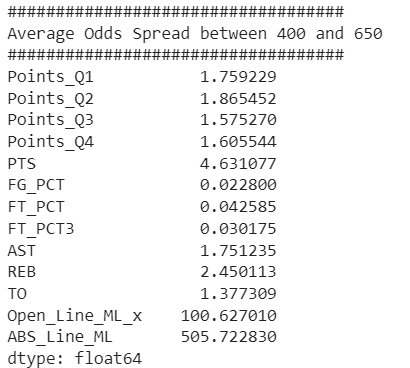
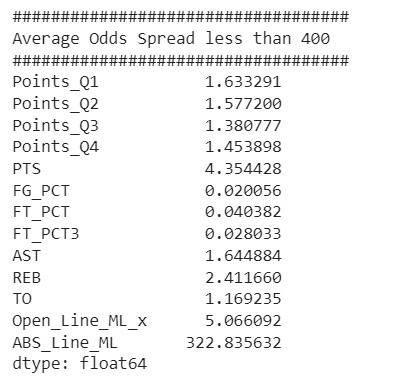
Since the primary focus of the project is to measure what aspects of data quality are most important in predicting sports betting odds, we need to build a variety of models on the multiple versions of the data to see what versions yield the best cross sectional results. Accordingly AutoML packages such as FLAML and PyCarret can be used to build a number of models on each dataset. We plan on implementing multiple regression models in our testing ensemble: linear and polynomial regression, SVM, Decision Trees, and Ridge Regression. The models will be trained on team/player specific data from different holdout periods before the target prediction data. The model will then be asked to predict the odds for the game and the prediction will be compared to the odds published by the bookmakers. The difference in the prediction versions of the bookmakers odds will then be compared to the result of the game to see if the lines our models predicted accurately exploit the bookmakers’ crowding bias. Through measurements of model success—using Mean Squared Error and Mean Percentage Error—we hope to understand the following at the end of these experiments: 1) which data quality metrics are most valuable when training sports betting models; and 2) which models work best with different types of poor quality data.

#### *Preprocessing*

Before getting into the results of the models from the perturbed datasets we did some feature space cleaning. First, we wanted each line in our dataset to represent one game, with the stats and odds for both teams included in one line. Accordingly we needed to combine the odds to create our target column. Since odds are both positive and negative, we took the sum of the absolute value of the odds, making the domain of our target space positive values. The larger this value corresponded to a more heavily supported favorite.(ie. Favorite odds -450 versus underdog odds +400 - resulting in a target column of 450 + 400 = 950).

Accordingly, we looked at each team that was playing and calculated the average points per game, average rebounds per game, average turnover per game, average field goal percentage per game,average three point percentage per game, etc. over all previous games in the dataset. We then took the absolute value of the difference in averages between the two teams.(ie. Favorite team’s average points per game 85, underdogs average points per game 67, abs(85 - 67) = 18). The intuition being that games where there is a higher differential between the teams average stats are likely the games where there is a larger spread in the odds, in other words, one team is more heavily favored than the other.

After further exploratory data analysis, this intuition appears to hold fairly well.



## Evaluation

#### *Data Perturbations*

<https://colab.research.google.com/drive/1EsML6MiZjHHhKDY9tWCjaXiXs12NoYc1?usp=sharing>

The notebook above contains implementations of functions that perturb a dataset. It contains functions that do the following to a dataframe: add duplicate rows; use untimely data; create bootstrap-sampled synthetic data from our dataset mean and std; add noise; drop features; and add dummy features. While we could write more, we think our code can explain our implementations better than we can. We’ve finalized our perturbation implementations and saved copies of the perturbed data for odds modeling. Below, is a list of each copy of our dataset, and the perturbations applied to it:

| ID | Perturbation | Description | Parameters |
| --- | --- | --- | --- |
| 1 | Row Uniformity | Add p \* n duplicate rows | p = .25 |
| 2 | Add Noise | Add noise to each column | Noise: and . |
| 3 | Timeliness | Utilize old data | Only use data that is more than 1 year old |
| 4 | Feature Statistics | Generate a new feature sampled randomly | and from the original feature |
| 5 | Drop Features | Drop n columns | N = 2; column selected randomly |
| 6 | Add Dummy Features | Add n features sampled from random noise | N = 3; and . |

Modeling on these datasets will inform us how each key data quality metric impacts accuracy and data utility. While this is the focus of our project, we have spent time this week trying to first create the best model possible on the original, clean, benchmark dataset.

#### *Odds Modeling*

<https://colab.research.google.com/drive/1HxvLl6sJpfsx4LP-zmbTtmHaBpOcMgWB?usp=sharing>

We’ve implemented an initial model to predict odds on our Kaggle dataset. This initial regression model predicts the odds for a *single* team based on that team’s data. We’ve developed a second model where we can predict the disparity in two teams' odds. Instead of:, we will model , where and are respectively the team data and odds for team in game . Now, instead of predicting the team odds one at a time, we can predict the odds of each two-team pair for a given game. Here, for our regression label, , a negative difference in odds would mean the first team is favorite, and for a positive difference the second team is the favorite. Also, as the absolute value of the difference gets larger, the favorite team is more strongly favored to win.

#### *Auto Machine Learning:*

<https://colab.research.google.com/drive/1nT2TaQUqLaA7DMz1uK4a-aGPkWDUJaHe?usp=sharing>

We then fit a variety models on our perturbed datasets, to different data quality features to see how different quality metrics affected the results of Auto Machine Learning Applications.

To do this we created a PyCaret pipeline to feed the various perturbed datasets through. The results of the pipeline are recorded in the results section.

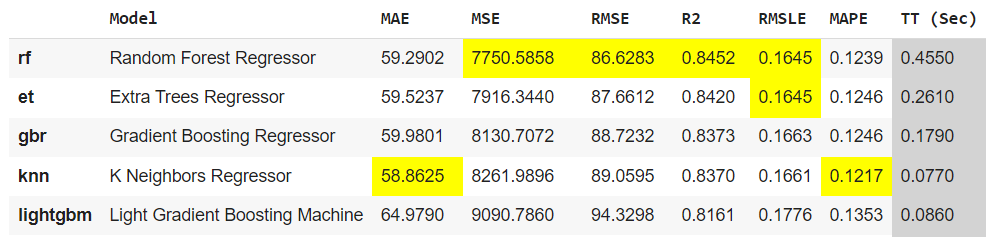
#### *Attempts to Memorize the Data Distribution*

In order to confirm that the mapping from our features to labels was learnable, we attempted to overfit our dataset with an MLP. With a 10,000 neuron-deep by 10,000 layer wide neural network, training of an MLP converged to an R^2 score of only 0.16. Even without more thorough feature engineering, we would expect a non-regularized, enormous neural network to be able to memorize the mapping from our features to our labels. But this was not the case. We attribute this to the fact that multiple labels in our dataset correspond to very similar inputs. As a result, in a regression problem, it would be impossible to model a function that maps one input to multiple outputs.

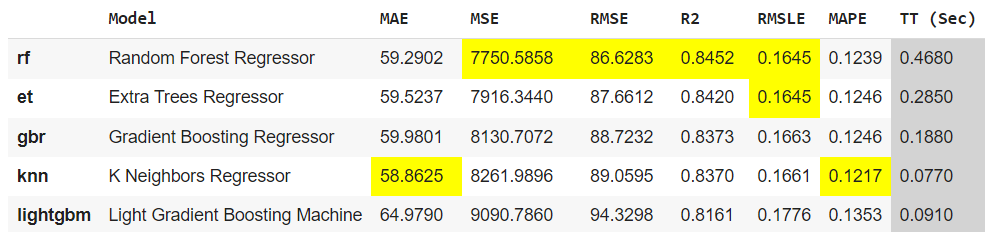
## Results

See results of our model training on each perturbed dataset below:

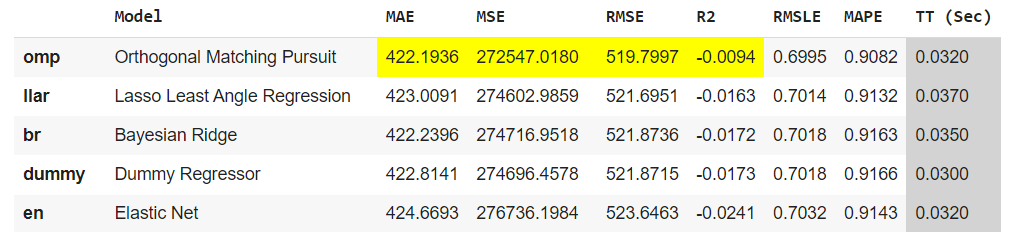
*Full Dataset:*



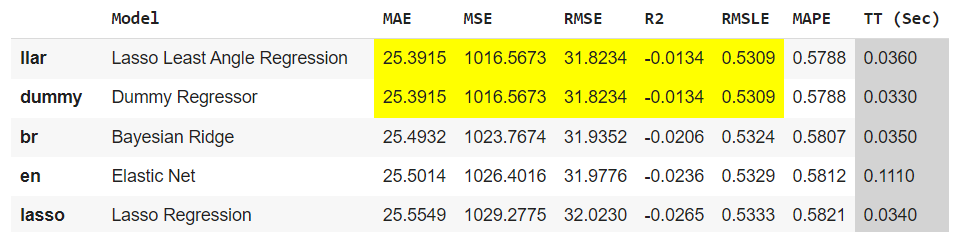
*Duplicate rows:*

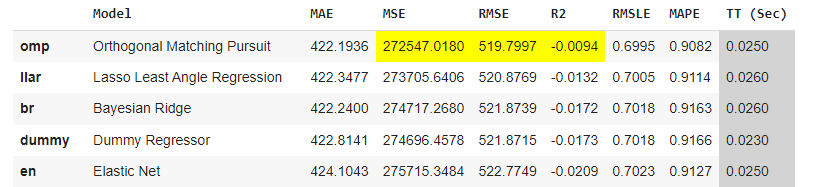


*Random Noise*

**

*Bootstrapping*

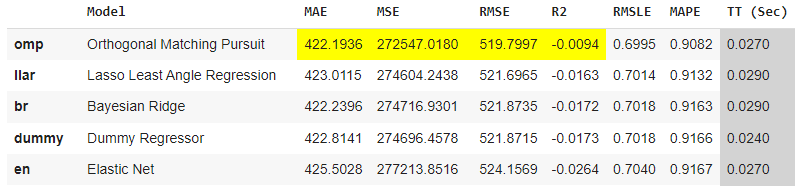
**

*Drop Features*

#### 

#### 

#### *Add Dummy Columns*



## Discussion

Below we will share a discussion of the effects of each dataset perturbation on the model.

1. Duplicate Rows: A dataset with duplicate rows performed identically to the baseline model. In our investigation, we found that all models tested were able to ignore duplicate rows and converge to the same optimal solution. While we believe that this might not be true for every model, but for the ones we tested, our dataset was large enough that additional duplicate points did not affect results.
2. Add Noise: The addition of random noise to our dataset significantly degraded our model’s performance. After experimenting with the amount of noise added, we have two possible pieces of intuition as to why: first, we are altering the distribution of the dataset; or second, our models are all overfitting the distribution, and adding noise prevents it from doing that. We know that in practice, adding noise is a data augmentation technique used as a form of implicit regularization; but, because of the fact that the increase in MSE is so drastic, we do not believe the noise is acting in this manner. Instead in this case, because the distribution of odds and team-data is not very clean, our first conclusion seems more reasonable; the noise is altering the data distribution and corrupting the model.
3. Bootstrap-sampled features: When we sampled features from a bootstrap distribution using the original feature’s mean and standard deviation, our model’s performance increased significantly. However, we found our model failed through a mode collapse. Our best models returned a very small range of values, where error was smallest—essentially ignoring many features and outputting similar odds for every matchup. As a result, bootstrap sampling features rendered our model useless.
4. Drop Features: After dropping ‘PTS\_Q4’ we saw a sharp decline in model performance. In general, avg. season points were some of our most important features, and as a result, dropping a whole quarter of that information drastically reduced our models ability to predict odds.
5. Add Dummy Features: Adding dummy features also radically affects our model. As of writing this report, we have no explanation. In the future, we would like to continue exploring the purpose for the magnitude of the effect on model performance.

## Challenges

One of the challenges that we faced with the original mode was low accuracy prediction with an MSE of around 50,000. Historically, sports betting odds has been a difficult thing to predict—similar to the stock market. However, after further examination we bevel the people had to do with the scale of the values relative to the range of the target columns.

## Conclusion

Over the last four months, we reviewed the most important data quality metrics; collected a dataset of team performance information and bookkeeper odds; trained machine learning models on datasets with each identified data perturbation; and compared our results.

Invariably, we found that, with respect to sports betting, degrading data quality metrics that have an effect on the distribution of the data—random noise, dropping features, bootstrap sampling—can nearly destroy model performance. Other data quality characteristics—adding dummy features and row uniformity—do not have nearly the effect on model performance. We believe that by changing the distribution of the data, model performance will collapse. A model will still be able to learn an optimal mapping of features to labels given perturbations that change the dataset without affecting the overarching distribution.

Throughout the semester, we also re-learned the age-old data science lesson: data collection is critical to the success of a machine learning model. We also learned that some patterns can not be easily learned, and that there is a reason sports-betting has not been a problem previously solved by machine learning. With reckless abandon, we assumed we would be able to throw together a model just good enough to analyze how data quality would affect results. We tried very hard for limited results. In earnest, our benchmark dataset just wasn’t good enough; which led to limited success to build a successful benchmark model; which led to very volatile effects to our already high error measurements. We’ve learned a lot of technical lessons, the most important of which may be that data is, in fact, king.

## Future Work

*Prepossessing:*

1. We are going to create sliding window averages, by generating features by taking the average of team data in games k-11 through k-1 in order to predict odds for game k. While we have had little success modeling odds, we have found better results with season-cumulative-average-features. As a result, we have elected to continue modeling with these features.
2. Look to remove the Open\_column\_odds\_x feature by somehow scaling the target columns

*Application:*

1. Using the current models, times when our predicted spreads are larger than those observed in the betting market is interpreted in our model as being more confident that the favorite will win than the betting market and vice versa. Recall from our background research that there are systematic reasons why these mis-pricings would exist, so the next step is to see how often our model ‘prefers’ the winner and if the magnitude of the preference has any correlation to confidence in who will win.

## Individual Contributions

Robert Krimetz: Robert Krimetz: Currently a senior studying Data Science and Economics. I am primarily interested in Quantitative finance and statistical arbitrage. Most of my experience is with time series analysis of US equities. I worked on building the machine learning pipeline for the various data perturbations and cleaned the feature space.

Avi Bewtra: I am a senior studying Computational Data Science. I am interested in deep learning theory and systems. I am strongest in these areas, mostly using Python. I researched all of the data quality metrics we used, basing our approach off of *The* *Dataset Nutrition Label*, and last year's group’s investigation into data quality metrics. Applying these to the problem of sports betting, I continued to implement them in [code](https://colab.research.google.com/drive/1EsML6MiZjHHhKDY9tWCjaXiXs12NoYc1?usp=sharing) and save copies of our dataset that could be fed into our models. I also collaborated in the conceptualization and design of our machine learning problem and the data cleaning.

Sarvar Khamidov: I am a senior student majoring in Applied Data Sciences with an Information Sciences and Technology minor. Interested in learning about different machine learning algorithms and finding new applications for it. I mostly worked on data pulling and data cleaning. However, I have also helped with other parts of the projects such as machine learning and data perturbations.

## References

Guardian News and Media. (2022, May 13). *Americans have bet $125BN on sports in four years since legalization*. The Guardian. Retrieved October 9, 2022, from https://www.theguardian.com/sport/2022/may/13/sports-gambling-125-billion-anniversary

Holland, S., Hosny, A., Newman, S., Joseph, J., & Chmielinski, K. (2018, May 9). *The Dataset Nutrition Label: A Framework to drive higher data quality standards*. arXiv.org. Retrieved October 9, 2022, from https://arxiv.org/abs/1805.03677

Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I. D., & Gebru, T. (2019, January 14). *Model cards for Model Reporting*. arXiv.org. Retrieved October 9, 2022, from https://arxiv.org/abs/1810.03993

*Why are gambling markets organised so differently from financial markets?\**. (n.d.). Retrieved October 10, 2022, from https://pricetheory.uchicago.edu/levitt/Papers/LevittWhyAreGamblingMarkets2004.pdf

## Appendix A, Source Code

[Kaggle Dataset - ‘NBA Historical Stats and Betting Data’](https://colab.research.google.com/drive/1EKZXzPZMpxa2W4R6dyXCtuo6lDd2NlqN?usp=sharing)

[Odds API Dataset](https://colab.research.google.com/drive/1rIHf0832eVcr61oJZ6yE4yOmSKikAkZq?usp=sharing)

[APISports NBA pipeline](https://colab.research.google.com/drive/12bpwh7HZcFL9BS290bGiUJD-l-tI-2mF?usp=sharing)

[Web Scraping - TeamRankings.com](https://colab.research.google.com/drive/1x-QYt09evA3rErTDd5F9C-6uHygqTJeR?usp=sharing)

[Data Perturbations](https://colab.research.google.com/drive/1EsML6MiZjHHhKDY9tWCjaXiXs12NoYc1?usp=sharing)

[Odds Modeling](https://colab.research.google.com/drive/1HxvLl6sJpfsx4LP-zmbTtmHaBpOcMgWB?usp=sharing)

## Appendix B, Timeline

| Task: |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Problem Definition | X | X |  |  |  |  |  |  |  |
| Dataset identification |  | X | X | X |  |  |  |  |  |
| Quality Metric Research |  |  |  |  | X |  |  |  |  |
| Dataset Perturbation/ Augmentation |  |  |  |  | X | X |  |  |  |
| Dataset Quality Measurement |  |  |  |  |  | X | X |  |  |
| Odds Modeling |  |  |  |  | X | X | X | X |  |
| Writing + Wrap Up |  |  |  |  |  |  |  | X | X |
|  |  |  |  |  |  |  |  |  |  |
| Date: | 8/22 | 9/5 | 9/19 | 10/3 | 10/17 | 10/31 | 11/14 | 11/28 | 12/12 |