Data Mining Final Project
Karina Latsko

Using Gradient Boosted Trees to Decide the Over/Under in NFL Games

In today's sports market, the clashing helmets of the National Football League (NFL) have an extra layer of tension, a new playing field that's been recently legalized in the United States. It's the realm of sports betting, and it's brought an unexpected boom, with the NFL being the superstar attraction. An incredible 81% of sports betting enthusiasts are placing their bets on NFL games (*US Sports Betting News*, n.d.). Now there are several ways to play the odds - the spread, the money line, and the over/under. In my research, I've decided to plunge into the fascinating over/under game, where bettors take a gamble on whether the combined point totals of both teams will be over or under a set number. The current professional betting landscape is fixated on the number 52.5% - the magical threshold that separates the winners from the losers in sports betting (*How Sports Betting Odds Work*, 2018). But why aim for just breaking even when we can aim higher, much higher?

I first had to collect, parse, and clean my data which I collected from the open source data website, Kaggle.com (*NFL Scores and Betting Data*, n.d.). This dataset has multiple descriptive columns including data on weather, stadium location, date, playoff games, etc., with data spanning back to 1966. Any rows with missing data in the 'team_favorite_id', 'spread_favorite', and 'over_under_line' were discarded to maintain data integrity. I then filtered out the games that occurred before 1979, as those games did not contain the most complete information on betting scores and may not best reflect the current competitive landscape of the game. I manually had to create Y prediction vectors of 0 or 1 for over/under which I accomplished by creating and external function that checked the score of the game against the over/under and assigned a value based on the result.

My first run of the model was performing relatively poorly with just the features at hand, so I knew that I needed to add different features that would enable my model to make better predictions. I added a team wins columns for home and away and average team points column for home and away to supplement the pre-existing features in the dataset. To convert my categorical variables into a format suitable for machine learning algorithms, I performed one-hot encoding on columns like 'schedule_week', 'team_home_id', 'team_away_id', etc.

I decided to use XGBoost, short for eXtreme Gradient Boosted Trees for this modeling project as it is part of an ensemble learning method based on the gradient boosting framework. Ensemble methods are designed to improve the model performance by combining several weak learners to form a strong learner. This principle is particularly beneficial in predicting over/under outcomes in NFL games, where the patterns may be intricate and complex. Performance-wise, XGBoost is designed for efficiency and speed. It utilizes parallel processing, enabling it to execute several tasks simultaneously, thereby speeding up the training process. This feature was crucial in this case as the NFL dataset is sizable and includes a multitude of features, especially after one-hot encoding. Lastly, XGBoost provides several knobs for tuning, giving me greater

control over the model's learning process. In this project, I experimented with several hyperparameters, including the maximum depth of trees, the learning rate, and the number of estimators. After some exploration through pointed hyperparameter sweeps, I found that a maximum depth of 5, a learning rate of 0.1, and 200 estimators offered the best trade-off between learning efficiency and model performance.

After splitting the data into training and testing sets and training the model, I achieved an accuracy of 62.63% on my test data and up to 67% accuracy in cross validation, which exceeded the average sports bettor's success rate by a significant margin. The cross-validation scores also exhibited low variability, suggesting robustness in my model. My model additionally generated an AUC-ROC scores of 0.694 which means that the model has a reasonably good ability to differentiate games that will end up over the line from those that will end up under. This score is significant because it is not only a reflection of the model's accuracy but also of its robustness and generalization capabilities. Unlike the accuracy score, the AUC-ROC score considers the model's performance across various thresholds, providing a holistic understanding of the model's predictive power. Moreover, the AUC-ROC score of 0.694 ties into the iterative nature of hyperparameter tuning that I undertook. By manually adjusting parameters such as maximum depth, learning rate, and number of estimators, I was able to strike an optimal balance between bias and variance.

While the model has performed well within this specific dataset, there are certain limitations and challenges that may arise when applying it to real-world settings. First, the nature of sports is inherently unpredictable. Factors such as player injuries, team morale, penalty calls, or even unexpected events can significantly influence the outcome of games, and these may not be entirely captured by our dataset or model. Furthermore, this model assumes that past performance is a reliable predictor of future outcomes. However, in reality, team dynamics and organization structures can change drastically from season to season, making predictions based on historical data less reliable. Lastly, while our model was tuned to optimize the AUC-ROC score, it may not perform as well on other metrics. To ensure the model's robustness, we may need to evaluate and optimize it based on several different performance metrics, which could add complexity to the model development process. Overall, while the model shows promise, it's crucial to consider these limitations and challenges when applying it in real-world settings.

Ultimately, the carefully crafted features and a well-tuned XGBoost model allowed me to predict over/under outcomes in NFL games with considerable accuracy, and importantly a higher accuracy than a standard professional sports bettor. This project explores the power of machine learning in the ever-evolving sports betting landscape, and I'm thrilled about the potential of leveraging data science and data mining to revolutionize sports analytics further.

Data Visualizations

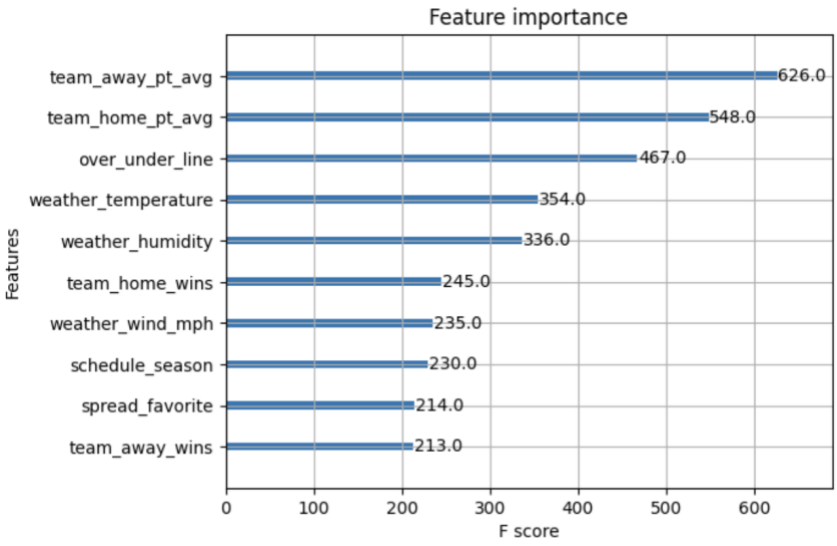

Figure 1. An F score histogram plot showing the features the model most heavily relied on to make decisions about if the game would go over or under on points. Unsurprisingly, the three top features were the team point averages and the line itself. I thought it was interesting that temperature was the next highest in importance over another weather feature like rain or snow.

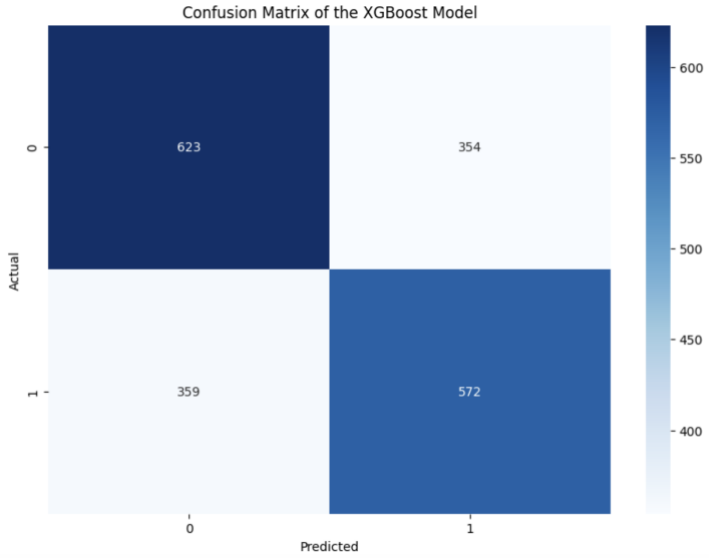

Figure 2. A plotted confusion matrix with a strong, dark, diagonal influence. A strong dark diagonal on a confusion matrix implies a high number of correct predictions by the model, both for positive and negative classes. The "dark diagonal" refers to high values in the cells corresponding to the True Positive (TP) and True Negative (TN) predictions. This pattern indicates a high degree of accuracy: the model has correctly predicted the positive cases as positive and the negative cases as negative.

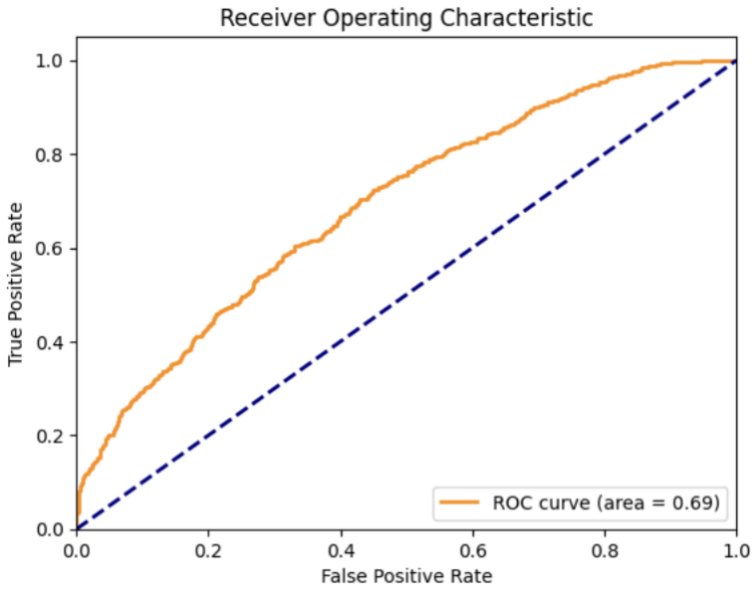

Figure 3. A graphical representation of the Receiver Operating Characteristic (ROC) curve, which is a critical measurement for evaluating the performance of our XGBoost classification model at different threshold settings. The ROC is a probability curve, and the Area Under the Curve (AUC) signifies the model's ability to distinguish between classes. The curved orange line in the graph represents our model's ROC curve and the navy-blue dashed line that runs diagonally from the bottom left to the top right symbolizes a 'no-skill' classifier. This classifier cannot differentiate between the classes and would predict a random class or a constant class irrespective of the input. The fact that our model's ROC curve lies significantly above this line indicates that it has a substantial level of skill.

Works Referenced

How Sports Betting Odds Work. (2018, August 7). BettingPros. https://www.bettingpros.com/articles/how-sports-betting-odds-work/

NFL scores and betting data. (n.d.). Retrieved May 31, 2023, from https://www.kaggle.com/datasets/tobycrabtree/nfl-scores-and-betting-data

US Sports Betting News: Which Is The Sport Americans Bet On Most? (n.d.). RotoWire. Retrieved May 31, 2023, from https://www.rotowire.com/article/us-sports-betting-news-which-is-the-sport-americans-bet-on-most-71608