Capstone Project Report: Predicting NFL betting outcomes

John (JP) Baselj

Problem

This analysis and modeling aimed to predict how NFL teams will perform against the betting spread in a given game based on data that is available through Kaggle and Pro Football Reference. Teams' scoring history, the location of the game, weather, and more factors were considered. The end goal of this modeling would be to apply the model to future games or weeks of NFL games and use the predictions to place bets on the outcome of games for a profit.

Data Aquisition, Processing, and Analysis

The raw dataset loaded from the "spreadspoke_scores.csv" file from Kaggle[1] contained 13,516 entries, each representing a single NFL game from 1966 to 2022. Complete betting data (favorite, spreads) were only available for 11,027 of these games, so all other entries were removed, as the primary goal of this exploration was to predict performance against those spreads. Weather data was also incomplete for every entry, so it was assumed that any game not recording their weather had neutral weather and was given a weather status of 'none'.

The scores dataset initially consisted of 44 different NFL teams, although there have never been simultaneously more than 32 teams. The "nfl_teams.csv" dataset from Kaggle[1] was imported and used to identify NFL *franchises*, and reduce teams that have changed names (such as the Baltimore Colts franchise moving and rebranding to the Indianapolis Colts) to give each *franchise* a unique identifier. This simplified the data so that there would not be more than the true 32 teams for model encoding.

The resulting dataset was analyzed to identify any clear trends in the target variable, team performance Against The betting Spread (ATS), and the distribution of all home team performances against the spread was found to be approximately normally distributed (figure 1), with a mean of about -0.14 points and standard deviation of abouit 12.7 points. It was also investigated whether mean and standard deviation had trends over time (figure 2), which does not appear to be the case.

1. https://www.kaggle.com/datasets/tobycrabtree/nfl-scores-and-betting-data?select=spreadspoke_scores.csv

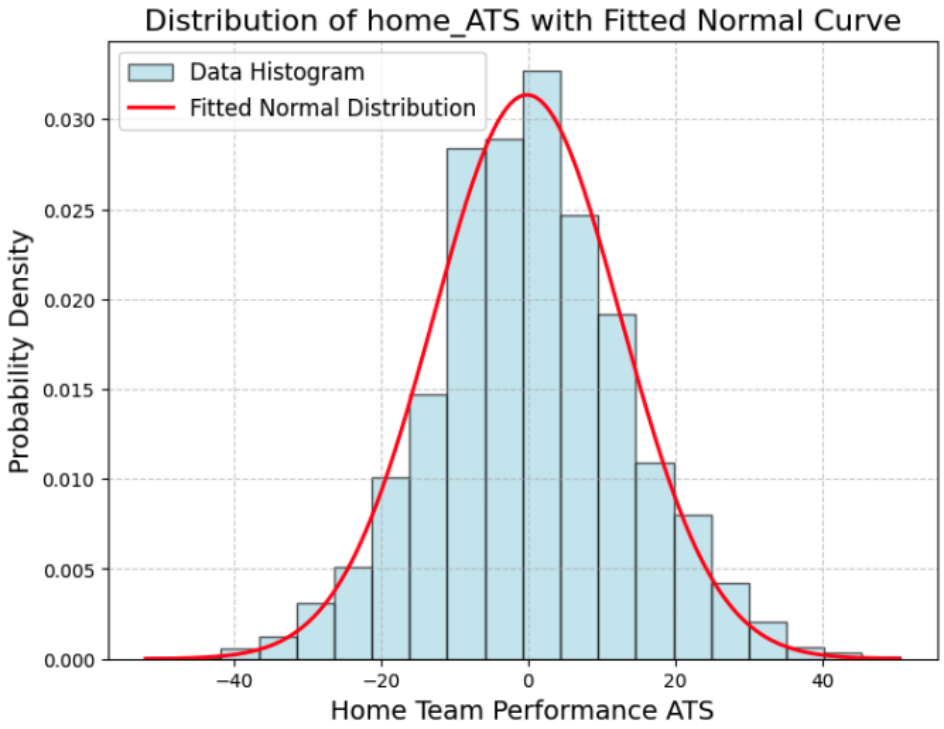

Figure 1

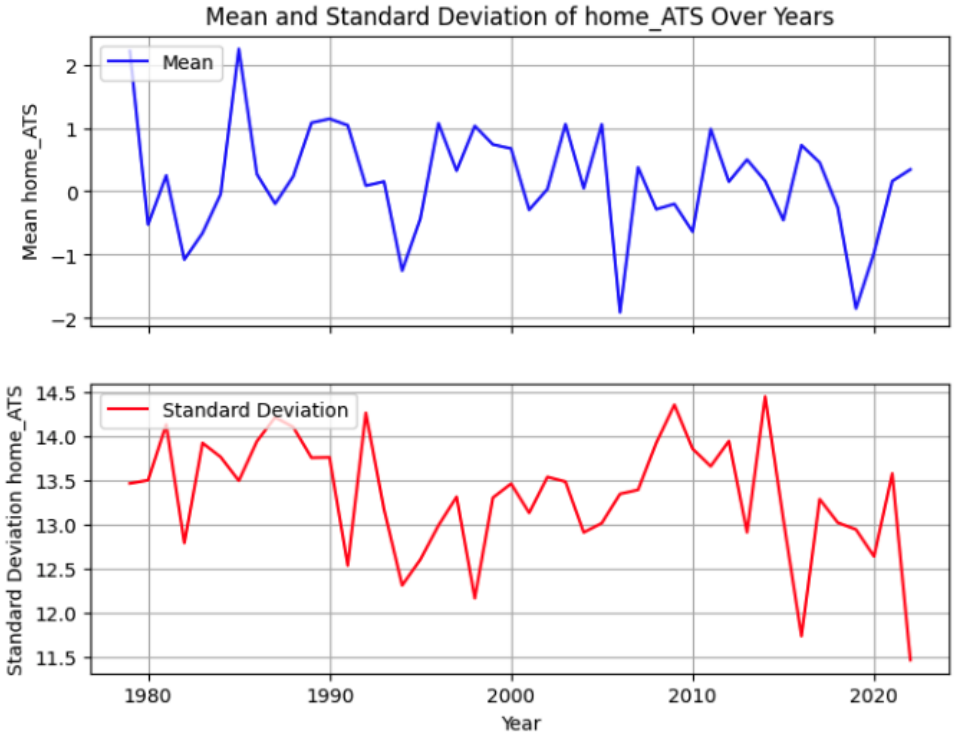

Figure 2

Analysis was performed across many fields in the dataset to determine which fields had significant correlation to the target variable home_ATS, and the team conference and team division fields were found to have little to no affect, and were therefore removed.

Data Preparation for Modeling

To create a predictive dataset X that can be used for model training and future game predictions, fields that were game-result dependant were removed from the dataset. For example, the home team's score in a game cannot be used as an input to a predictive model because that score will never yet be available for a future game that needs to be predicted.

Each field was modified to a format which would be compatible with standard ML modeling techniques, turning 'object' type fields into numerics as applicable and dummy encoding categorical variables. Additionally, numeric data was scaled using sklearn.preprocessing StandardScaler to reduce the effects of mismatched data scales.

A validation set of the most recent nfl season in the dataset, the 2022 season, was set aside to evaluate the best model from training.

Modeling Approach

Splitting data into training and testing sets through conventional randomized splits, such as that provided by sklearn's train_test_split function, were found to be inappropriate for training models to predict sports outcomes because more recent games are likely to have better predictive influence over upcoming games. For this reason, the historical NFL scoring data was treated as a modified time series, and train/test splits were defined by iteratively applying the sklearn TimeSeriesSplit function. This splitting technique is visualized in Figure 3's "Time-Based Train/Test Split". For each model type evaluated, iterations were performed of:

1) Set aside approximately 1 week of NFL games as a test set
2) Train a model on all of the games leading up to, but excluding, that test set
3) Evaluate how well that model could predict the outcome of the test set games

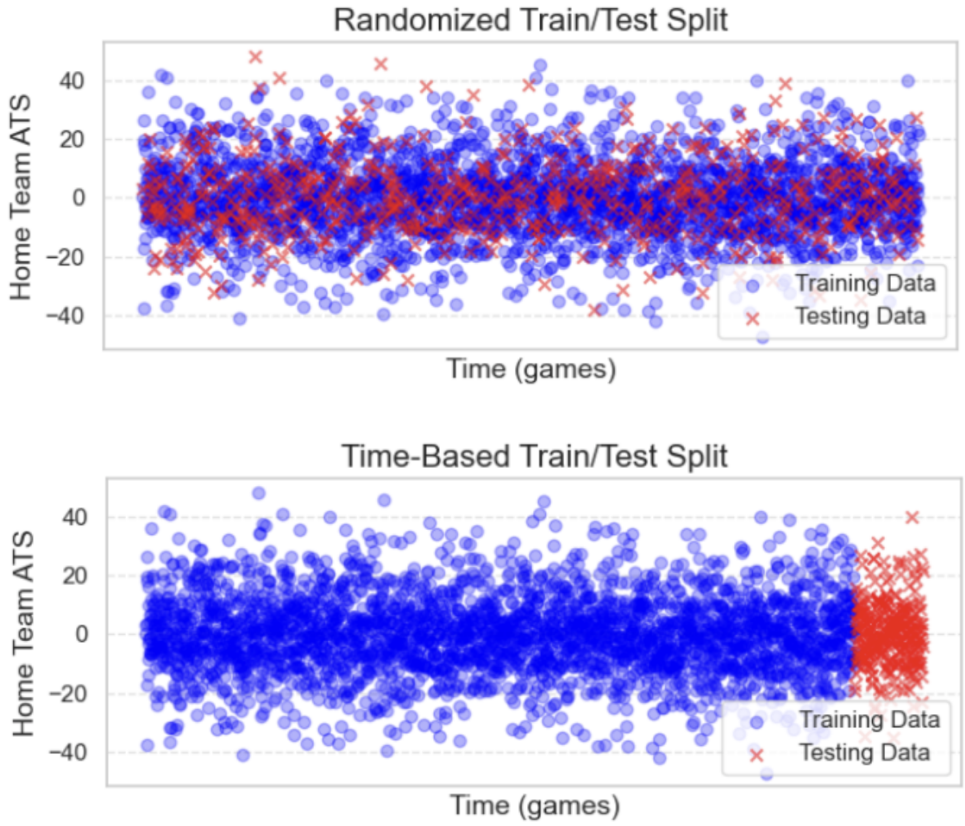

Figure 3

The target variable which this modeling effort aimed to predict, home team performance against the betting spread, can be represented either as a continous or a discrete variable. For each of the two approaches, two models were trained and evaluated:

1) Discrete variable - will the home team beat the spread?
    a) Logistic Regression Model
    b) Random Forest Classifier Model
2) Continuous variable - by how many points will the home team beat, or lose to the betting spread?
    a) Random Forest Regressor Model
    b) Linear Regressor Model

Modeling Results

The 4 model types were each trained and compared using the accuracy score as the primary performance metric, because the classes of whether or not the home team beats the spread are generally balanced, and we care primarily about maximizing the number of correct predictions. Upon training, tuning, and testing each model type, the accuracy scores ranked:

1) Random Forest Classifier Accuracy: 0.551
2) Random Forest Regressor Accuracy: 0.544
3) Logistic Regression Model Accuracy 0.523

4) Linear Regressor Accuracy: 0.511

The Random Forest Classifier (RFC) was therefore selected for further optimization and exploration. It was hypothesized that using only the strongest predicitons from the RFC built-in method predict_proba() may result in higher predictive power, however this proved not to be the case. A simulation was performed to 'place bets' on only the top 50% of game predictions, sorted by the RFCs confidence of each prediction, and the model only predicted about 54% of those bets correctly.

The RFC model was finally applied to the holdout dataset, games from the 2022 NFL season. It only predicted the correct outcome of games 51% of the time, which demonstrates that the relative success of predicting about 55% of games during training may be the result of model overfitting.

that the best model built from the dataset used in this project is most likely not a viable model to predict NFL games and generate a profit.

Conclusion

The validation testing results of correctly predicting about 51% of the games in the 2022 season indicate that that the best model built from the dataset used in this project is most likely not a viable model to predict NFL games and generate a profit.

The target variable in this study, home team performance against the betting spread, is inherently very complex to predict because it is a composite value not only of game outcome, but also must consider the complexity of the sports books developing that pregame predicted betting spread in the first place. It is not extremely surprising, then, that it proved challenging to develop a model to predict teams' performance against the spread using the relatively simple data used as input to this project.

In a future attempt to predict game outcomes better than sports books, it may be valuable to include more sophisticated game data not included here such as:

- relative team health (injuries)
- skill/experience of important people like head coach or quarterback
- team style of play over time, like when new coaches bring a change in strategy

As part of a larger-scope project, it could also be valuable to test more complex ML model types such as neural networks.