

NFL Betting Analysis

Brian D. Mochtyak



Introduction

Sports gambling in the United States is a popular and ever growing industry. The reason why it is so popular is because sports books, an establishment that takes on bets and pays out winnings on sporting events, are very profitable businesses, and the adrenaline and potential winnings for players is incredibly alluring.

Most players bet with their gut or anecdotal evidence which keeps money flowing from players pockets into sports books balance sheets. The goal of this project was to use historical data to generate a better way for players to place bets on NFL games to best the odds makers and the sports books,

Although there are many different ways to bet on NFL football, this analysis only applies to spreads. The way the spread works is in every game there is a favorite team and an underdog. The favorite (shockingly) is the team that is expected to win, and the underdog is the team that is expected to lose. If a player bets on the favorite, not only does that team need to win the game in order for the bet to hit, but the favorite also needs to win by a number of points designated by the sports book, the spread. In order for a bet on the underdog to hit, the underdog needs to either win the game or lose by a smaller amount of points less than the spread.

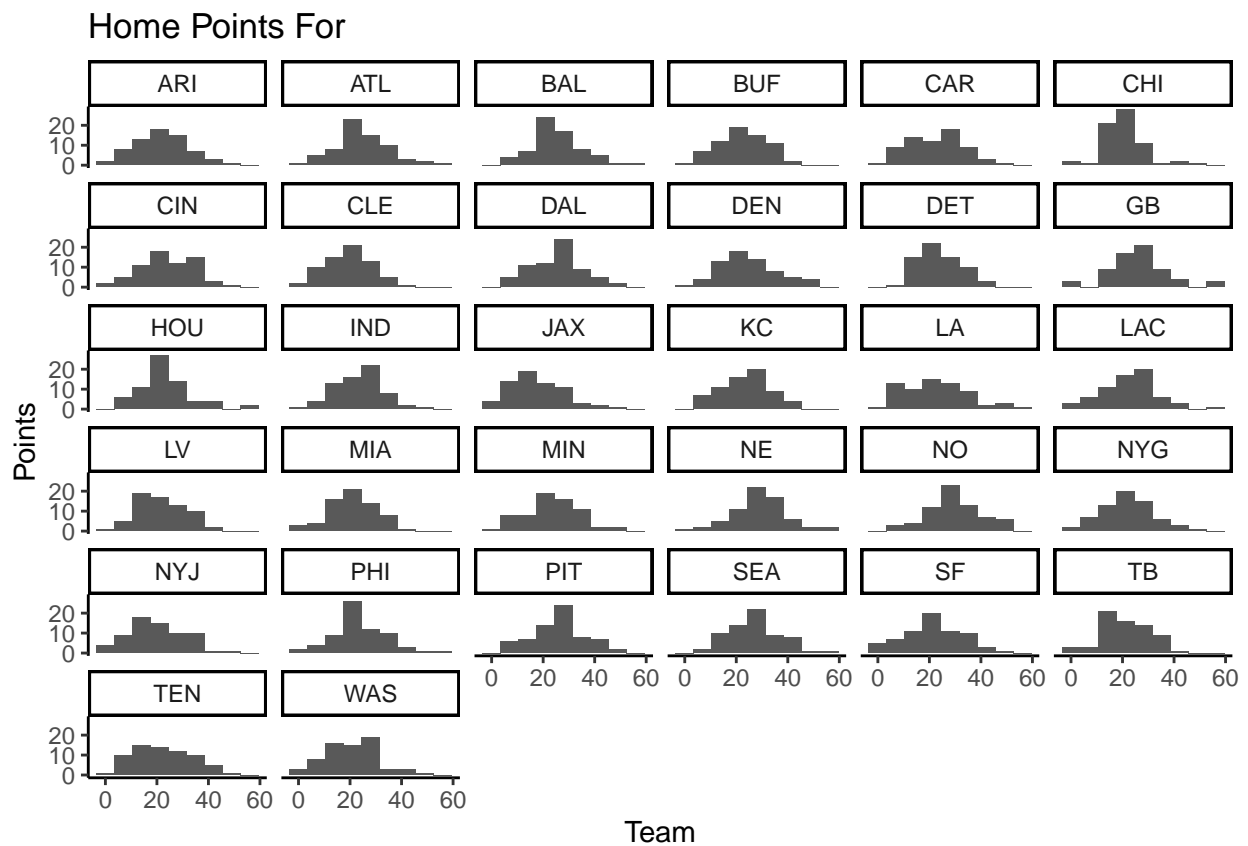
This analysis was focused on placing bets on favorite and underdogs for week 8 of the NFL 2020 regular season.

Data

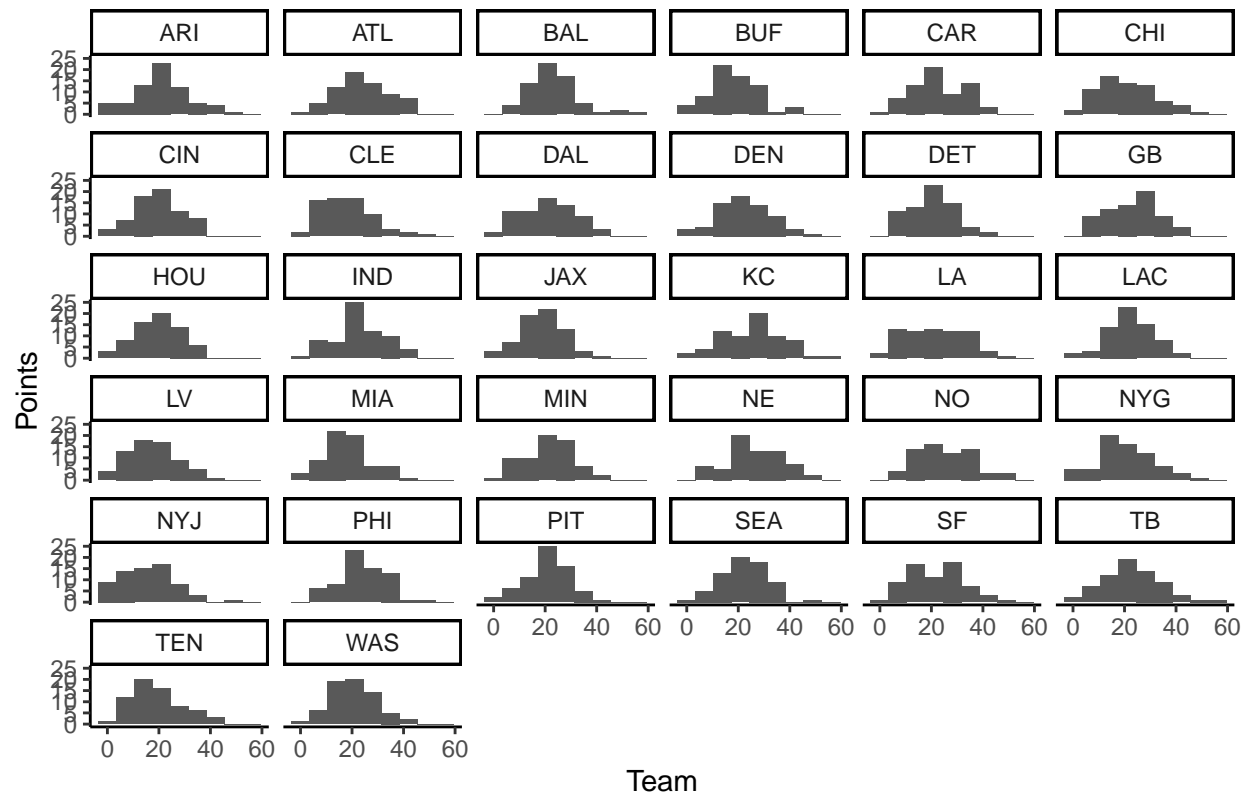
The Data used in this analysis was historical NFL regular season game scores from season 2012 through the week before the current week, and the week 8 2020 NFL betting lines.

Methodology

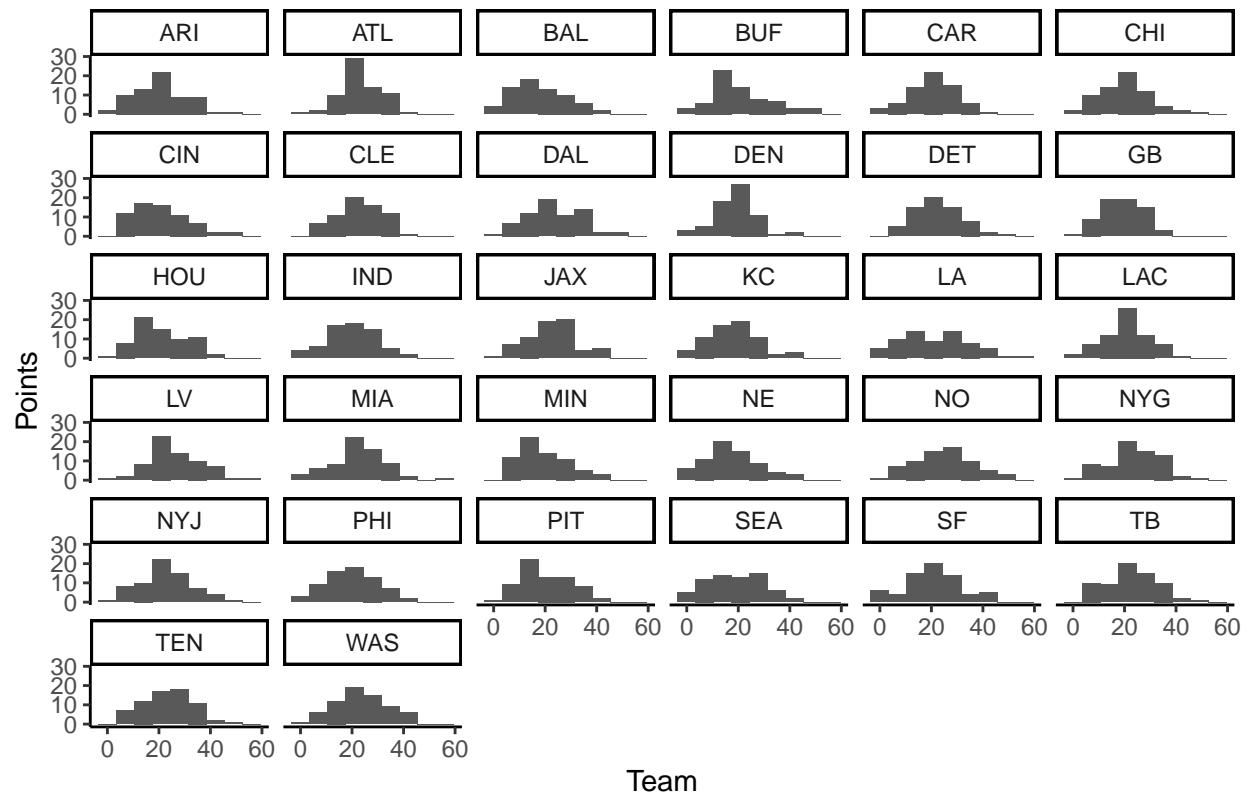
Before optimizing the allocation of money determining how likely certain bets were to be hit and their payoffs needed to be calculated. As can be seen in the distributions below, the home and away points for and home and away points against are distributed normally (although not all are perfect bell curves)

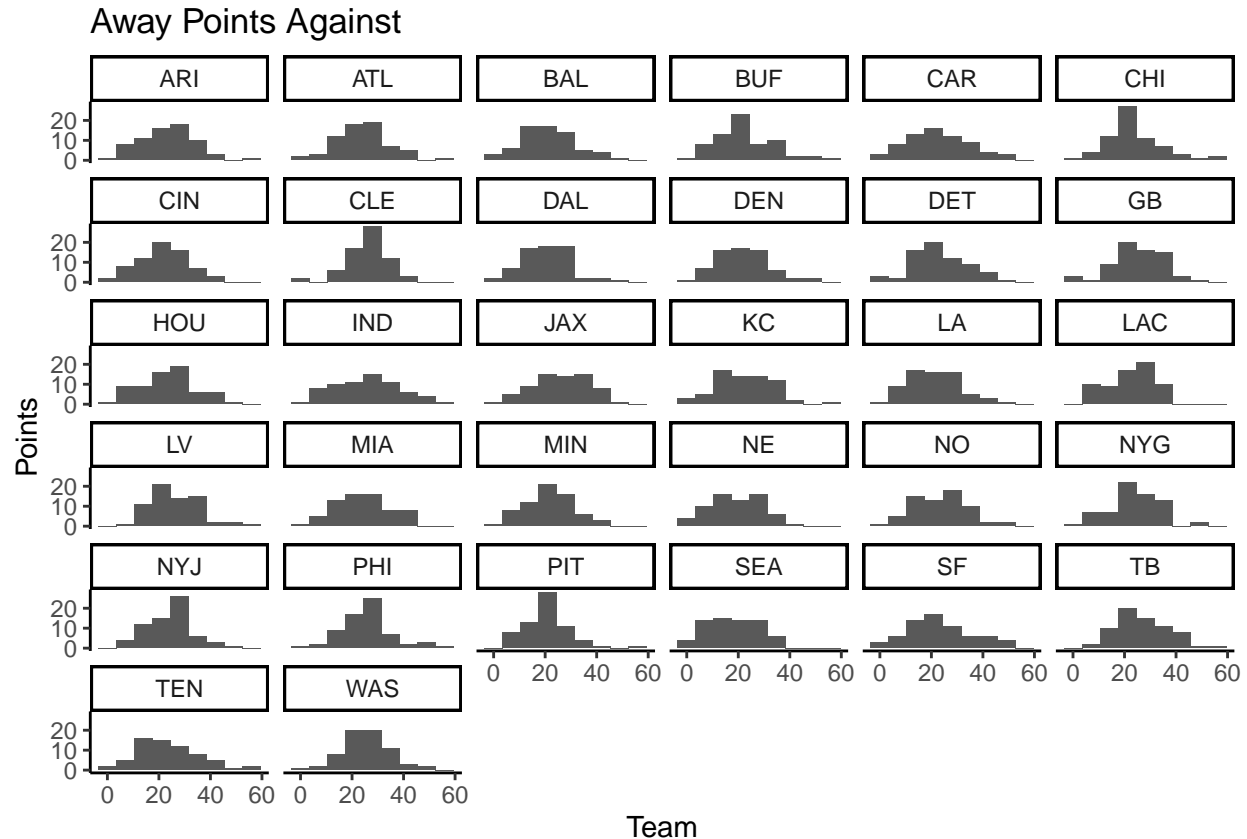


Away Points For



Home Points Against





Using this historical data, both a mean and standard deviation for points scored and points against home and away are calculated for every team.

Then taking the spreads and the totals (the expected amount of points to be scored by both teams combined) an implied points is created for both the favorite and the underdog. This is done using this formulas:

Implied Points Favorite = $\text{Total} / 2 + \text{Spread} / 2$

Implied Points Underdog = $\text{Total} / 2 - \text{Spread} / 2$

Payouts per dollar bet are then calculated for each potential bet as well using the odds provided by the sports book. If you are unsure how betting odds work there is plenty of useful information provided from casinos and sports books.

Using the means and standard deviations z scores are calculated for the implied points. The teams home means and standard deviations are used if the team is at home this week and vice versa if the team is away. Four total z scores are calculated:

Favorite points scored
 Favorite points allowed
 Underdog points scored
 Underdog points allowed

Then using the z scores a percent chance that the favorite and underdogs will score and allow these implied points based off where that number falls in the distribution. Taking an average of these four percentages creates a percent chance the favorite will cover the spread (betting on the favorite hits). Subtracting that percent chance from one then provides the percent chance that the favorite will not cover the spread (betting on the underdog hits).

Then multiplying these percentages and the payoff per dollar an expected payout for each bet is generated, which can be seen in the table below.

Table 1: NFL Week 8 Bets and Expected Payouts

Favorite	Underdog	Spread	Expected_Payout_Favorite	Expected_Payout_Underdog
CAR	ATL	-2	0.9746501	0.9567884
IND	DET	-3	0.7640240	1.1450669
LA	MIA	-4	0.8230760	1.0863418
CLE	LV	-3	0.9382724	0.9708185
GB	MIN	-7	0.8360231	1.0723276
BUF	NE	-4	0.7112395	1.1978514
KC	NYJ	-20	0.5072491	1.3815869
BAL	PIT	-4	0.9336646	0.9712242
TEN	CIN	-6	0.6406527	1.2684382
LAC	DEN	-4	0.6935219	1.1958768
NO	CHI	-4	0.7821606	1.1300339
SEA	SF	-3	1.0143102	0.8892614
PHI	DAL	-8	0.7066599	1.2024310
TB	NYG	-11	0.5413524	1.3677385

After the expected payouts have been calculated linear optimization is then used to maximize the total return.

The decision variables are the amount of money a player can place on each bet. The objective function is maximizing the expected returns from each bet placed.

There are two restraints, the budget, the total amount of money that can be wagered this week, (\$100 was chosen, but this was arbitrary) and the max amount that can be placed on a single bet which is 10% of the total budget. The reason the 10% max was chosen is because if this constraint wasn't in place the model would place the entire budget on the bet that had the highest potential payout and it is best to avoid putting all eggs in one basket.

Results

The table below shows the bets that the model placed for this particular week. All of the bets were placed on the underdogs, and the expected payout is \$120.48, or a 20% return.

If these were bets were to be placed this week the player would have seen a 34% and the bets would have hit 7 out of 10.

Favorite	Underdog	Spread	Favorite_Bet	Underdog_Bet
CAR	ATL	-2	0	0
IND	DET	-3	0	10
LA	MIA	-4	0	10
CLE	LV	-3	0	0
GB	MIN	-7	0	10
BUF	NE	-4	0	10
KC	NYJ	-20	0	10
BAL	PIT	-4	0	0
TEN	CIN	-6	0	10
LAC	DEN	-4	0	10
NO	CHI	-4	0	10
SEA	SF	-3	0	0
PHI	DAL	-8	0	10
TB	NYG	-11	0	10

Conclusion

Although, this is a good start on this problem there is much to be improved upon. There are potentially better methods to calculate the percent chance for each bet. Also, there are many different wagers you can make on NFL games such as totals, money lines and props. Coming up with system to include these different bets would most likely improve performance.

Lastly, this was used for just one week and results need to be seen over the course of the entire season before determining the full effectiveness of the model.

All code can be found on my github:

<https://github.com/Theotecles/NFLData>

Thank you!