Jeremy Sanchez
9 December 2020

Over Forty-Nine: Predicting NFL Point Totals

Objective, Background, and Motivation:

The goal of this project was to predict the total points scored in NFL games taking place in weeks twelve and thirteen during the 2020 regular season. In particular, I predicted whether games played in these weeks will reach totals over or equal to forty-nine points.

For many years, sports betting in the United States was legal only in Nevada. However, in the wake of *Murphy vs National Collegiate Athletic Association* (2018)[1], nineteen states and the District of Columbia have legalized betting on sport[2]. 'Over/Under' bets are among the standard bets placed in sportsbooks. Furthermore, the ruling has spawned daily fantasy football, where players submit a team they believe will score the most points and play for a cash reward. My motivation for pursuing this project is two-fold. Firstly, I can attempt to make my own betting lines and compare them with those made by oddsmakers. If I find good value — in other words, if the probability of a game going over 49 points in my model is larger than that of the sportsbooks' model — then placing a bet on this game is advisable. In addition, higher point totals bode well for offenses of one or both teams involved in the game. With this knowledge, I can create daily fantasy lineups that have a greater chance of cashing in.

The Dataset:

For this analysis, I compiled a dataset from ESPN, TeamRankings and Pro Football Reference which contains information on games played from weeks one through twelve of the 2020 NFL regular season. The final dataset consists of **191** rows and **12** columns.

- **Response:**
• *Over 49* – Indicates whether the game described by the row finished with a combined team point total of greater than 49.

- **Covariates**:
• Points Per Game Combined (PPG) – The average points scored per game by both teams heading into the game played, added together.
• Points Per Game Allowed Combined (PPAG) – The average points allowed per game by both teams heading into the game played, added together.
• Yards Per Game (YPG) Averaged – The points allowed per game by both teams heading into the game played, averaged.
• Yards Per Game Allowed Averaged (YPAG) – The yards allowed per game by both teams heading into the game played, averaged.
• Passer Rating Averaged (PR) - The average passer rating for the player with the

most pass attempts in a game for each team heading into the game played, averaged.
Passer Rating is the NFL's traditional statistic for evaluating passer performance in a single game. It considers pass attempts, completions, yards, touchdowns, and interceptions (when a defensive player catches a ball intended for an offensive player).

- Passer Rating Against Averaged (PRA) - The average passer rating recorded by the player with the most pass attempts in a game against each team heading into the game played, averaged.

- Quarterback Rating Averaged (QBR) - The average quarterback rating for starting quarterbacks for each team heading into the game played, averaged.

  QBR is a proprietary statistic developed by ESPN, which assesses higher scores that perform better against tougher opponents and in important game situations[3].

- The statistics corresponding to Plays Per Game Averaged (PLPG), Passing Yards Per Game Averaged (PYPG), and Rushing Yards Per Game Averaged (RYPG) are defined similarlyIn the first week of the season, when averages were not available, I consulted the 2019 NFL regular season averages.

- Here is the row in my dataset representing the game played between the Detroit Lions and the Atlanta Falcons during Week 7 of the 2020 NFL regular season:

Over 49	PPG	PPAG	YPG	YPAG	QBR	PR	PRA	PLPG	PYPG	PYAPG	RYPG	RYAPG
0	54.40	53.24	375.48	405.75	66.63	97.96	105.84	70.75	260.78	114.70	284.66	121.08

- **Effects on my analysis due to COVID-19**
  1. Week twelve – New Orleans Saints at Denver Broncos
     In this game, all listed quarterbacks on the Denver Broncos were ineligible due to a breach of COVID-19 protocols. Therefore, QBR and PR entries for the Broncos, as well as the PRA entry for the Saints, were not recorded.

  2. Week thirteen – Dallas Cowboys at Baltimore Ravens
     This game, originally scheduled for Thursday, December 3, was pushed back to the following Tuesday due to positive tests on the Ravens. Due to time

constraints, this game was excluded as a testing point in week thirteen.

• Training and Testing Split:
I had two pairs of training and testing sets. The first pair was used to predict point totals in week twelve. The training set had **161** rows (for games through week eleven) and there were **16** testing points (games played in week twelve). The second was used to predict point totals in week thirteen. The training set had **177** rows (adding those rows from the games played in week thirteen), and **14** testing points (games played in week thirteen, excluding Dallas Cowboys at Baltimore Ravens).

• Why Forty-Nine?
The median of the total points scored in NFL games through week eleven of the 2020 regular season was forty-nine. I elected to choose the median over the mean simply because the mean is less robust against high-scoring or low-scoring games, and the dataset is relatively small.

Modeling:

• Principal Components Regression (PCR)
I used this model because I believed some of the data were correlated. If an offense has a successful day, there is a good chance that their corresponding statistics will be higher across the board. Indeed, here are some correlation coefficients between the covariates of my training dataset for week thirteen:

PPG and QBR	0.75
YPG and PPAG	0.721
YPG and QBR	0.637

To implement PCR, I used the first n covariates from Principal Components Analysis (PCA) that explained ninety-five percent of the variance in the original covariates. I then performed a logistic regression on these new covariates, with each covariate having a linear term.

• Trees
I ran three tree models in this analysis: a single, pruned tree, bagging, and random forest. I chose trees to implement a model that might account for matchups. For example, if a team throws for a less than average amount of yards per game, but their next opponent gives up a lot of passing yards, we might suspect that the team's passing yards might be higher this week than usual. I thought that a tree could make splits on something like this and come up with a reasonable prediction.

• Support Vector Machines

I fit two support vector machines to my data, one with a linear kernel, and another using a radial kernel. I chose support vector machines because they are an algorithm meant for classification, and because they incorporate non-linearity quite easily. I was not convinced that there was a linear relationship in my dataset at the start of this analysis.

• 'Naive' model
This 'model' was implemented as a check to see if the models I fit were performing well. I simply predicted all games to see over forty-nine points scored.

Results:

• Key parameters for respective models:

Model	Week 12 Parameters	Week 13 Parameters
Principal Components Regression	Covariates: 8	Covariates: 8
Single, Pruned Tree	Number of splits: 1	Number of splits: 1
Bagging	Trees constructed: 500	Trees constructed: 500
Random Forest	Trees constructed: 500 Number of covariates to split on: 3 out of 12	Trees constructed: 500 Number of covariates to split on: 4 out of 12
Support Vector Machine (linear kernel)	Cost variable: 89 γ: 0.0001	Cost variable: 55 γ: 0.0001
Support Vector Machine (radial kernel)	Cost variable: .01	Cost variable: .01

• Model Performance, sorted by average (weighted) Misclassification Rate (MCR):

Model	Week 12 MCR (16 test points)	Week 13 MCR (14 test points)	Average (weighted) MCR
'Naive'	0.5000	0.5718	0.5333
Principle Components Regression	0.5000	0.5718	0.5333
Single, Pruned Tree	0.5000	0.5000	0.5000
Bagging	0.3750	0.5000	0.4333
Support Vector Machine (Linear kernel)	0.3125	0.5000	**0.4000**

Model	Week 12 MCR (16 test points)	Week 13 MCR (14 test points)	Average (weighted) MCR
Support Vector Machine (Radial Kernel)	0.3125	0.5000	**0.4000**
Random Forest	0.3750	0.4286	**0.4000**

Takeaways and Future Work:

Random Forest and the two versions of Support Vector Machine (radial and linear kern els) performed the best, with Random Forest predicting better than 50 percent of score outcomes in both weeks twelve and thirteen. I expected Random Forest to outperform the other tree models I used here, particularly bagging, because we can choose exactly how many covariates to split on, and because this feature can help de-correlate trees, especially with the correlation in our covariates mentioned earlier. It also turns out that the Support Vector Machines were up to this classification job, particularly in week thirteen.

Most importantly, **all models performed as well as or better than the 'naive' model.** If I had to choose a best model, I would choose **Random Forest**. However, our sample size is still very small, so we will have to run these models in future weeks to see if our performance holds steady.

One model which I did not have time to implement but bears mentioning is Partial Least Squares (PLS). Unlike PCA which is unsupervised, PLS creates new covariates so that the variability in these covariates is that which explains the response variable. In this way, it could pick up on a relationship in my dataset which PCA could not.

Here are some other variables I can add to this dataset which I believe might enhance the performance of my models:

- **Pass percentage:** this would be an improvement on plays per game, since teams that throw the ball more generally traverse the field faster and therefore allow for more points scored in a game.
- **'Strength index':** This reinforces my desire to capture matchups in my models. This could resemble something like:

$$\text{Strength index}_{team} = \frac{number\ of\ wins_{team}}{\sum_{o \in Oteam} s_o}$$

where O_{team} is the set of opponents that team has faced over the course of the year, and s_0 is a value based on team strength for each of those opponents which varies around 1. Lower values of s_0 would indicate a stronger team. Therefore, indexes lower than 1 would indicate teams who have been helped by their schedule. To include this in the model, I would take a ratio of strength indexes. If this ratio is near 1, then the teams should be at even strength.

- **Weather and temperature:** Games played in colder temperatures and/ or an unfavorable outdoor environment might lead to a lower point total. In week ten, the Cleveland Browns bested the Houston Texans by a score of 10 to 7 under extremely windy conditions[4]. This game happened to be the lowest scoring game of the season up to that point. To have a marked effect on my predictions, however, I'd likely need to have many more data points than I have now.

References

1. Wolf, Richard. "Supreme Court Strikes down Ban on Sports Betting in Victory for New Jersey." *USA Today*, Gannett Satellite Information Network, 14 May 2018, www.usatoday.com/story/news/politics/2018/05/14/supreme-court-strikes- down-ban-sports-betting-new-jersey/1053022001/.

2. Rodenberg, Ryan. "United States of Sports Betting: An Updated Map of Where Every State Stands." *ESPN*, ESPN Internet Ventures, 3 Nov. 2020, www.espn.com/ chalk/ story/_/id/19740480/the-united-states-sports-betting-where-all-50- states-stand-legalization.

3. "Total Quarterback Rating." *Wikipedia*, Wikimedia Foundation, 6 Oct. 2020, en.wikipedia.org/wiki/Total_quarterback_rating.

4. Lane, Mark. "Texans QB Deshaun Watson Was Affected by the Windy Weather in 10-7 Loss to the Browns." *USA Today*, Gannett Satellite Information Network, 16 Nov. 2020, texanswire.usatoday.com/2020/11/16/windy-weather-affected- texans-deshaun-watson-10-7-loss-browns/.