Zach Austin and Jimmy Brunette
ITAO 40420
10.7.2021

Predicting the next play of a college football game in the 2020 College Football Season

Data sourced from cfbfastR

Table of Contents

Introduction

Adapting to an opponent's next play in a football game is a crucial piece of strategy that can give a team a significant edge on defense. However, some teams have introduced apparently random schemes, audibles, play signals, and other play calling methodologies that deviate from typical football norms; play calling styles have become more erratic and less predictable for an opponent.

The goal of this analysis is to present information from models correlating several measures of an offense to the play call outcome of either run or pass. In order to explore these relationships, extensive play-by-play college football data was utilized, fed into XGBoost models optimized for specific groups of college football teams to successfully categorize play predictions.

Models for play calling allow for a robust memory of how a team acted in prior situations and extrapolating the characteristics of the team's strategy and risk profile to other game situations. This type of prediction was found to be intriguing, as this involved modeling this seemingly unpredictable event in sports, and the model output has seamless applicability in coaching, spectator, and betting settings.

Related Work

An extensive academic or other formally published study of this exact form of play calling modeling was not found to exist. However, there were two alternative routes that were helpful in analyzing comparable results.

Firstly, several blog type posts were located in which the writer outlines a machine learning process employed very similarly with play-by-play data to predict an offensive outcome. Each of the studies were conducted on NFL data and not college football; yet, linearities between the two leagues indicate that this shouldn't create any drastic differences. Each of the blogs indicated use of an XGBoost model that obtained quite similar accuracy measures to that created for this report on college football data. Variable importance very much overlapped between studies, but the major discrepancy was that the NFL dataset employed in both cases had information regarding offensive formation at snap; a variable not included in the college dataset.

Secondly, a select group of casinos, such as Live-Bet, offer betting on the offensive play calling during a game. A bettor can select, for example, run or pass in the brief window of time from when the offense lines up to when the ball is snapped. Of course, the casinos alter probability lines for this sort of bet depending on the in-game situation; the odds implied by each bet can be utilized to reach a probability of the play call outcome, which can then be compared to the model's indication of play outcome which is on the logistic probability gradient from 0 to 1.

Data Description

Data on college football play by play results was obtained from the cfbfastR package, a community R package. The initial data had roughly 104,000 observations with 330 descriptive variables per observation. 141 teams were included in the dataset. The sheer magnitude of plays provided a strong opportunity for analysis on the qualities and metrics of interest. However, with the goal of predicting a binary run or pass outcome, many of the variables in this dataset were conceptually unrelated to the predicted outcome. Variables were therefore prescreened and eliminated based on conceptual relatability to the outcome. Additionally, variables were introduced that were calculated using the original measures in the data set to target certain areas of insight. The form of the tidied dataset used for modeling had roughly 80,000 observations spanning 32 predictor variables.

60% of this tidied data set was used as a training partition, and thus 40% was used as a testing partition. Figure 1 in the appendix visualizes the data narrowing process.

The below table briefs over variables that were used, removed, or calculated. This does not include to the fullest extent the variables that were used in the modeling; this is simply to bolster a conceptual understanding of what was included.

Used from original data	**Removed from data**	**Calculated from data**
Down	All player names	Rush yards per play in game and season
Distance to first down	All special teams plays	Pass yards per play in game and season
Win probability before a play	Unrelated casino metrics	Pass completion rate in game and season
Time remaining in the game	Unrelated play outcome statistics	Defense rush yards allowed in game and season
Yards to goal	Extensive qualitative descriptions of a play	Defense pass yards per game allowed in game and season

Methods

In the modeling process, two primary decisions had to be made: the scope of teams applied in models, and which modeling methodology would be used. Each decision will be elaborated upon below.

As for the scope of teams applied in models, there were generally three paths to select from. These included: modeling every college football team in one model; modeling each team individually; or grouping the teams meaningfully and modeling each group. When every college football team was modeled, macrotrends were generally captured and there was beyond sufficient data, yet the unique decision making differences between teams were largely eliminated. When teams were modeled on an individual basis, sample sizes became quite thin, and the model was incredibly overfitted to data. Predictive power was sufficient, yet the overfitting can lead some in-game situations incredibly askew if too similar to the training data. The final option of modeling groups of teams was selected for this project; clustering the teams meaningfully and modeling the data in the cluster; as it maintained the characteristics that make each team unique in decision making while also supplying beyond sufficient samples. Overfitting was mitigated and predictive power was quite strong.

To group teams meaningfully, a k-means clustering algorithm was conducted on the data. The data supplied for clustering was derived from the original dataset, yet was quite different from the data used for the XGBoost model. Mutations were made that better reflect decision making and common defensive points of exploitation rather than play outcomes. Thus, teams would be grouped by their similar play calling tendencies in response to weaknesses of other teams. To do so, variables for the clustering dataset were calculated and thus introduced that reflect decision making, such as average yards per play (for both rush and pass, defense and offense) as well as rush frequency (by each down). These mutations were necessary, as k-means clustering is an unsupervised learning method and would not have been able to make much use of the XGBoost dataset that only supplies run or pass as a play outcome.

Upon conducting clustering, it was determined that 6 groups were optimal. As is demonstrated by Figure 2 in the appendix, anywhere from 6-8 clusters seemed to be at the "elbow" of the plot - when more than 6 clusters were used, the subsequent XGBoost model performances varied far more drastically by group.

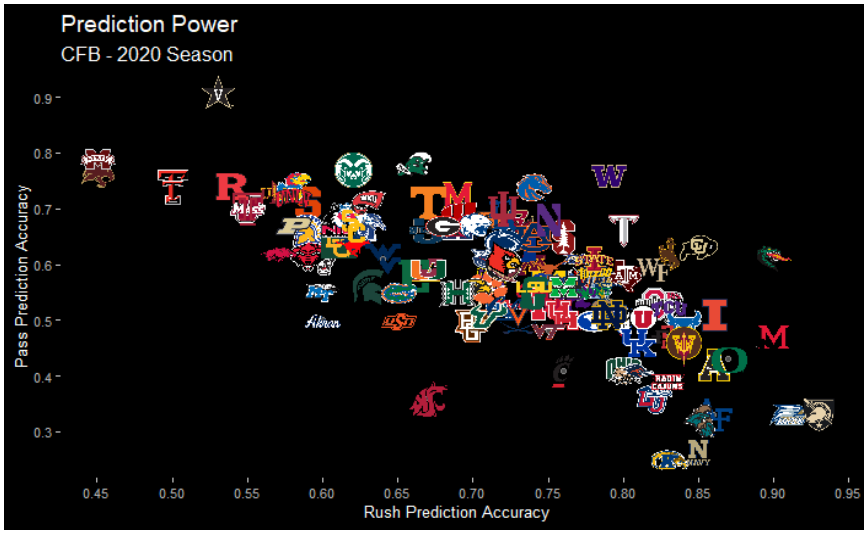
Upon clustering the teams, six XGBoost models were created for each cluster, using the data engineered specifically for XGBoost use. The data would be filtered only to include the teams involved in the cluster, and then the model would be formed. 60% of the data was randomly sampled for the training set. The six models within each cluster all utilized various learning rates; therefore, the model that produced the greatest accuracy at a specific number of trees and learning rate could be selected as the primary model for the cluster. When deployed on all of the training data, the model calculated an optimal cutoff for each individual team that would be used on testing data.

Ideally, other modeling methodologies, such as bagging and random foresting, could have been used to compare accuracy to the XGBoost models. However, the dataset used for modeling had too many instances of 'NA' that bagging and random foresting could not be used; the function for XGBoost is designed to handle instances of NA, whereas the random foresting and bagging functions are not. While there are several ways around this, namely imputing values for the NA, it was decided that using XGBoost alone was certainly the wisest decision, as the NAs in the dataset were all incredibly intentional. The NA values all resided in calculated columns and were kept as blank to avoid misrepresentation of in game situations. For example, a column was created that was the average rush yards per carry in a given game. It was decided that this calculation would not be conducted until at least 7 carries had occurred, and the first 6 would be deemed NA. This is to avoid an instance where a team may have one huge run on the first play of a game, resulting in a heavily overweighting variable. NA is more suited to be in this column than an unrepresentative average, or even an imputed value that might cloud the effect and momentum of a large play that has impact seen on other variables such as win probability. Further, the XGBoost was selected as an optimal combination of learning rate and number of trees, and thus isn't by any means a 'shot in the dark' modeling method that needs comparison to bagging and random foresting; already it had been optimized.

Results

Upon selecting an XGBoost model for each cluster, a specific in-game instance of a team could be supplied to the corresponding cluster's model to output the anticipated probability of a run. On the 40% of the original data partitioned as testing data, categorization tests could be performed using the model that corresponds to the cluster of the team being tested on.

The primary measures explored as models were tested included sensitivity and specificity. In the instance of this prediction, sensitivity represents the model's ability to correctly predict a run, while specificity represents the model's ability to correctly predict a pass. The sensitivity and specificity of models on all teams can be seen below:



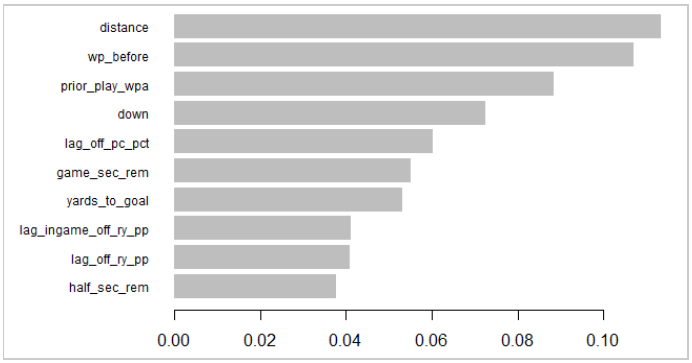
Using individual team data in tandem with overall prediction accuracy can allow for a true assessment of model accuracy. Though six unique XGBoost models were deployed in testing for accuracy, the results of each instance can be compiled for an aggregate confusion matrix. The results can be seen below:

Grouped Confusion Matrix			Accuracy	Sensitivity	Specificity
	Pass	Rush	0.665	0.728	0.599
Pass	9345	4414			
Rush	6254	11843			

As seen in the appendix in Figure 3, the cluster approach may not produce the most optimal accuracy. However, why this route was selected as the best model is because it creates a strong balance between sensitivity and specificity, which is most applicable and safe for a coach to use. For instance, being far more confident that a play is a run is only helpful to a certain extent; the runner will likely be tackled within a few yards regardless. However, for a pass play, the expected yards gained as well as potential for a breakout play is far greater; therefore, a higher specificity establishes a safeguard against high yardage plays by learning how to better predict passes achieved through clustering.

Discussion

The results provide actionable insight into variable significance and team predictability that can serve to adjust coaching styles. As for variable significance, the importance of variables in the dataset was explored, looking across all trials and models. These top 10 variables ranked by importance can be seen below:



These results were found to be rather intuitive. The three most important variables included distance to the first down marker, win probability of the team with the ball prior to the snap, and the win probability added by the previous play. Each of these are sensible given they reflect either the necessity for a big play, or the recency of a big play, which impact the choice of an offense tremendously. Also important were what number down the play was on, the offensive pass completion rate and other offensive success measures, the time remaining in the game, and distance to the goal line. These factors together adjust offensive momentum and willingness to take risks. Noting the variable importance in this order can assist in supporting or adjusting the intuition of coaches in assessing how the opposing team's offense will react to in-game conditions, perhaps reprioritizing assumptions.

Secondly, as for the predictability of a team, the lower the predicting accuracy of a team tends to indicate that the team is more erratic and unpredictable. This is because they deviate from the unwritten standards within their cluster to a greater extent, and thus perhaps act against the gradient of normality in football. The teams that suffered low predictability in the testing effort should serve as somewhat of a red flag to coaches in adhering to traditional football standards and expectations, warning them not to go all in on defense against an unpredictable team. On the other hand, teams with very high predictive accuracy may be noted as more routine in relation to the important variables in this dataset that may often pair with a coach's intuition; therefore, decisions on defense can be made more confidently.

Conclusion and Future Work

This report explored the relationships between team and in-game variables to the offensive decision on any given play, delineating between a rush or a pass. It became evident relationships surely exist, and differ among teams given their tendencies and characteristics. A prediction for a play can be usefully employed by a coach in adjusting defense for an opposing team's offensive expectations, as the relatively strong accuracy of a model paired with the general awareness and knowledge of coaches can ensure wise adjustments.

In the event of future work with this model, two additions would be prioritized. Firstly, a variable that would be incredibly useful in modeling would be offensive formation. For instance, noting that a team arranges themselves in a shotgun formation at snap can be quite indicative to the run or pass prediction. Identifying a dataset that contains this information or collecting information to supplement this dataset may increase predictive accuracy. Secondly, this modeling has large applicability in a setting with live data feeds. Before a game, the model itself may be constructed for a team; then, during the game, plays can be updated as they occur. Therefore, a prediction can be classified with a given confidence range right before the snap for the next play in the game, which would assuredly be the most useful output for a coach employing this sort of programming.

Contributions

Team Member 1: Zach Austin

Section	**Percentage Contribution**
Data Cleaning	70%
Modelling	70%
Result interpretation	50%
Presentation creation	30%
Report Writing	30%

Team Member 2: Jimmy Brunette

Section	**Percentage Contribution**
Data Cleaning	30%
Modelling	30%
Result interpretation	50%
Presentation creation	70%
Report Writing	70%

Additional Comments:

We felt that this project overall had an even workload. We believe that we adequately adjusted to each team members' strengths and passions to tweak work load within each category; however, we felt that each team member was informed enough on the happenings across areas where adjustments and suggestions could be made helpfully.

Bibliography

"An R Package to Quickly Obtain Clean and Tidy College Football Play by Play Data." *Saiem Gilani*, https://saiemgilani.github.io/cfbfastR/.

RahulJain28. "Rahuljain28/NFLPredictions: An Exercise in Existential Contemplation of the Pedagogical Virtues of Contemporary Interpretations of Machiavelli and Nietzsche." *GitHub*, https://github.com/RahulJain28/NFLPredictions.

Cheema, AuthorAhmed. "Predicting NFL Offensive Play-Calling with Python." *The Spax*, 23 Sept. 2019, https://www.thespax.com/nfl/predicting-nfl-offensive-play-calling-with-python/.

Appendix

Figure 1:

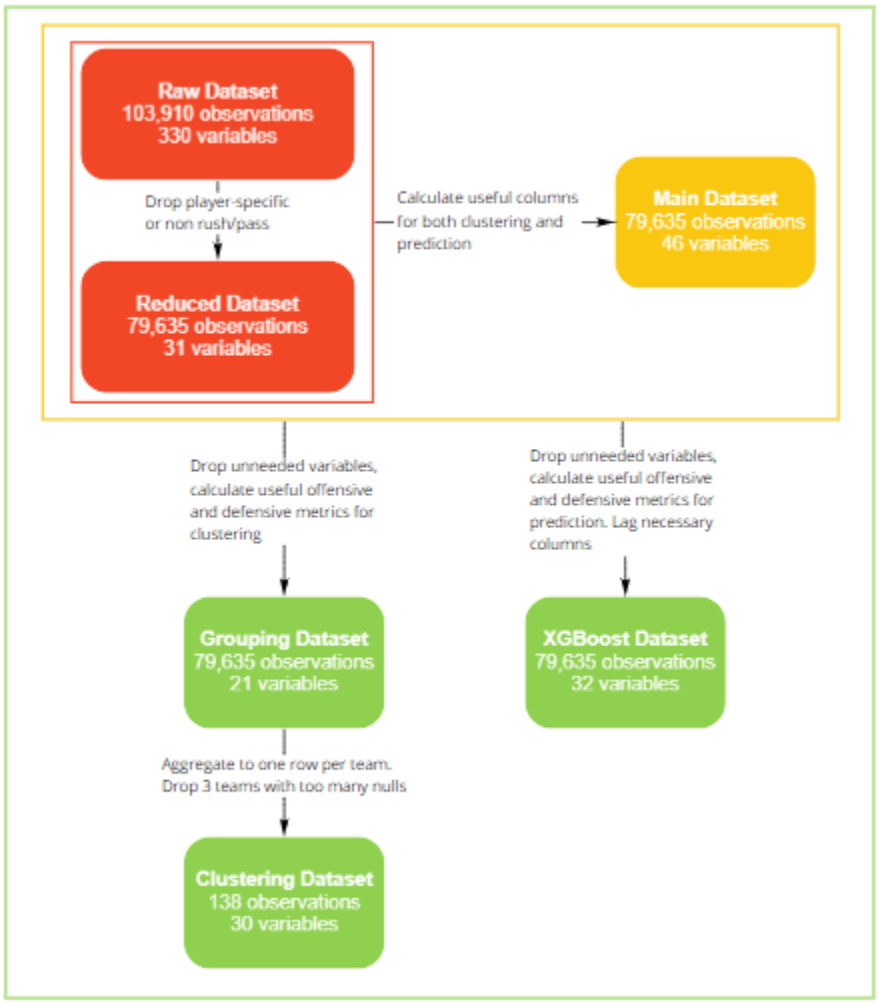


Figure 2:

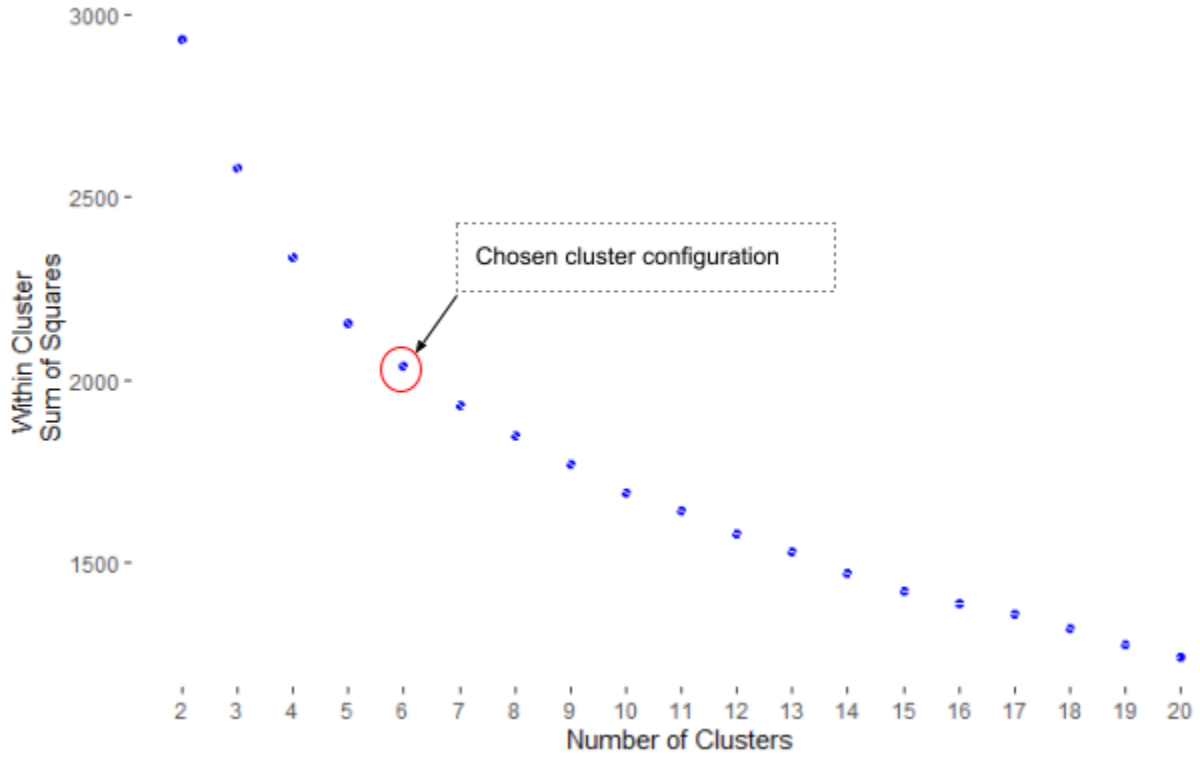


Figure 3:

Model	Process	Accuracy	Sensitivity	Specificity
Single Team	Train on Notre Dame plays, test on Notre Dame plays	0.680	0.902	0.393
All Teams	Train on plays from all teams, test on plays from all teams	0.669	0.761	0.573
Grouped Teams	Train individual models for each cluster, test using model corresponding with each team's cluster model	0.665	0.728	0.599