Using ML to Determine Efficiency and Predictability of Football Offenses

Vincent Etherton

December 2024

1 Abstract

American Football, at its core, is a sport that isn't immediately intuitive for data analysis or application of machine learning to assist in coaching and strategy. Up until now, coaching has relied upon copious film review combined with learned intuition of the game and its strategic nuances. As an offensive coaching staff, it is of utmost importance that plays are effectively called, not only in an attempt to beat the defensive look that they are facing, but to reduce predictability while increasing variability in the plays that they call. By building on previous route classification/clustering models and defining a successful versus unsuccessful offensive play, this paper suggests a methodology for clustering and labeling routes to then identify popular route combinations run by a football offense. From there, the paper outlines the utilization of a classification model to determine the likelihood of a successful play given information about the opposing defense.

2 Introduction

2.1 Motivation

As mentioned above, football scheming is largely based on learned experience and intuition. Dozens of hours at the collegiate level and presumably more in the professional scene are spent “drawing up the X’s and O’s” as it is called, or charting offensive plays to understand and fully develop an answer for any defensive scenario a football offense might face on game day. As this process is based on gameday or practice film, much time is spent first “tagging” the film, or compiling relevant information about each clip for more streamlined review. From there, unseasoned individuals such as myself make significant use of the replay button as they try to absorb what is happening with all twenty two players on the field. On average, there are 153 plays in an NFL game, so by the time the tenth week rolls around, teams scouting their opponent will have tagged and attempted to understand patterns in over 1,500 clips of film. From there, offensive coaching staffs will create their game plan for that week including plays from their staple playbook combined with plays designed to answer the team they are facing. At smaller collegiate-level schools, the data analytics process during this stage mainly consists of simple summary statistics, tabulating frequencies of formations run in different game scenarios whether that be down and distance or area of the field. Here, we find an opportunity for improvement as rather than simply summarizing the plays that were ran in what scenario, one step further would be to understand what made the plays that were run successful, what route combinations worked effectively, and ultimately how did the play that was called lead to a successful outcome.

2.2 Objectives

Having the goal of an efficient methodology where plays can be clustered, labeled, and finally classified based on their outcome, it is only natural to turn to applied machine learning to understand and try to tackle this problem. Thus, this paper sets out to answer the following guiding questions:

Given positional data of route-running football players, can a methodology be designed utilizing clustering and classification to identify the route combination that was run and determine its future effectiveness against a certain known defensive look? Can we ultimately use these findings to determine trends in an offensive scheme and guide the development of offensive strategy?

With these questions in mind, we visualize a multi-step process dependent on the type of data a given football team has access to. In the NFL, teams have access to thorough positional data of all 22 players on the field at any given time. With this positional data, I propose the following methodology:

1. Clustering and manually labeling the clusters based on the median route in each cluster and the relative start and end points of most of the routes in the cluster (clustering)

2. Understand summary statistics using these route combinations such as the most common combinations of routes in different game situations or the distribution of routes run by specific key players

3. Leverage regression analysis to predict the effectiveness of route combinations when facing certain defensive looks and classification analysis to predict route combinations involved in the play given game situation and defensive look

Many works in the following literature review section address different approaches to clustering or classifying routes. Other works outlined a potential methodology for the described regression analysis. This paper suggests clustering with no labeled data and no manual labeling until the end while many methods utilize partially or fully labeled training sets. It also suggests utilizing different defensive characteristics for a regression model to predict effectiveness and a classification model to predict route combinations involved in a play given a given game scenario.

3 Relevant Work

Two works submitted for the Big Data Bowl competition put on by the NFL explored similar topics of using positional data to determine the effectiveness of certain route combinations run in the NFL. Chu et al. explore a process of clustering routes using k-means clustering with Bézier curves, a methodology I relied heavily on to get quite accurate labels for the routes in my data. From there, they go over general statistics and trends in the league as well as a proposal for using Expected Points Added, or

EPA, and another method to determine receiver-controlled zones on the field of a given play to predict route combination success using classification (Chu et al.).

Nathan Sterken wrote a paper on the same Data Bowl dataset in which he trained a 'RouteNet' neural network to classify routes, a venture that I would have pursued given its benefits towards higher classification accuracy, except it requires manually labeling thousands of routes individually. Sterken then proposes a similar methodology of using EPA to determine route combination effectiveness (Sterken).

Both of these works cover significant legwork when it comes to making unlabeled positional data digestible allowing for individual route and route combination analysis. However, there is clearly a lack of progress towards a holistic methodology to analyze predictability and effectiveness of an offensive scheme that this paper attempts to address. Given these route labels determined by a clustering model more closely tied to Chu et al. 's methodology, we can use regression analysis and classification analysis to determine predictability and efficiency of various route combinations to make informed decisions on the future of a team's offensive strategy.

4 Methods

4.1 Datasets

- Play data from the 2021 NFL season containing play-level information from each game through 8 weeks of the season
- PFF Scouting data from the 2021 NFL season containing player-level scouting information for each game and play
- Tracking data spanning Week 1 through Week 8 of the 2021 NFL season containing player tracking data for each game and play for each week

4.2 The Relevance of Positional Football Data

The proposed methodology for summarizing offensive efficiency and predictability starts with positional data and relevant play information as an input. In the NFL, each venue is fitted with receivers and transmitters which communicate with RFID tags in the players' pads and NFL NextGen Stats powered by AWS makes data points such as position, speed, and time on the field available to NFL teams for further analysis (NFL Football Operations). NFL Operations mentions many uses for this positional data from scheming, which this paper addresses, improving player safety, finding new storytelling opportunities for broadcast, or rules improvements. At the college level, this positional data is also available, with companies like Catapult on the rise providing positional data collection for use in analytics. With companies offering affordable and approachable data gathering software at all levels of the sport, positional data is becoming more accessible to gain actionable insights about the performance of a football team.

4.3 Exploring and Preliminarily Transforming the Data

A couple of key transformations of the data are important to allow for further analysis. First, we group offensive plays into more generalized down and distance categories. The dataset originally has one column representing the down and another representing the distance to get a first down. We group plays into more general buckets consisting of the down followed by short, medium, long, and extra-long (XL). These buckets correspond to 'yards-to-go' values of 1-3 yards, 4-6 yards, 7-10 yards, and 11+ yards respectively.

Looking at features of the data that can be used to refine our positional data, columns like 'pff_role' and 'pff_positionLinedUp' will be important to track positions of relevant pass route runners. In each of the datasets, 'gameId' represents the game identifier which was unique to that season, 'playId' which was unique to that game, and 'nflId' which is a playerId, unique across all players that season. From the plays dataset, we eliminate all non-pass plays by filtering first by 'dropBackType'. The type of drop-back on the play describes the quarterback's movement post-snap before he gets rid of the ball. Any play that is not a traditional drop-back pass or a designed rollout meaning that the quarterback has a determined movement outside the pocket is eliminated. Plays are also eliminated that do not contain the string "pass" in the play description which is an English transcription of the play that has occurred. For example, "(13:33) (Shotgun) T.Brady pass incomplete deep right to C.Godwin."

The positional data in the dataset is set up on a traditional xy-plane ranging from 0 to 53.3 in the y-direction and 0 to 120 in the x direction. The x-direction represents progress down the field in the direction of the visitor team's endzone. To standardize this data, I propose a method using the absolute yardline number — the distance to go to the endzone for the offense — and standardize all x-positions relative to this yardline as a starting point. To do this, I take into account the relative direction of the play which is stored in the positional data; if the direction is to the right, the x-coordinates for a player should be increasing with forward progress and the opposite for a play progressing in the left direction.

Finally, I find it necessary that we standardize both x and y-coordinates throughout each route relative to the starting position. Since we already standardized to the relative line of scrimmage starting point, we can now just standardize with the first position in the route being zero and the x-positions stemming from zero. The y-coordinate standardization is more complex as I propose that we keep in mind that we might want to retain the side of the ball that the route was run from, namely which side of the center and offensive line that the players started from to identify route combinations. In the clustering model, I propose that we standardize the y-coordinates as well so they all stem from the same xy-coordinate, namely (0,0). Looking at the offensive features, columns such as 'absoluteYardlineNumber', 'yardsToGo', 'offenseFormation', 'personnelO', 'down', 'quarter', and 'gameClock' are all relevant features describing the situation the offense is faced with with coming up with a play. These features could be used to create a classification model for the predictability of route concepts run in different game situations. This paper will explore all of these as features for a classification model to attempt to predict route combinations run in different game situations for a given team.

Addressing the efficiency piece of the proposed methodology, this paper suggests 'playResult' as a response variable in a regression model to determine the efficacy of a route combination on a given play. Features of this model could include the previously mentioned offensive characteristics as well as standard defensive characteristics like 'personnelD', 'pff_passCoverage', 'pff_passCoverageType', and 'defendersInTheBox'. I will briefly summarize some of these more technical defensive characteristics. Personnel on defense speaks to the number of each defensive position on the field broken up into linemen, linebackers, and secondary or pass rushers, intermediate pass coverage, or secondary pass coverage. Pass coverage speaks to the coverage call of the defense which wouldn't be known by the offense at ball snap, but is often narrowed down to a couple of options before the play is called. Finally, pass coverage type covers broader buckets in which these more specific pass coverages are categorized. All of these aspects combined with scouting efforts the week before the game give the offense a holistic picture of general defensive trends of their opponent to hopefully allow for the most possible inference as to what defensive look the offense might be facing based on pre-snap information.

5 Analysis/Modeling

With the positional information and relevant play data, I propose the following pipeline to begin offensive reporting and analysis:

1. **Clustering routes**: Cluster routes ran utilizing a k-means clustering algorithm. First, transform a route to a Bézier curve made using control points throughout the route and then sample to determine a curve corresponding to that route. K-means cluster those curves and manually label based on visualization of the curves and visualization of the median curve of that cluster.

2. **Generate visualizations**: At this stage, generate visualizations identifying common routes run by starting route runners and popular route combinations utilized across weeks.

3. **Evaluate predictability with classification**: At this stage, leverage classification models to attempt to predict route combinations when faced with a given game scenario.

4. **Predicting efficiency with regression**: At this stage, utilize features on both offense and defense as mentioned in section to predict the average Euclidean distance between the receivers running a route and the nearest defender. A process that was outlined as a future possibility in Chu et al. 's work.

5.1 Clustering Routes Using Bézier Curves

Given positional data with x-coordinate, y-coordinate, speed, acceleration, and direction at each frame for each play, I build a model capable of clustering similar routes together. As mentioned in section methods for clustering routes currently exist. Chu et al. describe a Bézier curve approach that I attempt first. Bézier curves are mathematically described curves defined by a set of control points. The method outlined in Chu et al. describes sampling 200 evenly spaced xy-points from the Bézier curves. I leverage the bezier Python library and create Bézier curves with eight evenly spaced control points across the routes and resample to 50 xy-points, experimenting with simpler Bézier curves hoping to keep the dimensionality of my model features down to prevent over-fitting. Bézier curves are then represented as a vector with the following format, $[x_1, y_1, x_2, y_2, \ldots, x_{50}, y_{50}]$. First we attempt the method described in Chu et al. 's paper where routes are clustered using a k-means clustering model made up of 50 clusters. After the Bézier curves are clustered, we can use general intuition of football routes combined with looking at the general trend of routes in a given cluster to manually label each cluster. While there are not 50 routes in a common route tree, these 50 clusters could then be manually combined together into buckets of football routes.

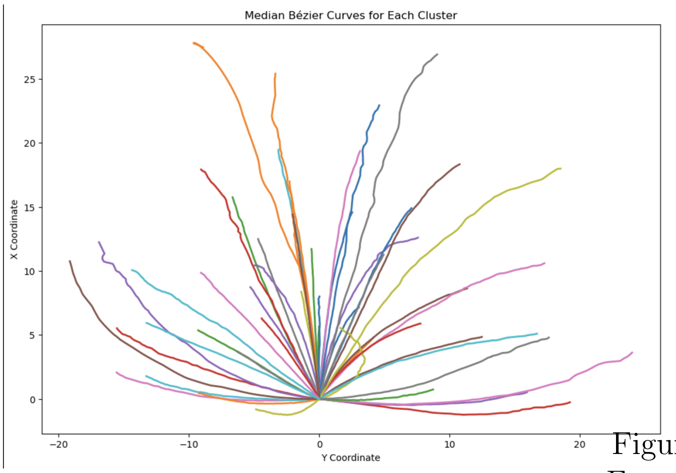

Figure 1: My graphic of 50 median routes for each cluster resembles different levels of the routes shown in an NFL route tree.

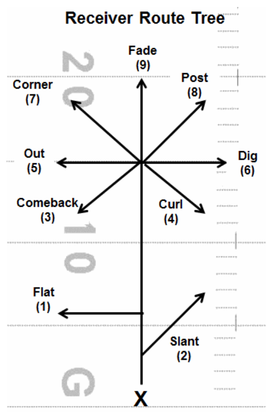

Figure 2: A common NFL route tree. Fantasy Footballers. RouteTree, 2018, `https://fantasyfootballers.org/wp-content/uploads/2018/09/RouteTree.png`. Accessed 6 Dec. 2024.

While we are dealing with strictly unlabeled positional data and routes are not known for any of our trajectories ahead of time, we can use the concept of silhouette score to evaluate our clustering model. The silhouette score or silhouette coefficient is a method for determining cluster accuracy by taking the mean intra-cluster distance, we'll call a, and the mean nearest-cluster distance, we'll call b, for each sample and performing the following calculation $(b-a)/\max(a, b)$ ("sklearn.metrics.silhouette_score"). Using a simplifying example with two clusters and two 2D points, a would be 0 while b is some number so the silhouette score would be at a max of 1. Another trivial example of two Gaussian clusters that overlap significantly would have an intra-cluster distance of some number and the nearest-cluster distance would be close to zero. This sets up our range for a silhouette score that captures a relationship between how close elements of a cluster are to each other and how distinct each cluster is.

With our first proposed model of simply accounting for the Bézier curve path, we obtain a silhouette score of 0.188 on our training data. With a visual test, as well, it is clear that the clusters are not capturing a lot of meaningful characteristics of the routes that are being classified. It is clear that if we want to utilize these simpler Bézier curves than the ones that Chu et al. proposed, we need to introduce new features to understand more about the routes we are attempting to cluster.

Based on intuition, important features to add to a route clustering algorithm would be speed and acceleration characteristics, final route position, total distance travelled, or length of the route stem (the length of travel before the first significant shift in trajectory). These all have their benefits. Speed and acceleration determine routes like comebacks or curls where at the end of the route, there is a significant change in speed as receivers break down and pivot in another direction. Also, what will below be classified as either a 'mid' route or a 'go' route representing a forward-moving straight route of either around 5-10 or 10+ yards respectively. While these are all intriguing options to add to features in a clustering algorithm, I propose we use final route position in conjunction with Bézier curves to ultimately cluster our routes. This addresses a key issue with solely using Bézier curves as they cannot differentiate commonly based on direction so a 5-yard in could be clustered with a 5-yard out as they are the same route just in different directions or a 5-yard out versus a corner which is 5-yards followed by a 45 degree pivot to the sideline. Adding a clustering component such as final position allows similar depth routes to be clustered together and allows for these differentiations to be much more apparent.

5.2 Clustering Routes Adding Final Position Clustering

I propose that we create two clustering models that use the k-means algorithm to cluster routes. Ultimately, we can combine the clusters so that 'cluster i_j' represents routes in the i-th cluster when taking into account their Bézier curve and the j-th cluster when taking into account their final position. Since we have to determine the number of clusters for each model, we will use a similar method to that outlined in M A Syakur et al. 's research in which the optimal number of clusters in a clustering algorithm is determined with an elbow plot. In this method, we plot what is called the inertia curve for each option for k (the number of clusters) and take the elbow point where increasing k any further would have continually diminishing return on clustering performance, but any decrease in k would be significantly detrimental to the model. Inertia describes how well a dataset was clustered by k-means clustering, and it is a field of a fit k-means model. So, a downward sloping curve is generated as we iterate through possible k values. Looking at the plots for inertia vs. k-value, we observe that the elbow of each plot is around k=10.

We use scikit-learn's standard scaler first independently on both the Bézier curves and the final positions and then we fit a $k - means$ model with ten clusters to the training set of both the Bézier curves and finalpositions separately. Then, for each route, we combine the labels of the two clusters into one label that can be used to identify the route cluster. Using this method, there are 67 clusters created. It is worth noting at this point that hierarchical clustering or another similar method for allowing multiple features in the clustering process would be necessary as with each clustering model we add, the combination of cluster labels would lead to too many clusters and perhaps overfit the data. Using the same silhouette score as above, the silhouette scores on the testing set are 0.260 and 0.331 on the Bézier curve model and the final position model respectively. Already, we see an improvement and visually I observe more consistency with the directionality of the routes, yet I notice that route depth is still highly variable. Nonetheless, I propose utilizing the cluster median route to determine the label of the routes in the given cluster. When I feel that the median is less representative of the cluster as a whole, I look at the visual average of the routes in the cluster to determine the route names.

As a sanity check, we can perform two interesting experiments to evaluate how our clusters have performed. First, I suggest plotting random plays and calculating the average percentage of correct routes on each play. It is important to consider that these route labels are only convenient and useful in conjunction with the other routes in the play. Ultimately, I propose this is a valuable statistic to determine the clustering model's efficacy in its clustering and our efficacy in cluster labeling. Next, I suggest taking subsets of players in the NFL from different positional groups and different teams and performing basic visualizations on their commonly run routes to determine whether or not our clustering algorithm works as intended. Both of these approaches require some intuition in what routes are run when and for what purpose, but both should indicate generally how our model performs. With a successful model to cluster unlabeled routes run in the NFL, we can continue with our offensive reporting and analysis by understanding when these routes and namely the combinations of these routes that are run and in what situation.

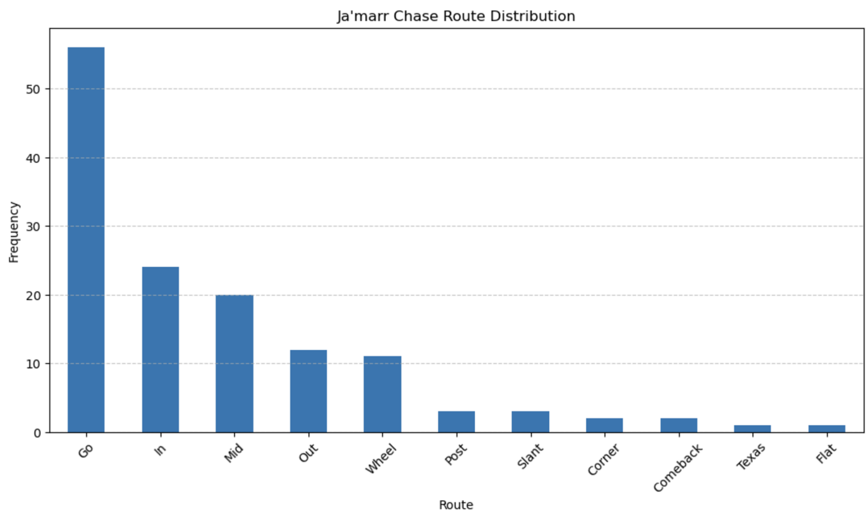

Figure 3: Ja'marr Chase is a deep threat, tall receiver that is often put on go-routes to draw defenders away. He is also among the best in the NFL, so often the Cincinatti Bengals will design plays where he'll take defenders with him out of contention for the ball.

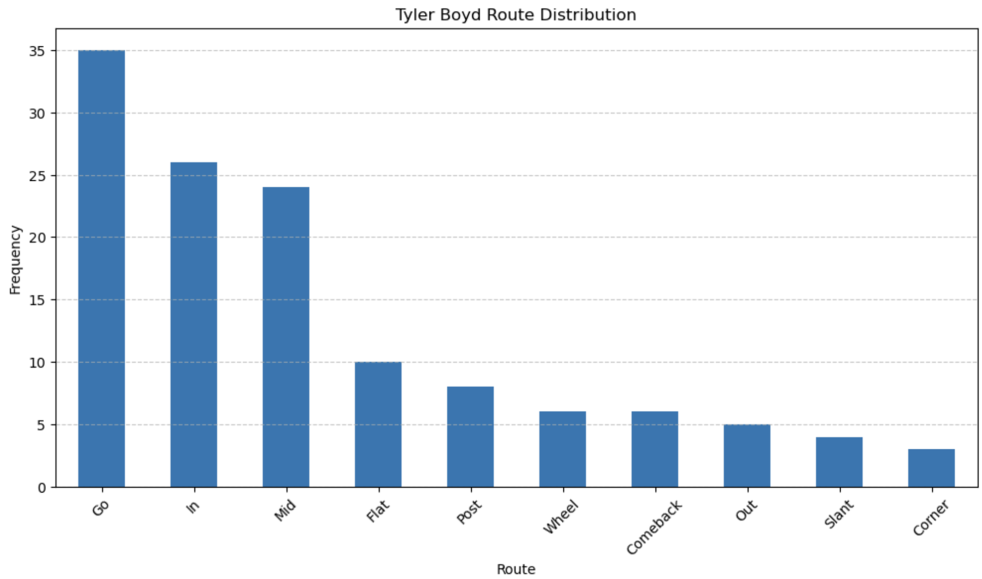

Figure 4: Tyler Boyd also plays for the Cincinatti Bengals but is more of a utility receiver. It is a good sign that the model was able to differentiate significantly between these two as he has a lot more variation in his routes run.

5.3 Evaluating Route Combination Predictability with Classification

Understanding when route combinations are run in certain game situations is a vital concept for opposing defenses to allow for less reactive and more proactive thinking on the field during a game. Oftentimes, route combinations 'flood' certain zones of a defensive coverage making a defensive player in coverage make an impossible decision as to who to cover, often leaving someone open to catch a pass. Examples of this are concepts like post-wheel and slot-fade as pictured in Figure 5 which are two examples of what is called a high-low or an opportunity for the deep-level defender to need to choose between following the receiver going for a longer route or bite down on the short-level receiver leaving the opportunity for a long pass open. Understanding the likelihood of being faced with a certain route combination before the play even starts is vital towards being proactive with defensive preparation. From the other point of view, offenses desire to be completely unpredictable showing no clear patterns in the face of certain game situations or defensive looks. It is because of this I propose the following continuation to our methodology of developing a classification model to predict a route combination that will be run in a specific game situation given relevant pre-snap information. In this circumstance, a high predictability might not be the best for a team in the NFL as experienced players and coaching staff could leverage this to get a competitive edge. Nevertheless, we can determine some features that play an important role in predicting route combinations run on a given offensive play.

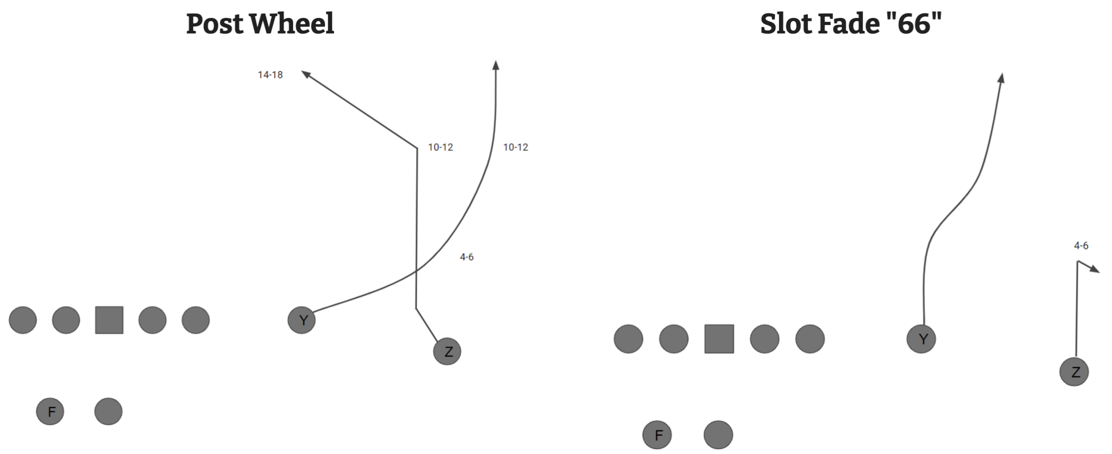

Figure 5: A post wheel and slot fade concept which both drag the deep defender away from the route intended to be the target.

First, we must transform our data to allow us to perform classification to predict route combinations, which begins with determining route combinations on plays in the first place. In the world of football scheming, there are two-man, three-man, and even four-man route concepts where each route is intentionally run to lure away enough defenders to cause at least one open man to receive the ball. In the simple application of this model, we will only predict one-man routes and two-man route combinations.

There are a couple of different intuitive approaches to determining route combinations involved in a given play. First, we can look at the starting positions of the routes to determine which routes were designed to go together. In the two route combinations we discussed earlier, the post-wheel and the slot-fade, this would certainly hold true as these remain on the side they started on and work together to create vertical separation in the defense. However, in other examples, we have routes that start on opposite sides of the offensive formation working in tandem to separate or confuse the defense. Some routes, rather than stretching the defense vertically, creates confusion in the middle of the field as four or more players are navigating the tight brushing of the two receivers often leaving one open on the other side. If we used solely starting positions, this route combination wouldn't be captured. Instead, I propose that we create route combinations based on the closest and second closest pairing of routes according to the y-coordinate at the end of the route. The closest y-coordinate pair most likely represents an important route combination on the play and the algorithm assigns the pairs iteratively. If they are running an empty set — meaning that there is no receiver or running back in the backfield — or a set with only three receivers lined up, there will obviously be an extra route in our pairings, so this route is considered its own combination itself. We now have a target variable to predict based on the game situation that an offense is facing.

I propose that we make two transformations to our existing data to develop features to create a classification model to understand the different game situations in which certain route combinations are run. First, I one-hot encode the offensive formation variable which is a categorical variable indicating the structure of the offense including categories like shotgun, I-form, and pistol which speak to the alignment of the running back and the quarterback. This process is done with scikit-learn's built-in one-hot encoder, but essentially it converts a categorical variable into multiple columns with 0's or 1's corresponding to the correct category that row fell under. From there, we capture the game situation given variables such as 'quarter', 'minutes_remaining', 'down', 'yardsToGo', and 'absoluteYardlineNumber'. 'Minutes_remaining' was a variable I created from the string variable corresponding to the time remaining in the quarter in "MM:SS" format. Regarding the defense, I included the variable 'defendersInBox' which encapsulates the number of defensive players who are not deep back or off to the sides, something that can be measured pre-snap. Presumably, offenses would be calling their plays or even changing them pre-snap based on the defenders in box characteristic, so intuitively it has some bearing on play-calling and therefore route combinations. Finally, I included three categorical variables corresponding to the number of running backs, wide receivers, and tight ends at the time of the snap, as this intuitively correlates to the combinations run on a play.

5.4 Building and Evaluating Classification Models for Route Combinations

I first tested multiple classification models with my intuitive features to determine which might be most successful at understanding important aspects regarding route-combination choice pre-snap. The training and test accuracies are reported in Figure 6. Accuracy is the most basic measure for effectiveness of a classification model, as it is simply the ratio of correct predictions to total predictions. On applicable models like random forests and the decision tree model, we use grid search to tune hyperparameters to maximize accuracy. As we can observe, the random forests model has the highest accuracy on both the test set and the training set, however it seems to overfit to the training set a bit too much and poorly generalize to the testing data. The logistic regression model seems to be the best overall at understanding what goes into predicting the route combination on a given play in the training set and generalizing its findings to the test set. The accuracy at first glance is a little low, however, when thinking in context of a complex game like football, any small edge in the predictability of an offense in certain game situations is of utmost importance and can be combined with seasoned veteran coaching experience. We are also attempting to determine the route combination to be run against a defense from over 70 different route combinations, so the model does fairly well compared to a random choice which would be around 1%. The model could definitely be expanded upon, however, as certain route combinations could be condensed into broader buckets. I uncovered an issue that certain route combinations were too rare and the model didn't have enough experience modeling these. Generally route combinations can be broken into larger categories like high-low, crossing routes, and others that might increase your accuracy.

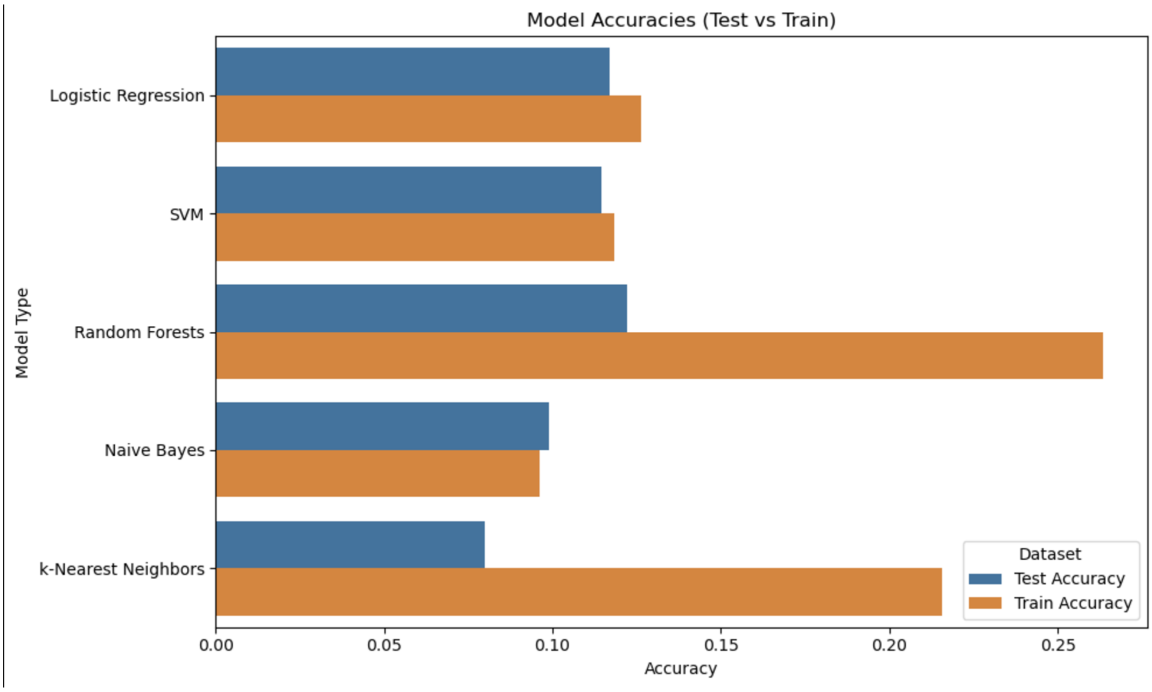

Figure 6: A plot of accuracy across model types for both the training and testing set. The 'best' model is found to be the logistic regression model with an accuracy of around 12%.

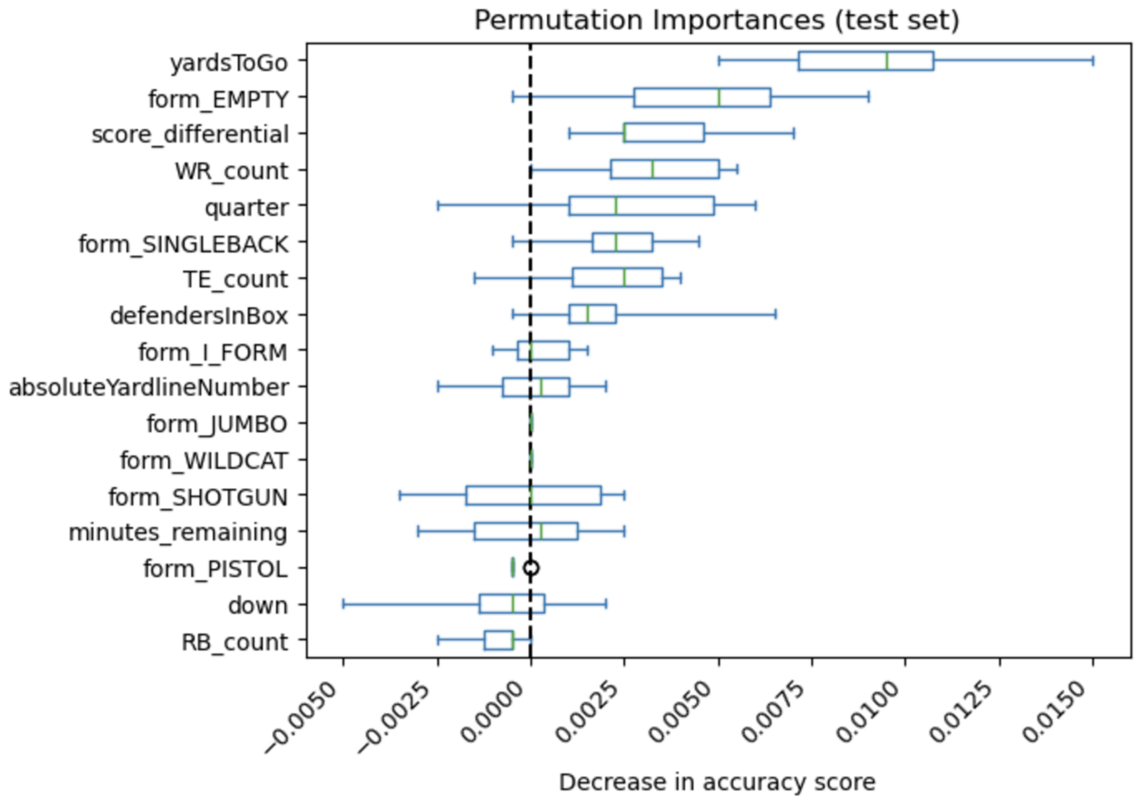

Figure 7: This figure illustrates the decrease in accuracy score if certain features were removed from the model. We can see that as yards to a first down, score differential, and quarter are eliminated from the model, the accuracy goes down the most indicating that what route combinations are run rely heavily on the timing during the game and the situation the offense is in.

5.5 Predicting Efficiency of Route Combinations with Regression

Finally, to address the second performance metric, I suggest that we analyze the efficiency of route combinations with a regression analysis. I propose a metric for route combination success not mentioned by Chu et al. which would be the result of the play, namely the yards gained on a given pass play. We could predict this outcome taking into account the defensive pass coverage run on the play, whether that coverage was man or zone, the defensive personnel on the field, and the defenders in the box.

However, similar to the model proposed by Chu et al. we will train a Ridge regression model to determine coefficients of route combinations. A positive coefficient coincides with a general positive effect on yards gained on average. A negative coefficient means the opposite that it on average has a negative impact on yards gained, or is inefficient.

The coefficients for the positive effect plays are as follows:

- Go-Go: 0.336
- Out-Post: 0.328
- Go: 0.281
- Mid-Post: 0.239

The coefficients for the negative effect plays are as follows:

- In-Slant: -0.399
- Mid-Slant: -0.269
- Corner-Slant: -0.241
- In-Mid: -0.237

I suggest that these findings prove to be a sanity check for the model as these are quite intuitive. Route combinations with go-routes are likely to be big result plays if they succeed, but this may point out a flaw in the sole usage of play result as a target variable. Go routes aren't always the best for targeting themselves, but they always draw defenders away from the receivers who really intend to be thrown the ball. Other routes like Out-Post and Mid-Post make sense as the post route is taking away the defender and the out/mid is receiving the ball. Understanding the amount of space each route combination gives between each offensive player and each defensive player on average could be an important metric going forward to predict with a regression model, an idea for future research.

6 Future Work

One significant drawback to film analysis in the NFL and in college football is the necessity to tag what is called 'all-22' film, or film shot from the side angle capturing all 22 players on the field. Timothy Lee from Stanford University published a paper illustrating the rise of computer vision in football. He demonstrates a pipeline in which relevant positional data can be determined from all-22 film. I envision a future in which this pipeline can be used in conjunction with a methodology as I defined in this paper to determine positional data of players in the film, cluster and eventually classify routes run on offensive pass plays, and perform regression or classification analysis on route combinations to come to important conclusions about the efficiency and predictability of an offensive scheme. Future areas of work also include more clearly defining one and two man pass concepts and eventually progressing to identifying full field route concepts. While there is a lot of creativity and originality involved in football offensive scheming, many coaches still rely on the tried and tested route combinations and pass concepts allowing for significant analysis.

7 Conclusion

In this paper, I've proposed a framework for how to effectively evaluate a football offensive scheme when given infinitely valuable play-by-play positional information. I first propose that we can leverage k-means clustering to cluster and manually label routes run on a given pass play. From this point, we have labeled routes and can determine combinations of routes that were run on each play and derive important insights about both their effectiveness when faced with a certain defensive situation and their predictability when it comes to the current game situation the offense is faced with. Both of these insights are meaningful to the team itself and the opponent team. If I am creating my own offensive scheme I would focus more on the effectiveness metric to determine which of my route combinations are most successful against what defensive looks. On the defensive side of the ball, I might want to understand what combos teams run in different game scenarios with a certain amount of confidence and what combos they run well against defensive looks I may throw at them. Overall, improvements can most certainly be made to this rudimentary pipeline, but this lays down the groundwork to really understand offensive tendencies while giving already seasoned coaches an extra edge when preparing their strategies and philosophies for such a complex and methodical game.

8 References

Chu, Jiarui, et al. 2021. NFL Big Data Bowl. https://www.semanticscholar.org/paper/NFL-Big-Data-Bowl-Chu-Wu/b6cec04d7158fdbb1db5afdb80682a15da072bd4. Accessed December 6, 2024.

NFL Football Operations. NFL Next Gen Stats, NFL, https://operations.nfl.com/gameday/technology/nfl-next-gen-stats/. Accessed 6 Dec. 2024.

Shi, Congming, et al. "A Quantitative Discriminant Method of Elbow Point for the Optimal Number of Clusters in Clustering Algorithm." EURASIP Journal on Wireless Communications and Networking, vol. 2021, Article 31, 2021, https://doi.org/10.1186/s13638-021-02035-7. Accessed 6 Dec. 2024.

"sklearn.metrics.silhouette_score." Scikit-learn 1.7.dev0 Documentation, 2024, https://scikit-learn.org/dev/modules/generated/sklearn.metrics.silhouette_score.html. Accessed 6 Dec. 2024.

Sterken, Nathan. RouteNet: A Convolutional Neural Network for Classifying Routes. NFL Operations, Big Data Bowl.