Background Information

In this project, I set out to explore how machine learning can be applied to predict the outcome of football pass plays. With data from professional games, my goal was to determine if a pass would be completed based on a variety of features, such as the game clock, offensive formations, and down. By analyzing these factors, I aim to provide insights that could help coaches and analysts make more informed decisions during games.

I have previously worked on various data science projects across domains like natural language processing in political science and sports analytics, but this project focuses specifically on pass prediction. By narrowing the scope to a single aspect of the game, I hoped to improve the accuracy and interpretability of the model, contributing to more actionable insights for in-game strategy.

Problem Statement

The purpose of this project is to predict whether a pass will be completed based on pre-snap features in football games. This is a binary classification problem, with the target variable being whether the pass was completed (1) or not (0)

Methods

To begin, I cleaned the dataset and engineered some new features. isNearOwnSide was a metric I developed that would contain a value 1 if the play was run near the offense's own sideline, and 0 otherwise. gameClock was converted from time left in the quarter to time_left_in_game, which represented the amount of time left in the match as a whole. Finally, I had to engineer a feature that would allow us to distinguish between pass and run plays, so isPass was engineered based on if a play had a value for passResult. Irrelevant features such as the gameId were dropped, and categorical features such as offensiveFormation were one hot encoded, ensuring that they could be processed by a logistic regression.

Afterwards, I standardized the features using StandardScaler to ensure they were all on the same scale, which is particularly important for models like logistic regression. This process helps to prevent features with larger scales from disproportionately influencing the model. Right before training, I used a correlation matrix to drop features that had high correlation with one another, and also dropped any rows with NA values.

I decided to focus on logistic regression due to its simplicity and the clear interpretability it offers for binary classification tasks. I used LogisticRegression from the sklearn library and performed 5-fold cross-validation to evaluate the model's performance. Cross-validation allowed me to assess how well the model would generalize to unseen data and reduce the risk of overfitting.

Results and Discussion

The logistic regression model performed well, with an average cross-validation score of 0.74, suggesting a reasonable ability to predict whether a pass would be completed. The confusion matrix indicated that the model had a recall score of 0.82, meaning it was quite effective in predicting successful passes. I chose recall in order to minimize false negatives. However, there were still some false negatives—plays where the model predicted a run but it was actually a pass. In a game, this error could result in a disaster for the defense, as committing to defending the run would draw significant manpower away from pass defense, potentially allowing for a large gain by the offense. Conversely, pass defense against a run play is generally less punishing, as defenders are often better positioned to react quickly to runs compared to the reverse.

Moving forward, I could explore more advanced models like XGBoost or Random Forest, which may capture more complex interactions between features and further improve recall, as well as balance the trade-off between false positives and false negatives.