**Final Report – NFL Success and College Conferences**

**Problem Statement**:

The primary focus of my project was to investigate the impact of college conference affiliation on a player's success in the National Football League (NFL). I aimed to answer the question: "Does the level of competition in a college conference influence a player's success in the NFL?" I was curious to see if players who compete in more competitive college conferences may be better equipped to thrive in the demanding environment of the NFL.

**Objective**:

The overarching goal was to analyze whether there is a correlation between the competitiveness of a college conference and the success of its players in the NFL. To achieve this, I employed machine-learning models that utilized a player's statistics during an NFL season to predict the college conference they played in.

**Rationale for Multiple Models**:

The decision to explore multiple machine-learning models stemmed from the complexity of the data and the desire to ensure robustness in the analysis. Different models offer diverse perspectives on the relationships within the data, allowing for a more comprehensive understanding of the factors influencing a player's success in the NFL. I wanted to enhance predictive accuracy, uncover variable importance (if any), and evaluate model robustness – which are made easier when fitting and tuning multiple models.

**Exploratory Data Analysis (EDA)**:

As I explored my data, I had to tweak the data I was using and how I was using it many times. I went between multiple datasets, finally deciding to define success as yards per season, touchdowns per season, and touches on the ball per season. This allowed me to get a feel for how much players were actually playing in games. I originally calculated ratings for wide receivers and running backs like quarterbacks have, but it proved insufficient with the data I was given based on how player’s positions were listed. To simplify things, I stuck to those three metrics to define “success.” I created plots to understand the data more fully by comparing these metrics between college conferences. (Shown on the next page)

Because I had multiple columns detailing success, I decided to predict the college conference of a player based on their stats. This allowed me to use multiple variables to predict college conferences in hopes that it would lead to more accurate predictions.

A graph of a graph showing the results of a conference

Description automatically generated with medium confidenceA graph of a college conference

Description automatically generatedA graph of college conference

Description automatically generated with medium confidence

**Overview of Models Tried**:

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Description | Key Hyperparameters Explored | Highest-level Results |
| Logistic Regression | Linear model suitable for binary and multiclass classification | N/A | F1: 0.213018  Recall: 0.250452  Precision: 0.347983 |
| Random Forest | Ensemble learning method using multiple decision trees | First model: N/A  Second model: n\_estimators: [100, 200, 300]  max\_depth: [5, 10, 15]  max\_features: ['auto', 'sqrt', 'log2'] | F1: 0.218648  Recall: 0.223359  Precision: 0.216082  F1: 0.194454  Recall: 0.254064  Precision: 0.341162 |
| SVM (Support Vector Machine) | Classifies data points by finding the hyperplane that best separates classes | C: [0.1, 1, 10, 100]  gamma: [0.001, 0.01, 0.1, 1] | F1: 0.219909  Recall: 0.272125  Precision: 0.383279 |
| AdaBoost | Boosting ensemble method that combines multiple weak classifiers to form a strong classifier | n\_estimators: [50, 100, 200, 500]  learning\_rate: [0.01, 0.1, 0.5, 1] | F1: 0.158966  Recall: 0.224564  Precision: 0.400537 |

**Model Selection**:

1. Tree-Based Models Performance:
   * Both Random Forest models had lower F1 scores compared to other models, suggesting challenges in capturing the complexity of the relationship between NFL success and college conference affiliation.
   * Random Forest, even after hyperparameter tuning, did not significantly improve its F1 score, indicating potential limitations in its ability to enhance predictive performance here.
2. Ensemble Models vs. Single Models:
   * AdaBoost, an ensemble model, showed relatively lower F1 scores compared to individual models like Support Vector Machines, Logistic Regression, and Random Forest.
   * The ensemble approach did not consistently outperform individual models in this case.
3. Support Vector Machine Performance:
   * The Support Vector Machine model demonstrated competitive F1 scores, recall, and precision, indicating its effectiveness compared to the other models in capturing the underlying patterns in the data.
   * Comparatively, the model showcased a balanced trade-off between precision and recall.
4. Discarded Models:
   * The Decision Tree model was discarded due to poor performance on validation data and difficulty in hyperparameter training.
5. Challenges and Pitfalls:
   * When trying to fit models to a multiclass classification problem, there were a few problems with finding the balance in hyperparameters. As I was able to adjust and add more data, however, the models performed better and better fit the data.

**Best Performing Model: Support Vector Machine (SVM)**

The Support Vector Machine model demonstrated the highest overall performance among the models evaluated. The hyperparameters tuned were C (the regularization parameter): [0.1, 1, 10, 100], and Gamma (the kernel coefficient): [0.001, 0.01, 0.1, 1]. The tuning process involved systematically exploring different combinations of these hyperparameters through a grid search to identify the configuration that maximized the model’s performance. This configuration was C = 10, and gamma = 0.1. SVM models do not inherently provide feature importance like tree-based models, but it’s possible to analyze the coefficients of the support vectors and look at permutation importance. For this, I analyzed the SHAP values of another model to visualize the features and give some sort of idea of how they all work together:

A screenshot of a computer

Description automatically generated A screen shot of a graph

Description automatically generated

Below is a visualization to better understand the model’s performance. The confusion matrix is a representation of the model’s performance in terms of true positives, true negatives, false positives, and false negatives.

A graph of a number of blue squares

Description automatically generated with medium confidence

SVMS are powerful, but they’re less interpretable compared to linear models or decision trees. For this model, it’s obvious that Yards and Touches impact predictions the most. We don’t know much more, however, but interpretability can be enhanced through techniques such as:

* Kernel Analysis: Understanding the effect of the chosen kernel on the decision boundary. For example, a radial basis function (RBF) kernel may create a more complex decision boundary than a linear kernel. The kernel used in this model was the RBF kernel.
* Support Vector Analysis: Identifying and analyzing the support vectors, which are the data points crucial for defining the decision boundary.
* Feature Importance Techniques: Permutation importance can provide insights into the features that contribute most to the model’s predictions. The permutation feature importance is defined to be the decrease in a model score when a single feature value is randomly shuffled. This procedure breaks the relationship between the feature and the target, so the drop in the model is indicative of how much the model depends on the feature. I wasn’t able to perform this successfully on my model, but this would be something that would be helpful to further explore in the future.

The SVM model’s superior performance suggests its efficacy in capturing the underlying patterns in the data compared to other models, but It still doesn’t do an amazing job. It’s likely that success in the NFL is not correlated to college conferences strongly enough, making it hard for models to have success in predictions.

**Conclusion and Next Steps**:

The Support Vector Machine was the best-performing model in predicting college conference affiliation based on NFL player success. It had a superior F1 score, recall, and precision compared to the other models, but the scores still weren’t high enough for us to say that these two things are definitely correlated. With this in mind, it would be good for collegiate athletes to know. If an athlete is choosing between being mediocre and riding the bench for a season or two at a school in the SEC compared to being the star at a school in the Big 12, they should know that a better conference likely doesn’t affect their potential success in the NFL.

In the future, I would like to explore what “success” could mean in a data set. It would be helpful to create a ranking system with accuracy, team wins, etc., but it would be difficult to create a grading system that every type of player could be on. Alternatively, there could be multiple models – one for each position, so that new variables don’t mess up other types of players. For models, exploring more ensemble techniques that combine the strengths of multiple models could be helpful. Stacking or blending models could potentially improve predictive performance because of the diverse perspectives offered by different algorithms.