Introduction/Background

This project predicts NFL game outcomes using machine learning, building on existing sports prediction models by leveraging historical game data with features like team stats, player performance.

Literature Review

Several studies have employed machine learning techniques to predict NFL game outcomes. One study utilized a Bayesian linear regression model to estimate an individual player's impact, controlling for the influence of other players on the field, and incorporating a wide range of performance indicators over an extended timeframe [1]. Another study replaced Total Play Margin with values of a new marginal statistic, which were then input into point spread and logistic models using OLS regression, logistic regression, discriminant analysis, and proportional odds models [2].Research from the University of New Hampshire found that linear regression models were the most accurate, by modeling the relationship between various independent variables and a dichotomous dependent variable [3].

Dataset Description

The dataset contains various attributes regarding NFL game characteristics. These key features include:

- **Team win percentage**
- **Pass yards**
- **Sack yards**
- **Rush yards**
- **Amount of downs**
- **Interceptions**
- **Fumbles**

Dataset Link

https://www.kaggle.com/datasets/cviaxmiwnptr/nfl-team-stats-20022019-espn

Problem

Problem - Predicting the outcome of NFL games is complex due to numerous factors.Accurately forecasting wins and losses using machine learning poses a significant challenge, but it can offer valuable insights for fans and analysts.

Motivation - The motivation behind this project is to create a model that maximizes the excitement and entertainment value of NFL games by providing accurate score predictions. This can help fans and analysts better understand team performance trends and make better decisions on aspects like attendance and game expectations.

Methods

Data Preprocessing Methods Implemented

- **Missing Data Handling**: Filled missing values with the mean of numeric columns to maintain dataset integrity.
- **Feature Engineering**:
  - Created a new win column indicating if the away team won or lost.
  - Converted possession times to total seconds for consistent numerical analysis.
- **Data Cleaning**: Removed duplicates to ensure each game is uniquely represented.
- **Feature Selection**: Retained key performance metrics for training the model.

ML Algorithms/Models Implemented

Logistic Regression (Supervised)

- **Reason for Selection**: Chosen for its simplicity and speed in training, making it an excellent baseline model for predicting binary outcomes (win/loss).
- **Insights Provided**: Offers insights into feature importance and the type of relationships between features and game outcomes (e.g., positive or negative influences).

Random Forest (Supervised)

- **Reason for Selection**: Useful for its ability to model complex, non-linear relationships and interactions between features while reducing overfitting and providing robust predictions.
- **Insights Provided**: Helps identify the most influential features, such as possession time and turnovers, and offers interpretable results through feature importance, helping to understand key factors driving NFL game outcomes.

K-Means Clustering (Unsupervised)

- **Reason for Selection**: Useful for identifying natural groupings in the dataset without prior labels, helping uncover hidden patterns in game data.
- **Insights Provided**: Helps group games based on feature similarity, which can assist in understanding game patterns and trends.

Results and Discussion

Visualizations

Here is a link to our visualizations: Final Visualizations

Analysis of Quantitative Metrics:

Logistic Regression

- **Accuracy (0.9000)**: This high value means that the model was able to correctly predict the outcome of the NFL match 90 percent of the time.
- **Precision (0.8817)**: Among the games the model predicted as winners, 88.17 percent of them were actual winners, indicating relatively high precision.
- **Recall (0.88)**: The model correctly identified 88 percent of actual winners from all the games, proving its effectiveness.

- **F1-score (0.8823)**: The balance between precision and recall is 88.23 percent, showcasing great performance.

Random Forest

- **Accuracy (0.8735)**: This value indicates that the model correctly predicted the outcome of NFL matches 87 percent of the time, showcasing strong overall performance.
- **Precision (0.8734)**: Among the games the model predicted as winners, 87 percent of them were actual winners, reflecting a high level of precision and reliability in its predictions.
- **Recall (0.8735)**: The model correctly identified 87 percent of actual winners from all the games, demonstrating excellent effectiveness in capturing true winners.
- **F1-score (0.8730)**: The balance between precision and recall achieved by the model is 87 percent, highlighting its robust and well-rounded performance.

K-Means

- **Accuracy (0.67)**: The clustering algorithm properly grouped similar outcomes 67 percent of the time, showing slightly above-average performance but room for improvement.
- **Precision (0.61)**: Of the predicted clusters for winning teams, 61 percent were actual winners, indicating moderate results but needing improvement.
- **Recall (0.68)**: The model successfully captured 68 percent of all actual winning team clusters.
- **F1-score (0.64)**: The balance of precision and recall attained is 64 percent, indicating decent performance.

Analysis of Algorithms/Models:

Logistic Regression

- **Description**: Logistic regression is a supervised learning model that applies a logistic function to classify a binary target variable, and it can be adapted for multiclass classification tasks.
- **Mechanism**: It calculates probabilities using the logistic function, making it effective when a linear decision boundary exists between classes.
- **Optimization**: The model’s coefficients are optimized to predict the log-odds of the target variable, assuming a linear relationship between the independent variables and the log-odds of the outcome.

Random Forest

- **Description**: Random Forest is an ensemble learning model that operates by constructing multiple decision trees during training and aggregating their outputs to make predictions.
- **Mechanism**: It is a supervised learning algorithm that can be used for both classification and regression tasks, offering high accuracy, versatility, and robustness to overfitting, especially in large datasets.
- **Stability**: The model's performance is enhanced by randomness, as it selects random subsets of features for splitting at each tree node and builds trees on random samples of the data, which increases diversity among the trees and improves generalization.

K-Means Clustering

- **Description**: K-Means is an unsupervised learning algorithm used to partition data into a predefined number of clusters based on feature similarity.

- **Mechanism**: It minimizes variance within clusters by iteratively assigning data points to the nearest cluster center, then updating the center based on the mean of assigned points.
- **Stability**: The algorithm continues until cluster assignments stabilize, making it effective for finding natural groupings in data without prior labels.

Comparative Analysis and Recommendations

When comparing the 3 models, all of the models were able to output verifiable results based on the dataset, with logistic regression being better than K Means in terms of all conditions. The supervised learning models(Logistic Regression and Random Forest) proved to be a lot more fruitful than the unsupervised model(K-means clustering). Logistic regression performed well due to the simplicity and effectiveness, while Random Forest, which was slightly less accurate, proved to be robust with datasets of non-linear relationships. The high values associated with accuracy, precision, recall, and F1-score suggest that the model is well suited for predicting the winner of an NFL game based on the feature data provided. Additionally, the Random Forest model also demonstrated exceptional performance, achieving strong metrics across all evaluation categories, which highlights its robustness and effectiveness as a predictive tool in this context. Its ensemble approach makes it a reliable choice for handling complex datasets and improving prediction accuracy. K-means on the other hand does not predict the outcomes but rather it adds value by grouping all of the NFL games based on feature similarity, and shows that certain game outcomes are associated with certain thresholds of turnovers or possession times. Overall, the 3 models worked together to provide a very descriptive analysis of our NFL games.

Why did your model perform well?

- **Relevant Features**: The features used (passing yards, rushing yards, touchdowns, sacks, etc.) are good indicators of NFL teams' success, allowing the models to make accurate predictions.
- **Appropriate Model Choice**: Both models are resilient classifiers that effectively manage the variability and complexity within the dataset.
- **Overfitting**: Both models were able to handle overfitting by using regularization techniques (such as Lasso and Ridge regression) to prevent overfitting.
- Preprocessing: Effective data preprocessing(handling missing values and feature engineering) made sure that the models were trained on clean and reliable data.

Next Steps

- **Hyperparameter Tuning**: Testing and tuning the model's hyperparameters would optimize performance, ensuring an ideal balance of accuracy, recall, and precision.
- **Feature Engineering**: Discover and incorporate different features to enhance model accuracy and precision, such as incorporating more niche metrics into our calculations.
- **Explore Additional Models**: We could test out other types of models and compare and contrast with our model to combine results and see how we could optimize our results and identify specific methods to again improve our outcomes.
- **Data Augmentation: Incorporate data (such as FanPlay app team at Georgia Tech) from more recent seasons to improve model generality and account for changes in team performances.

References

[1] M. Gifford and Tuncay Bayrak, “A predictive analytics model for forecasting outcomes in the National Football League games using decision tree and logistic regression,” *Decision Analytics Journal*, vol. 8, pp. 100296–100296, Aug. 2023. Available: https://doi.org/10.1016/j.dajour.2023.100296.

[2] J. Roith et al., “Predicting NFL football games based on simulation and modeling,” *International Journal of Statistics and Applied Mathematics*, vol. 3, no. 2, pp. 101–106, 2018. Available: https://www.mathsjournal.com/pdf/2018/vol3issue2/PartB/3-1-45-589.pdf.

[3] S. Bouzianis, “Predicting the Outcome of NFL Games Using Logistic Regression.” Available: https://scholars.unh.edu/cgi/viewcontent.cgi?article=1472&context=honors.

Team Contributions

	Name	Proposal Contribution
0	Ishaan	Analysis of quantitative metrics, analysis of algorithm, Comparative analysis
1	Rahul	Website, Final Presentation Video, Gantt Chart, Comparative Analysis
2	Vishnu	Final Presentation Slides, Model Implementation
3	Rohan	Website, Model Implementation
4	Sattwik	Final Presentation Slides, Model Implementation, Results

GANTT CHART

PROJECT TITLE	NFL Predictor (Rahul, Vishnu, Ishaan, Sattwik, Rohan)

TASK TITLE	TASK OWNER	START DATE	DUE DATE	DURATION
Project Proposal				
Introduction & Background	Ishaan, Vishnu	10/2/24	10/4/24	2
Problem Definition	Ishaan	10/2/24	10/4/24	2
Methods	Rahul, Vishnu,Sattwik	10/2/24	10/4/24	2
Potential Dataset	Vishnu, Rohan	10/3/24	10/4/24	1
Potential Results & Discussion	Sattwik, Rohan	10/3/24	10/4/24	1
Video Creation & Recording	Rahul , Rohan	10/3/24	10/4/24	1
Presentation	Sattwik, Ishaan	10/2/24	10/4/24	2
Streamlit Page	Rahul	9/27/24	10/4/24	7
Midterm Report				
Model 1 (M1) Design & Selection	Rahul, Vishnu	10/5/24	10/13/24	8
M1 Data Cleaning	Rohan, Rahul, Vishnu	10/7/24	10/15/24	8
M1 Data Visualization	Ishaan, Sattwik, Rohan	10/15/24	10/18/24	3
M1 Feature Reduction	Vishnu, Ishaan	10/18/24	10/25/24	7
M1 Implementation & Coding	Rohan, Ishaan	10/25/24	10/31/24	6
M1 Results Evaluation	Rahul, Sattwik	11/1/24	11/4/24	3
Model 2 (M2) Design & Selection	Rahul, Vishnu	10/5/24	10/13/24	8
M2 Data Cleaning	Rohan, Rahul, Vishnu	10/7/24	10/15/24	8
M2 Data Visualization	Ishaan, Sattwik, Rohan	10/15/24	10/18/24	3
M2 Feature Reduction	Rohan, Rahul	10/18/24	10/25/24	7
M2 Coding & Implementation	rohan, sattwik	10/25/24	10/31/24	6
M2 Results Evaluation	Ishaan, Rohan, Vishnu	11/1/24	11/4/24	3
Midterm Report	ishaan, vishnu, Rahul, Rohan	11/4/24	11/11/24	7
Final Report				
Model 3 (M3) Design & Selection	Rohan,Ishaan	11/12/24	11/16/24	4
M3 Data Cleaning	Rahul, Vishnu, Sattwik	11/17/24	11/20/24	3
M3 Data Visualization	Rohan, Ishaan, Vishnu	11/21/24	11/26/24	5
M3 Feature Reduction	Rahul, Sattwik	11/26/24	11/27/24	1
M3 Implementation & Coding	Vishnu, Ishaan	11/27/24	11/29/24	2
M3 Results Evaluation	Ishaan, Vishnu	11/30/24	12/1/24	1
M1-M3 Comparison	Rahul, Sattwik	12/1/24	12/3/24	2
Video Creation & Recording	Vishnu, Rahul, Sattwik	12/2/24	12/3/24	1
Final Report	Rahul, Vishnu, Rohan, Ishaan, Sattwik	11/28/24	12/3/24	5

Current Gantt Chart for Team