Fantasy Football Classification

Joseph File, Jayden Jinks, Darin Hardie, Caleb Anderson

Overview

- Fantasy Football Background
- Our Project
- Dataset
- Data Cleaning & Preprocessing
- Decision Tree Model
- Naïve Bayes Model
- Gradient Boosting Model
- Application
- Result Analysis
- Conclusion

Fantasy Football Background

- Fantasy football is worth over $11 billion and over 29 million Americans play fantasy football yearly, meaning the ability to accurately predict fantasy football outcomes is both a practical and valuable skill [1]
- Lineup in a standard Fantasy Football League:
  - 1 QB, 2 RB, 2 WR, 1 TE, 1 FLEX (RB/WR/TE), 1 D/ST, 1 K
  - RB and WR usually make up 5/8 lineup spots, making them the most valuable
- Our study considers a point-per-reception (PPR) scoring format:
  - 1 point per reception,
  - 0.1 point per receiving/rushing yard
  - 6 points for receiving/rushing TD

Our Project

- We will use 20 years of WR and RB fantasy football data to train and test 3 different types of classification models:
  - Decision Tree
  - Naïve Bayes Classifier
  - Gradient Boosting Classifier
- We will classify each position into tiers which are determined based on their fantasy points per game:
  - A standard fantasy league has 12 teams
  - Rank 1-12 : WR1/RB1
  - Rank 13-24: WR2/WR3
  - Rank 25-36: FLEX
  - Rank 37-60: Bench
  - Rank 61+: Cut
- If we are able to accurately classify players into these tiers based on historical data we will be able to identify trade targets by applying current data to the trained models

Dataset Sample

		RECEIVING							RUSHING			MISC				
RANK	PLAYER	REC	TGT	YDS	Y/R	LG	20+	TD	ATT	YDS	TD	FL	G	FPTS	FPTS/G	ROST
1	CeeDee Lamb (DAL)	135	181	1,749	13.0	92	29	12	14	113	2	2	17	403.2	23.7	99.8%
2	Tyreek Hill (MIA)	119	171	1,799	15.1	78	29	13	6	15	0	1	16	376.4	23.5	99.6%
3	Keenan Allen (CHI)	108	150	1,243	11.5	42	19	7	2	6	0	1	13	278.9	21.5	71.7%

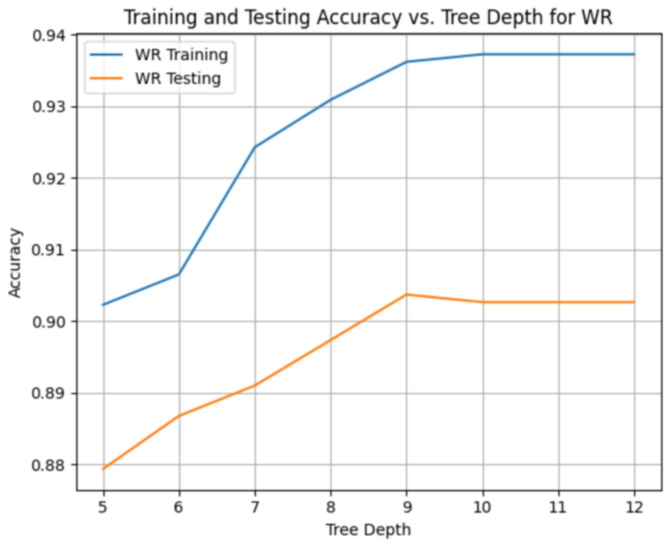
		RUSHING						RECEIVING					MISC				
RANK	PLAYER	ATT	YDS	Y/A	LG	20+	TD	REC	TGT	YDS	Y/R	TD	FL	G	FPTS	FPTS/G	ROST
1	Christian McCaffrey (SF)	272	1,459	5.4	72	9	14	67	83	564	8.4	7	2	16	391.3	24.5	99.6%
2	Kyren Williams (LAR)	228	1,144	5.0	56	7	12	32	48	206	6.4	3	2	12	255.0	21.3	99.6%
3	Alvin Kamara (NO)	180	694	3.9	17	0	5	75	86	466	6.2	1	0	13	233.0	17.9	99.3%

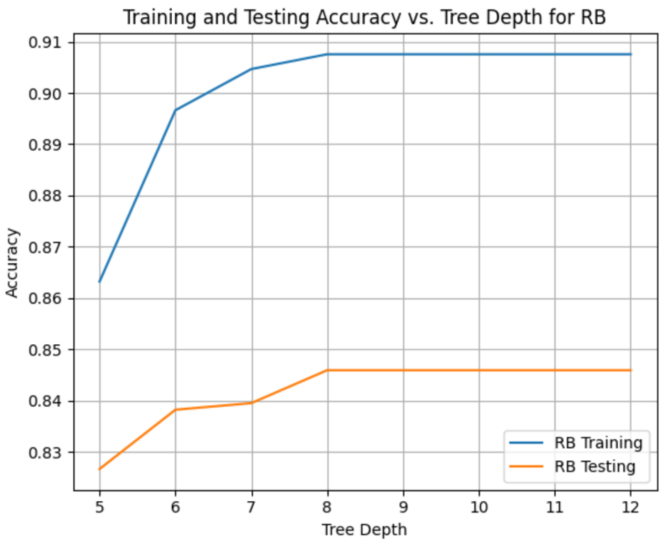
Data Preprocessing

- Combine all WR data 2002-2023 into a data frame and RB data into another.
- Convert Relevant columns to per-game statistics:
  - Receptions, Targets, Receiving Yards, Receiving TDs, Rush Attempts, Rushing Yards, Rushing TDs
- Convert the rank column into tiers according to the previously discussed guidelines
- Ignore data irrelevant to classification:
  - Games played, Roster percentage, Total Fantasy points
- Perform 80/20 train test split on both WR and RB data

Decision Tree Classifier

- Used entropy to determine best splitting point amongst features
- Included parameter for controlling the maximum depth of the tree
- Created trees with depths 5-12 for both RB and WR to determine which depth optimized training and testing performance:





Decision Tree Performance

	RB (Depth 8)	WR (Depth 9)
F1 Score	0.8392	0.9045
Testing Accuracy	0.8434	0.9058

Naïve Bayes Classifier

- Naïve Bayes Classifiers operate under the assumption that all features are independent of each other.
- Our team chose to implement a Gaussian Naïve Bayes model which uses the following function to calculate the probability that a given feature x_i belongs to a class *c:*

$$P(x_i|C = c) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp(-\frac{(x_i - \mu)^2}{2\sigma^2})$$

- The model then calculates the using the prior probabilities (P(C=c)) and the calculated likelihoods (P(x|C=c)) to calculate the posterior for a given sample x and class c using the following equation:

$$P(C = c|x) = P(C = c) * P(x|C = c)$$

- The model predicts for whichever class c has the greatest posterior

Naïve Bayes Classifier Performance

	RB	WR
F1 Score	0.8085	0.8464
Testing Accuracy	0.7959	0.8381

Gradient Boosting Model

- A gradient boosting model combines several weaker learning models to make a stronger learning model by minimizing the loss function using gradient descent [5].
  - In this case the models being combined are Decision Tree Regressors from sklearn
- First, each class is given a probability equal to 1/ number of classes, and the gradient is computed as:

$$Gradient = y - P$$

  - where y is the class labels encoded numerically, and P is the probability distribution of the sample.

Gradient Boosting Model

- A tree is trained for each tier class, then these trees are used to update the prediction values by adding the product of the learning rate at the prediction value for the class based on the current tree for that class.
- This is repeated with the prediction scores and gradients updating until a set number of iterations is met
- Then the prediction scores (F) are converted into probabilities using the following function, and the highest probability is the class predicted:

$$P_{i,c} = \frac{\exp(F_{i,c})}{\sum_j \exp(F_{i,j})}$$

Gradient Boosting Model Performance

	RB	WR
F1 Score	0.8482	0.8982
Testing Accuracy	0.8537	0.9037

Application

- In most Fantasy Football leagues, teams can trade amongst each other
- Using our classification models, we are able to identify trade targets by finding players who are classified in tiers higher than the tier their current rank places them in
- This indicates that historical statistical trends think their performance will improve in the remainder of the season based on their statistics thus far
- We took WR and RB data from week 1-6 of the 2024 NFL Season and asked our models, trained on the 2002-2023 data to classify these players, printing the WR1, WR2, RB1, and RB2, players alongside their actual tier through 6 weeks

Decision Tree Trade Targets

Name	Classifier Tier	Week 6 Tier	Week 12 Tier
Garrett Wilson	WR1	WR2	
Rashee Rice	WR1	WR2	
Tee Higgins	WR1	WR2	
Josh Downs	WR1	WR2	
CeeDee Lamb	WR1	WR2	
Terry McLaurin	WR1	WR2	
DJ Moore	WR2	FLEX	
Wan'Dale Robinson	WR2	FLEX	
J.K. Dobbins	RB1	RB2	
Breece Hall	RB1	RB2	
Brian Robinson Jr.	RB1	RB2	
Tony Pollard	RB1	RB2	
Rhamondre Stevenson	RB1	RB2	
Zach Charbonnet	RB2	FLEX	
Josh Jacobs	RB2	FLEX	
Chase Brown	RB2	FLEX	
Najee Harris	RB2	FLEX	

Decision Tree Trade Targets

	WR	RB
Success Rate	50%	33%
Fail Rate	12.5%	22.2%

Name	Classifier Tier	Week 6 Tier	Week 12 Tier
Garrett Wilson	WR1	WR2	WR1
Rashee Rice	WR1	WR2	WR1
Tee Higgins	WR1	WR2	WR1
Josh Downs	WR1	WR2	WR2
CeeDee Lamb	WR1	WR2	WR1
Terry McLaurin	WR1	WR2	WR2
DJ Moore	WR2	FLEX	FLEX
Wan'Dale Robinson	WR2	FLEX	Bench
J.K. Dobbins	RB1	RB2	RB2
Breece Hall	RB1	RB2	RB1
Brian Robinson Jr.	RB1	RB2	FLEX
Tony Pollard	RB1	RB2	RB2
Rhamondre Stevenson	RB1	RB2	FLEX
Zach Charbonnet	RB2	FLEX	FLEX
Josh Jacobs	RB2	FLEX	RB1
Chase Brown	RB2	FLEX	RB2
Najee Harris	RB2	FLEX	FLEX

Naïve Bayes Classifier Trade Targets

Name	Classifier Tier	Week 6 Tier	Week 12 Tier
Allen Lazard	WR1	WR2	
Mike Evans	WR1	WR2	
Deebo Samuel	WR1	FLEX	
Kristian Wilkerson	WR1	Bench	
Brian Robinson Jr.	RB1	RB2	

Naïve Bayes Classifier Trade Targets

	WR	RB
Success Rate	0%	0%
Fail Rate	75%	100%

Name	Classifier Tier	Week 6 Tier	Week 12 Tier
Allen Lazard	WR1	WR2	WR2
Mike Evans	WR1	WR2	FLEX
Deebo Samuel	WR1	FLEX	Bench
Kristian Wilkerson	WR1	Bench	Cut
Brian Robinson Jr.	RB1	RB2	FLEX

Gradient Boosting Model Trade Targets

Name	Classifier Tier	Week 6 Tier	Week 12 Tier
Garrett Wilson	WR1	WR2	
Rashee Rice	WR1	WR2	
Tee Higgins	WR1	WR2	
Josh Downs	WR1	WR2	
Terry McLaurin	WR1	WR2	
Jauan Jennings	WR2	FLEX	
Breece Hall	RB1	RB2	
Zach Charbonnet	RB2	FLEX	
Najee Harris	RB2	FLEX	

Gradient Boosting Model Trade Targets

	WR	RB
Success Rate	75%	33%
Fail Rate	0%	0%

Name	Classifier Tier	Week 6 Tier	Week 12 Tier
Garrett Wilson	WR1	WR2	WR1
Rashee Rice	WR1	WR2	WR1
Tee Higgins	WR1	WR2	WR1
Josh Downs	WR1	WR2	WR2
Terry McLaurin	WR1	WR2	WR2
Jauan Jennings	WR2	FLEX	WR2
Breece Hall	RB1	RB2	RB1
Zach Charbonnet	RB2	FLEX	FLEX
Najee Harris	RB2	FLEX	FLEX

Result Analysis

- Decision Tree and Gradient Boosting Model consistently outperformed Naïve Bayes
- Naïve Bayes may not be very suited to this specific classification task for various reasons:
  - Skewed Data, Naïve Bayes is made to follow a specific distribution of data
  - Relationships between features, handled well by Decision Tree and Gradient Boosting model, not Naïve Bayes
- The models also always performed better when classifying WRs than RBs
  - Possibly because in a PPR scoring format, WR stats translate to points a lot more directly than RB stats
    - A WR with a lot of targets & receptions is guaranteed to be receiving points, while a RB with a lot of attempts may not be accruing yards, making the position harder to classify based on statistics alone

Conclusion

- While it’s not perfect, a classification model like a decision tree or gradient boosting model trained on historical fantasy football data can certainly be used to identify potential trade targets
  - When paired with an understanding of injuries, matchups, etc. these tools would certainly be beneficial
- This research could be expanded upon in several ways:
  - Using a dataset with additional more in-depth statistics may allow for better classification
  - Using ranking algorithms to determine specific ranks rather than tiers could provide even more valuable insight
  - There are many other classification algorithms that could be considered, some may be more effective than those we tested

References

[1] S. Gardner, “Money. Power. Women. The driving forces behind fantasy football’s skyrocketing popularity.,” *USA TODAY*, Dec. 15, 2023. https://www.usatoday.com/story/sports/nfl/fantasy/2023/12/15/fantasy-football-sports-economy/71870731007/

[2] “2023 NFL WR Statistics | Fantasy Football PPR | FantasyPros,” *www.fantasypros.com*. https://www.fantasypros.com/nfl/stats/wr.php?scoring=PPR

[3] GeeksForGeeks, “Decision tree - geeksforgeeks,” *GeeksforGeeks*, May 17, 2024. https://www.geeksforgeeks.org/decision-tree/

[4] GeeksForGeeks, “Naive Bayes Classifiers,” *GeeksforGeeks*, Mar. 03, 2017. https://www.geeksforgeeks.org/naive-bayes-classifiers/#

[5] GeeksForGeeks, “ML - Gradient Boosting,” *GeeksforGeeks*, Aug. 25, 2020. https://www.geeksforgeeks.org/ml-gradient-boosting/

Questions?