Predicting a Player's NFL Draft Round

Springboard Capstone #1 Project

Pawandeep Jandir

Springboard

Overview

- Introduction
  - Background and problem statement
- Data Wrangling
  - Data sources used, cleaning performed and full list of variables
- Data Analysis
  - Exploring the data with closer look at Quarterbacks
- Data Modeling and Predicting
  - Preprocessing data, description of algorithms, and results of the models
- Conclusion
  - Further work

Introduction

- The National Football League (NFL) conducts an annual seven-round Draft to select college players
  - Only 256 can be drafted leaving many undrafted
  - Lots of resources are spent both by NFL teams to select the best players and by players to be as coveted as possible
- The NFL Scouting Combine (Combine) is an event where incoming player perform physical and mental tasks
  - E.g. forty yard dash, vertical jump, and Wonderlic test (intelligence test)
  - Part of the pre-draft scouting process
- This analysis uses the Combine to predict which round a player is drafted
  - An eight-label classification problem: seven draft rounds + undrafted
  - Python is used end-to-end

Data Sources

- NFL Savant: This site has Combine data from 1999 to 2015
  - Data available as a single CSV file

- Sports Reference: This site has Combine data (in addition to much more) as well and was used to check for completeness and augment the NFL Savant data
  - Data scraped and cleaned from multiple HTML tables

- Draft History: This site was used to check completeness of the actual drafted players and augment the NFL Savant data
  - Data scraped and cleaned from multiple HTML tables

Data Cleaning

- The original NFL Savant has missing values throughout the dataset
  - Including both Combine and Draft results
    - The Sports Reference data and Draft History were used to augment both sets of values, respectively
- Certain columns, like player college, needed to be cleaned after merging
  - The same college listed by different names (e.g. USC vs. Southern Cal)
- Some null data still remained after merging
  - Filled with median imputation grouped per positiongroup (see table in next slide) in each column

List of variables

- The fully cleaned dataset contains 21 columns, including the target variable (draft round)
- The table below lists these variables with a short description

Feature	Description	Notes	Useful in prediction?
name	Name of the player	-	No
year	Draft year	-	Possibly
college	Player's college attended	-	Possibly
position	Player's college position	Scraped value	Yes
positiongroup	Player's college position group	Assigned value	Yes
height*	Player's height (inches)	-	Yes
weight*	Player's weight (pounds)	-	Yes
fortyyd*	Player's 40-yard dash (seconds)	Full description	Yes
vertical*	Player's standing vertical jump (inches)	Full description	Yes
bench*	Player's bench press of 225 pounds (reps)	Full description	Yes
threecone*	Player's 3 cone drill (seconds)	Full description	Yes
shuttle*	Player's shuttle run (seconds)	Full description	Yes
broad*	Player's standing long jump (inches)	Full description	Yes
wonderlic*	Player's Wonderlic test result	Sparsely populated	Possibly
nflgrade*	Player's NFL Grade determined by experts	Sparsely populated	Possibly
arms*	Player's arm length (inches)	-	Yes
hands*	Player's hand length (inches)	-	Yes
team	Team that drafts player	-	Possibly
round	Player's draft round	Target variable	N/A
pick	Player's draft pick number within a round	-	No
overall	Player's overall draft number	-	No

Exploring Quarterbacks

- Exploratory analysis of the data was particularly focused on Quarterbacks (QB), likely the most important position in football
- Graph below show the number of QBs in the dataset as a function of (draft) year
  - This number has a bit of variance over the years, averaging about 20 per year

Exploring Quarterbacks II

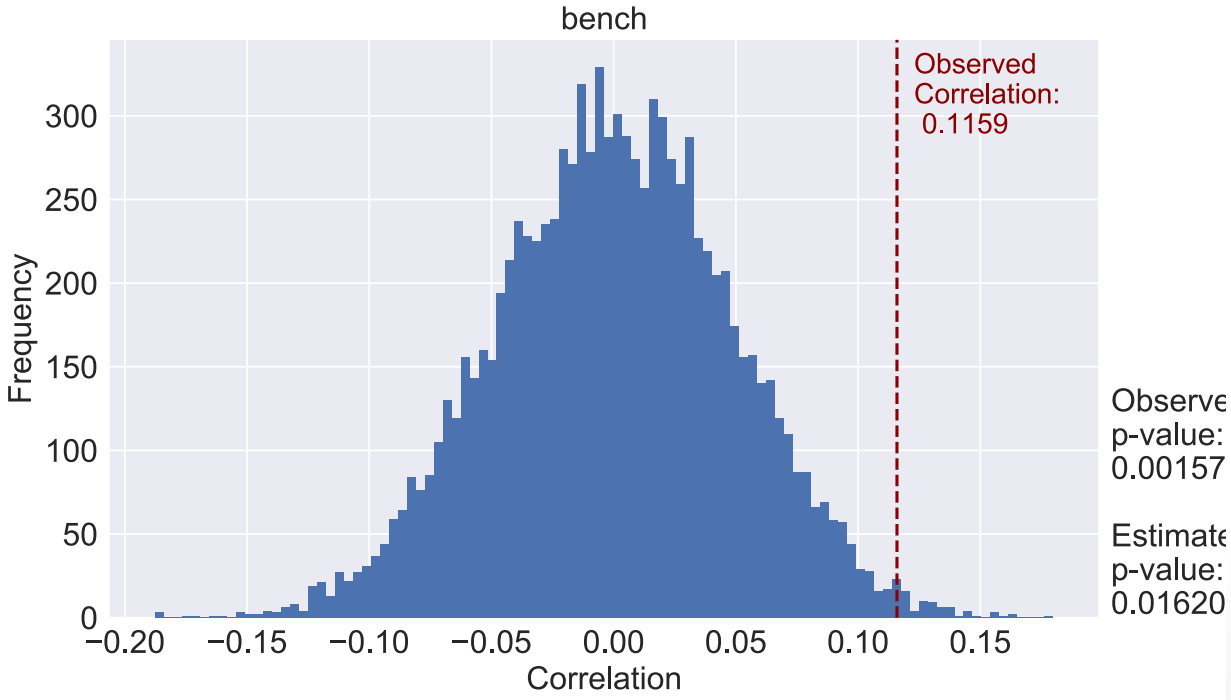
- We can also look at the distribution of QBs as a function of draft round.
  - Undrafted players are given a draft round of -1, to make it very obvious

Draft Round	-1	1	2	3	4	5	6	7
Count	141	47	19	22	23	29	27	26

- Two takeaways
  - Clear a majority of QBs are not drafted
  - The first round (i.e. highly regarded and coveted) is the most popular round to draft a QB

Exploring Quarterbacks III

- Can calculate the (Kendall) correlation between draft round and a feature (for QBs only) and its p-value for significance
  - Use permutation test: the draft round label is randomly exchanged and the correlation re-calculated. Process repeated 10,000 times
  - Calculate p-value from number of trials larger than original correlation
  - Example for bench features shown below



Preprocessing Data

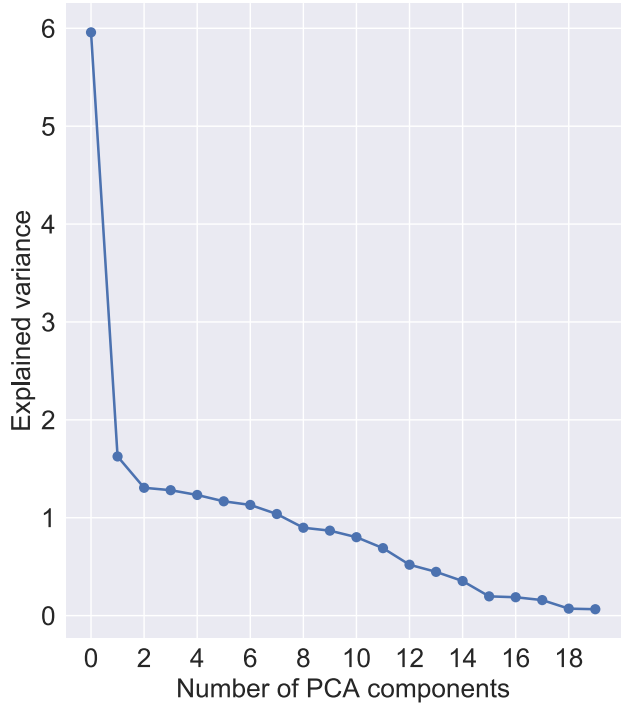
- Before building prediction models, need to preprocess and normalize data
  - Final list of 20 features used in model building is shown in table below

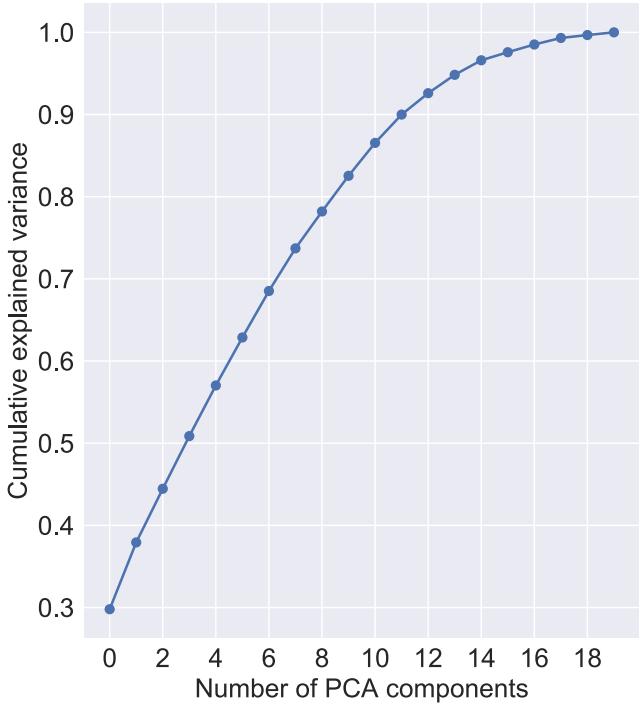
- Standardization or normalization of data needed since widely different ranges exist in different features
  - Year ranges from 1999-2015
  - Fortyyd ranges from 4.2 to 6.1
  - Different normalization methods are available
    - Scaling to max value of feature
      - Range from 0 to 1
    - Center and scale variance to unity

Feature	Type
year	Integer
height	Integer
weight	Float
fortyyd	Float
vertical	Float
bench	Float
threecone	Float
shuttle	Float
broad	Float
wonderlic	Integer
nflgrade	Float
arms	Float
hands	Float
positiongroup_DB	Dummy (binary)
positiongroup_DL	Dummy (binary)
positiongroup_LB	Dummy (binary)
positiongroup_OL	Dummy (binary)
positiongroup_QB	Dummy (binary)
positiongroup_RE	Dummy (binary)
positiongroup_ST	Dummy (binary)

Dimensionality Reduction

- Can apply Principal Component Analysis (PCA) on features to determine if reduction of dimensions might help
  - Explained variances are plotted below
    - No clear indication there is a better dimensional space for features
  - Tested on Logistic Regression
    - Did not improve the model





Cross-Validation and Scoring

- Weighted F1-score is used to score models
  - $F1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$
  - Both precision and recall are calculated per class and then averaged with weights
    - Weights are equivalent to the class ratios
      - Shown in table to the right

Round	Proportion [%]
-1	36.3
1	9.26
2	9.26
3	9.58
4	10.2
5	9.03
6	8.03
7	8.32

- An 80/20 training-testing data split is made
  - Training data is used alongside a Cross-Validated Grid Search to find the optimal parameters for specific algorithms
    - Stratified 5-fold strategy
  - Testing data is used to test and compare models
    - Cross-validation scoring is used to come up with final metric for each optimal model

Baseline score

- Before building complex models need a baseline score
  - Models will need to best this score to be worthwhile

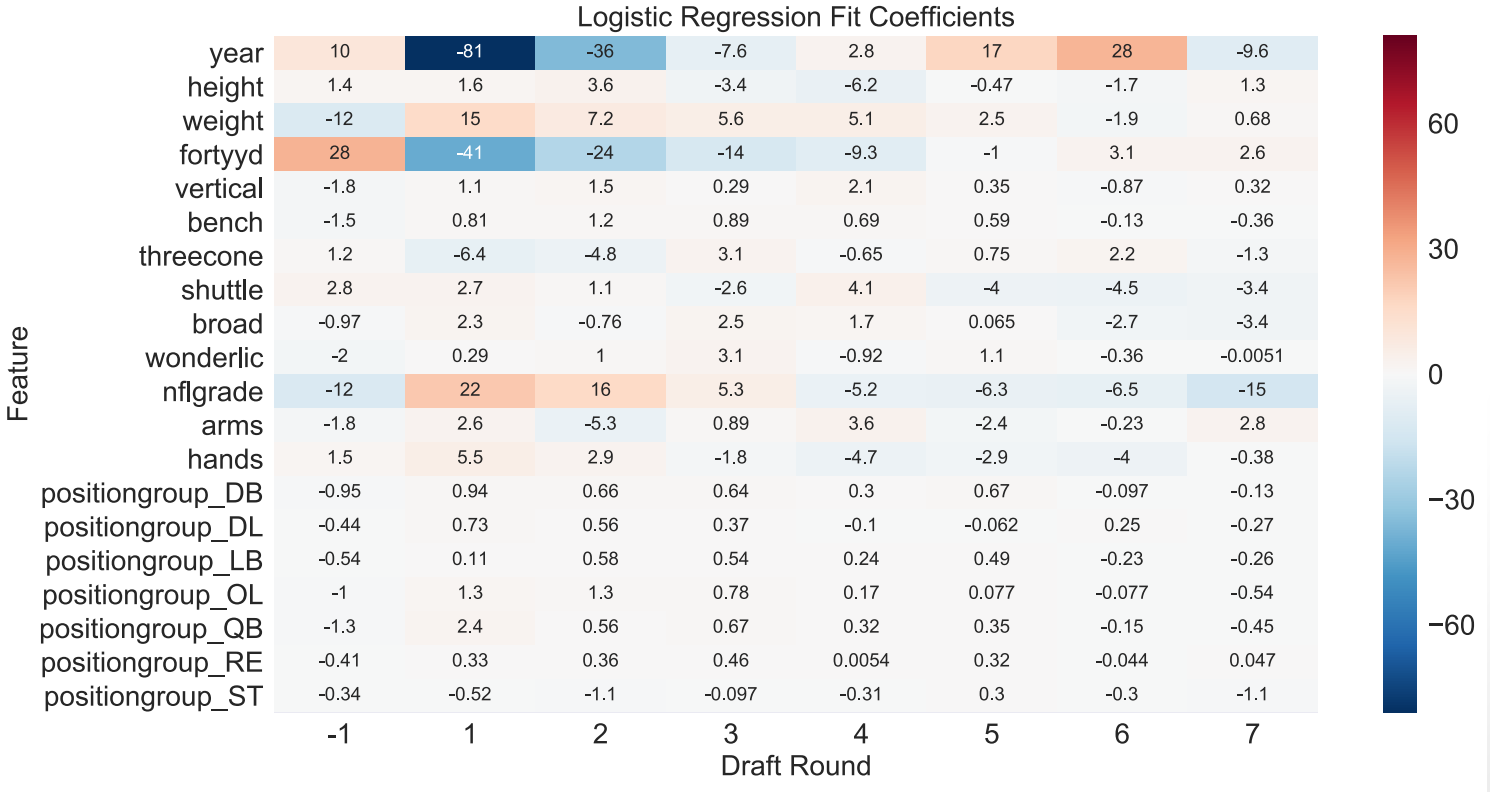
- Because classes are unbalanced, a simple, dummy classifier can guess all players to go undrafted (the largest class)
  - This results in a F1 score of 19%
  - Raw accuracy is, of course, 36% (see table on previous slide)

Algorithms

- Six algorithms were tested to predict the draft round
  - Three non-ensemble methods
    - Logistic Regression (LR), k-Nearest Neighbors (k-NN), and Support Vector Machine (SVM)
  - Three ensemble methods
    - Random Forest (RF), Extremely Randomized Trees (ERT), and a stacked ensemble method
      - Stacked ensemble method discussed in detail later
- The k-Nearest Neighbors and Support Vector Machine are not discussed in these slides, for brevity

Logistic Regression

- One of the simplest algorithms
  - Benefit is interpretation: there are fit coefficients for all features for each class
    - Easy to visualize how the model fit the data
    - Heatmap below shows this result

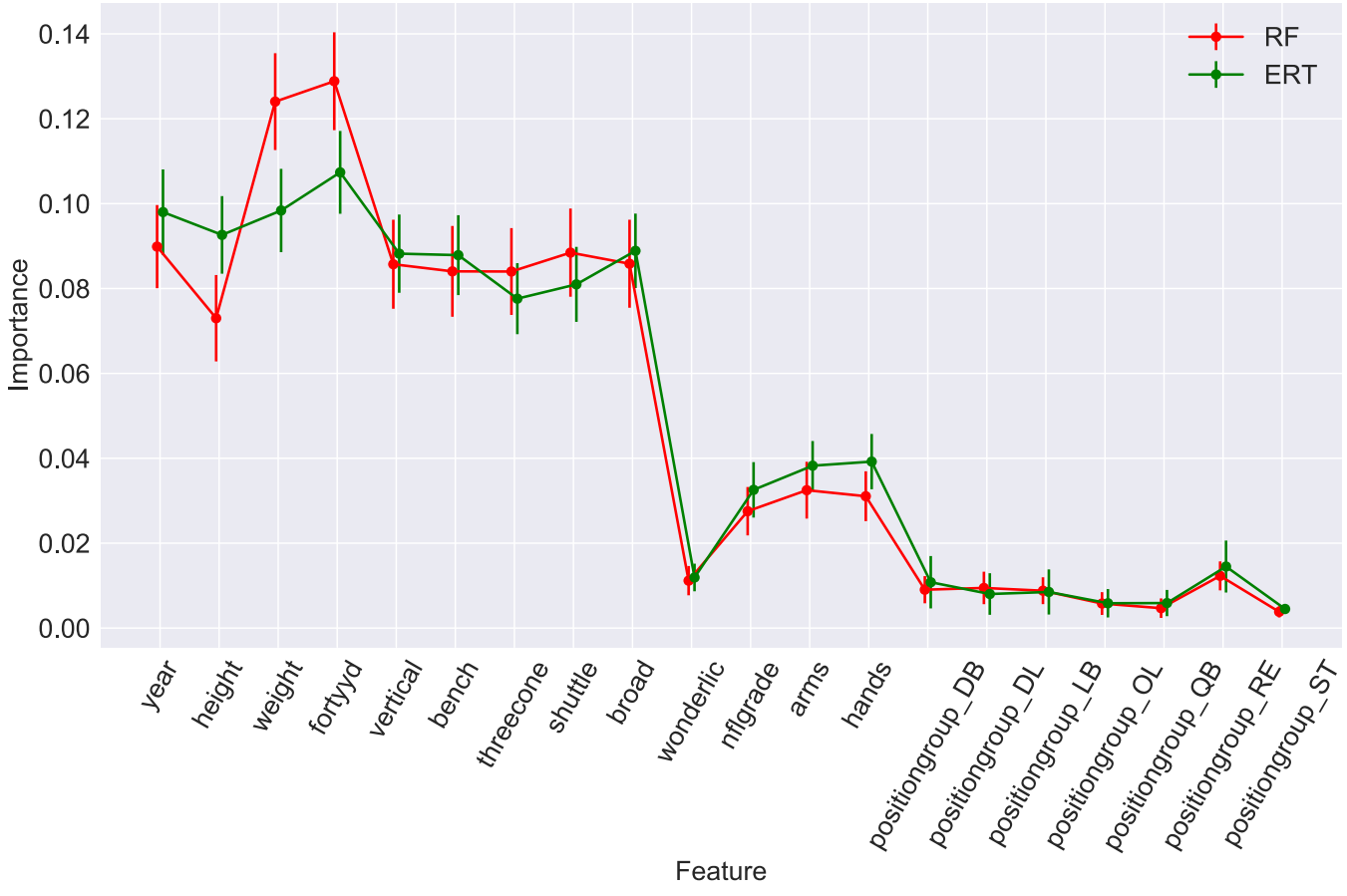


Tree-based methods

- RF grows multiple decision trees (DT) using a random subset of all features.
  - ERT takes the randomness further by also randomizing the splitting
- Complex model with many parameters
  - Number of trees grown, maximum depth of tree, minimum number of samples to split a node, number of features used in a tree, minimum number of samples needed at a leaf node, among others
  - Can be difficult to interpret
- Decreases variance and over-fitting of the model from the randomness
- Also generates ranking of features
  - See next slide

Feature Importance

- Feature importance (with errors) for RF and ERT
  - Similar results between the two methods
  - Compatible with fit coefficients from Logistic Regression model



Stacked Ensemble

- Majority rules classifier built from the other models
  - Each classifier in the ensemble is given a vote to predict the outcome, with a majority rule deciding the outcome
  - Predicted probabilities, not the predicted class, is used for each classifier's vote
  - Classifiers are also given weights according to a simple brute-force grid search
    - Non-equal voting say in the final outcome
    - Random Forest generally given more preference for final decisions
- Balances weakness in classifiers while keeping strengths
- Only Logistic Regression, Support Vector Machine, and Random Forest models are used

Model Results

- All models beat the baseline score
  - Good indication, there is predictive power in the data
- Stacked ensemble is best, by small margin
  - This is final model

Classifier	F1 score
Dummy (Baseline)	0.1936
Logistic Regression	0.2416
k-Nearest Neighbors	0.2479
Support Vector Machine	0.2309
Random Forest	0.2907
Extremely Randomized Trees	0.2797
Stacked Ensemble	0.2911

Conclusion

- Made a model to predict which round a college player would be drafted into the NFL using NFL Combine data
  - Variety of data sources used to
- A stacked majority rules ensemble classifier is used
  - Built from a Logistic Regression, Support Vector Machine, and Random Forest
  - Outperforms baseline F1 score: 29% to 19%
- Avenues for improvement
  - Gather additional years of data
  - Try additional algorithms (Boosted Trees, Neural Networks, etc.)
  - Add additional features
    - Info about a player's college, player's stats in college, player's honors/awards received in college, etc.

Thank You

https://xkcd.com/1838/