# Final Presentation

Zach Brazil, Richard Charles, and Adam Kiehl

12/8/21

## Introduction

Can we predict top fantasy point performers in the NFL using publicly available advanced metrics?

- ▶ Training set: 2018-2020 seasons; Testing set: 2021 season
- ▶ Data scraped from pro-football-reference.com
- ▶ Tried range of models to predict point-worthy statistics:

| Stat | Pts |
|---|---|
| PassYds | 0.04 |
| PassTD | 4.00 |
| PassInt | -1.00 |
| RushYds | 0.10 |
| RushTD | 6.00 |
| Rec | 1.00 |
| RecYds | 0.10 |
| RecTD | 6.00 |
| FL | -2.00 |

# Passing Analysis pt. 1

Models that were used:

- ▶ Multiple Linear Regression (MLR)
- ▶ LASSO
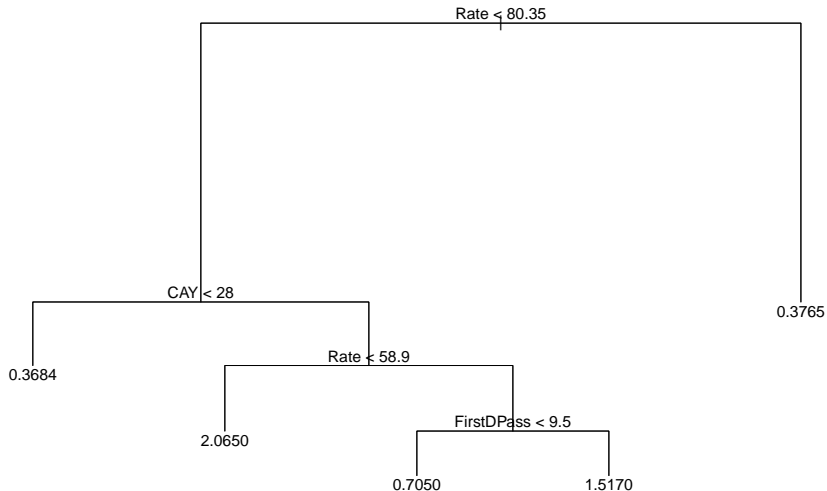- ▶ Principal Component Analysis/Regression
- ▶ Tree
- ▶ Bagging

Predictors: `Cmp`, `PassAtt`, `Sk`, `YdsLost`, `PassLng`, `Rate`, `FirstDPass`, `FirstDPassPer`, `CAY`, `YACPerCmp`, `PassDrops`, `BadThrow`, `BadPer`

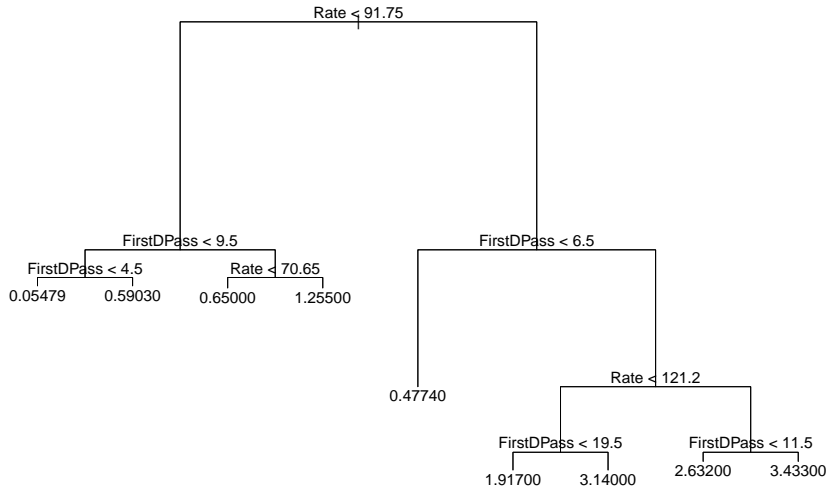Dropped `PassYAC` (multicollinearity)

# Tree (Passing Yards)
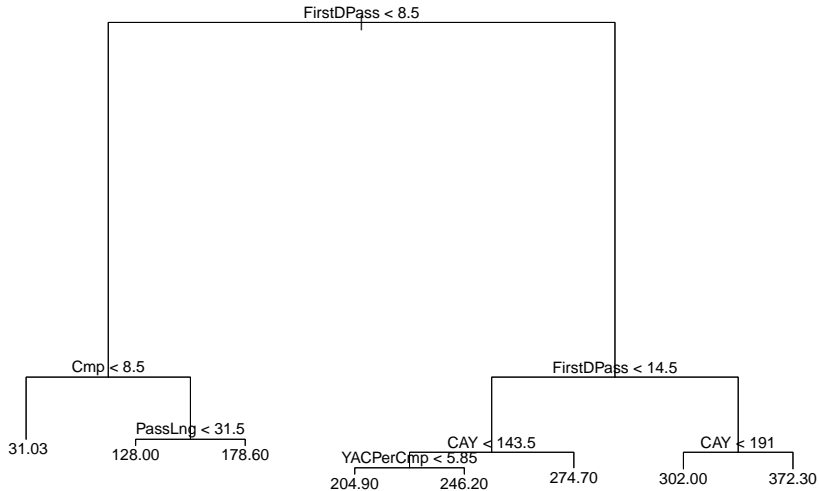
# Tree (Interceptions)

# Tree (Passing Touchdowns)

# Passing Analysis MSE Matrix

```
## # A tibble: 4 x 6
##   Model      MLR  LASSO    PCR   Tree Bagging
##   <chr>    <dbl>  <dbl>  <dbl>  <dbl>   <dbl>
## 1 PassYds 420.   429.   691.   1641.  168.
## 2 PassInt   0.513  0.515  0.656  0.51   0.305
## 3 PassTD    0.374  0.647  0.788  0.609  0.468
## 4 FL        0.191  0.211  0.193  0.201  0.21
```

# Receiving Analysis

Predictors: Tgt, RecLng, Fmb, FirstDRec, RecYBC, YBCPerR, RecYAC, YACPerR, ADOT, RecBrkTkl, RecPerBr, RecDrop, DropPerRec, RecInt, Rat

- ▶ Goal of simplicity and predictive accuracy
- ▶ Correlation matrix and VIF scores to identify multicollinearity (`FirstDRec`, `RecYBC`, `YBCPerR`, `YACPerR`, `DropPerRec`)
- ▶ `RecYAC` dropped

# Dimension Reduction

- Dimension reduction techniques:
  - Best Subset -> 3-5 predictors
  - LASSO -> 5-10 predictors
  - PC regression -> 7-8 principal components
  - Pruned trees -> 2 terminal nodes for `RecTD` and 6-7 otherwise
  - Bagging and Random Forest -> 1 important variable for `RecTD` and 4-5 otherwise
  - `Tgt` and `Rat` predictors appear often

# Receiving Results

| Model  | MLR    | Subset | LASSO  | PCR    | Tree    |
|--------|--------|--------|--------|--------|---------|
| Rec    | 0.513  | 0.532  | 0.587  | 0.607  | 1.178   |
| RecYds | 64.832 | 71.766 | 84.867 | 47.466 | 191.374 |
| RecTD  | 0.094  | 0.107  | 0.077  | 0.153  | 0.104   |

| Model  | Boost  | Bag    | RF     |
|--------|--------|--------|--------|
| Rec    | 0.398  | 0.154  | 0.135  |
| RecYds | 50.249 | 17.797 | 22.757 |
| RecTD  | 0.086  | 0.048  | 0.057  |

# Receptions Model

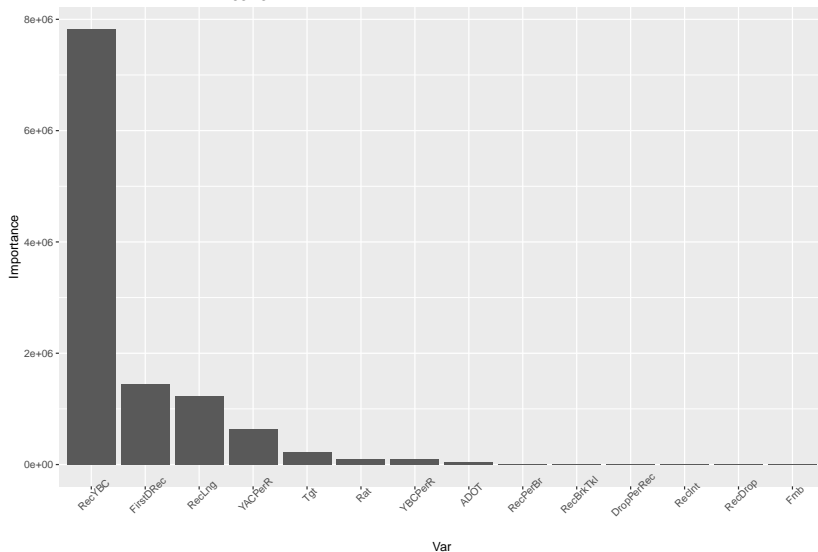Rec: Random forest model chosen with `Tgt`, `FirstDRec`, `RecYAC`,
`RecYBC`, `Rat`



Variable Selection with Random Forest
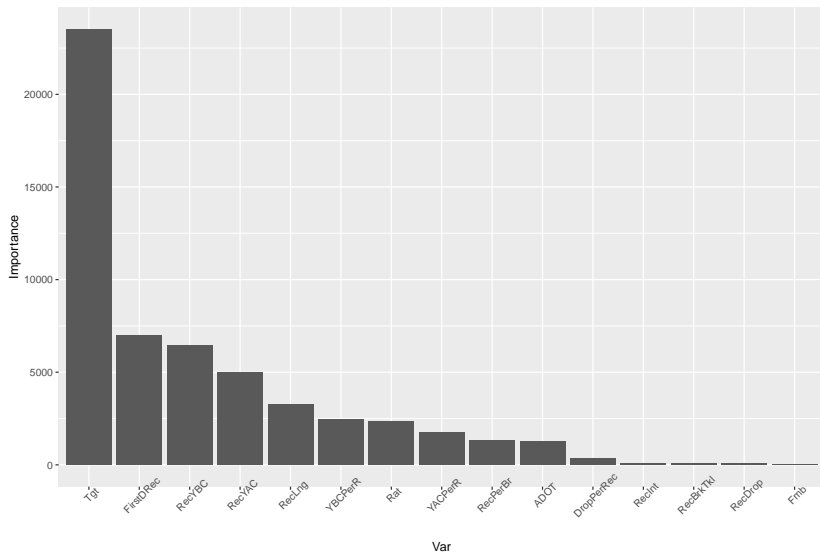
# Receiving Yards Model

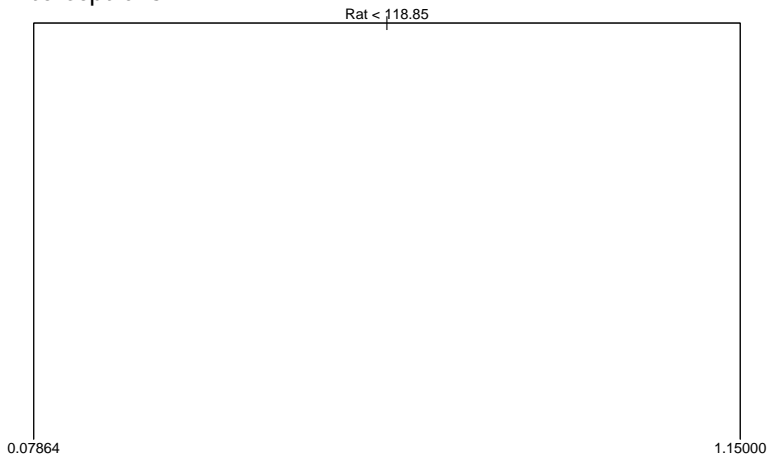RecYds: Bagged tree model chosen with `RecYBC`, `FirstDRec`, `RecLng`, `YACPerR`

# Receiving TDs Model

RecTD: Tree model chosen with only `Rat` as a predictor

▶ QBR uses frequencies of completions, yards, touchdowns, and interceptions
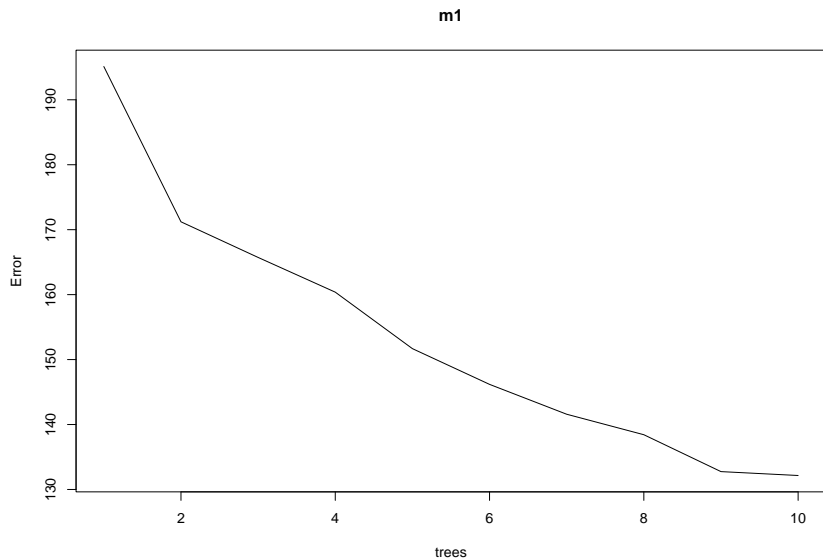
# Rushing Analysis pt. 1

Rushing Predictors: RushYds, RushTD, FL

- ▶ Started out with a number of predictors but determined that these 3 were the most important in this case.
- ▶ When analyzing rushing data, we found that there are a lot of variables that are colinear.
- ▶ One example of a predictor that I did not end up needing is 'FirstDRush'. This variable does not lead to fantasy points and relates closely with 'RushYds'.

# Rushing Analysis pt. 2

- Machine Learning Models:
  - The models I ended up using were Muliple Linear Regression, Lasso, PCR, Bagging, and Boosting
  - PCR ended up giving values that were not as useful for our rushing experimentation.
  - MLR provided the best MSE values for RushTD and FL (Fumbles lost)
  - Bagging gave the best value for RushYds
  - Thus these models were selected for final testing.

# Rushing Analysis pt. 3

# Fumbles Analysis

- ▶ Three different models chosen for `FL` from three different predictor sets:
  - ▶ MLR model using passing data: MSE=0.19
  - ▶ Tree model using receiving data: MSE=0.03
  - ▶ MLR model using rushing data: MSE=0.08
- ▶ Best overall was a 2-node tree model using `Fmb` as a predictor

# Validation

- ▶ Prediction performed on averaged data set of each player's last 17 games
- ▶ Resulted in an expected 'typical performance' for MSE calculations

| Stat | MSE |
|--------|---------|
| PassYds | 1647.63 |
| PassTD | 0.18 |
| PassInt | 0.98 |
| RushYds | 259.96 |
| RushTD | 0.12 |
| Rec | 3.84 |
| RecYds | 640.48 |
| RecTD | 0.20 |
| FL | 0.05 |
| fanPts | 50.29 |

# Results

|     | Player             | Pos | fanPts | MSE       |
| --- | ------------------ | --- | ------ | --------- |
| 223 | Lamar Jackson      | QB  | 30.00  | 103.62164 |
| 65  | Christian McCaffrey| RB  | 26.10  | 90.37000  |
| 222 | Kyler Murray       | QB  | 25.56  | 72.31625  |
| 282 | Patrick Mahomes    | QB  | 24.22  | 97.51247  |
| 307 | Russell Wilson     | QB  | 23.68  | 105.13337 |
| 192 | Josh Allen         | QB  | 22.50  | 70.22672  |
| 77  | Dak Prescott       | QB  | 22.46  | 71.10427  |
| 134 | Gardner Minshew II | QB  | 21.84  | 457.96000 |
| 4   | Aaron Rodgers      | QB  | 21.42  | 66.35480  |
| 107 | Derek Carr         | QB  | 21.42  | 45.94648  |
| 308 | Ryan Fitzpatrick   | QB  | 21.28  | 422.71360 |

# MSE



Distribution of Errors
Players used as unit for MSE calculation
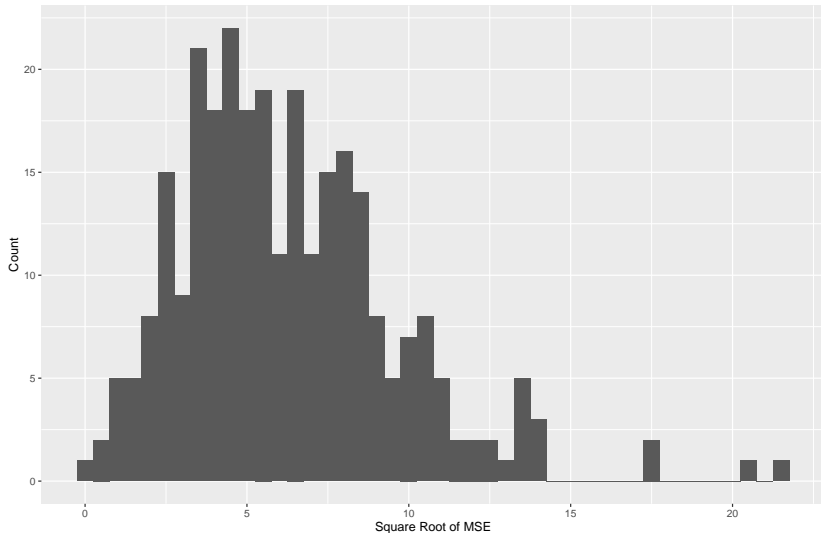
# Future Efforts

- Use bootstrap or MC methods to generate large sample size
- Assume a discrete (Poisson?) distribution for Rec, TD, Int, FL
- Other factors to consider:
  - Expectation of injury
  - Strength of opponent