NCAA MBB

MGSC 310 Final Project

By: Neel Shastri, Grant Sielman, Blake Khaleghi, Ryan King

Data Set

- College Basketball Dataset:
  - Kaggle
  - consists of data from years 2013 to 2021
    - we'll be looking at the data from only years 2019 to 2021
  - Dimensions:
    - 1053 rows, 23 columns(original dataset)
    - 1053 Rows, 11 Columns(cleaned dataset)
- Objective: Which metric has the biggest impact on winning games?

Data Cleaning

- Dropped variables related to March Madness
- Variables Added
  - Year
  - Team
  - Conf_Other
  - WINPER
- Renamed Variables
  - This was to better understand what each variable represents

```
CBB_variables <- CBB_data %>% select(W, ADJOE, ADJDE, EFG_O, EFG_D, TOR, TOR
M, YR)

glimpse(CBB_variables)
## Rows: 1,053
## Columns: 11
## $ W     <dbl> 25, 23, 21, 24, 18, 18, 13, 14, 19, 18, 10, 11, 12, 9,…
## $ ADJOE <dbl> 104.6, 101.3, 111.1, 107.3, 101.1, 103.5, 108.0, 105.3…
## $ ADJDE <dbl> 87.6, 95.7, 98.9, 99.8, 95.1, 99.9, 107.6, 105.3, 104.…
## $ EFG_O <dbl> 50.1, 46.7, 56.1, 53.1, 48.3, 47.4, 54.1, 48.0, 49.9, …
## $ EFG_D <dbl> 43.5, 46.9, 47.9, 48.3, 47.2, 48.7, 53.1, 51.7, 49.9, …
## $ TOR   <dbl> 20.0, 19.3, 18.8, 16.7, 18.7, 17.8, 15.3, 13.5, 19.8, …
## $ TORD  <dbl> 23.4, 19.4, 16.8, 16.7, 19.8, 20.2, 18.5, 15.7, 21.6, …
## $ FTR   <dbl> 35.7, 39.4, 32.1, 28.3, 28.2, 36.8, 30.7, 31.7, 35.0, …
## $ FTRD  <dbl> 36.6, 33.5, 23.8, 30.0, 31.5, 35.7, 28.7, 27.1, 31.9, …
## $ TEAM  <chr> "VCU", "Saint Louis", "Dayton", "Davidson", "St. Bonav…
## $ YR    <dbl> 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19…
```

```
is.character(CBB_variables$TEAM)
## [1] TRUE
```

```
CBB_variables <- CBB_variables %>% mutate(TEAM = as_factor(TEAM))

class(CBB_variables$TEAM)
## [1] "factor"
```

```
levels(CBB_variables$TEAM)
##   [1] "VCU"                    "Saint Louis"
##   [3] "Dayton"                 "Davidson"
##   [5] "St. Bonaventure"        "Rhode Island"
##   [7] "Richmond"               "Saint Joseph's"
##   [9] "Duquesne"               "George Mason"
##  [11] "La Salle"               "Massachusetts"
##  [13] "Fordham"                "George Washington"
##  [15] "Virginia"               "Duke"
##  [17] "North Carolina"         "Virginia Tech"
##  [19] "Florida St."            "Louisville"
##  [21] "Syracuse"               "Clemson"
##  [23] "North Carolina St."     "Miami FL"
##  [25] "Notre Dame"             "Pittsburgh"
##  [27] "Boston College"         "Georgia Tech"
```

Summary Statistics

Summary Statistics

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
wins	1053	15.178	6.14	0	11	20	35
adjusted_offensive_efficiency	1053	102.503	7.041	80	97.7	107.2	125.4
adjusted_defensive_efficiency	1053	102.51	6.369	85.2	98.1	106.8	122.7
adjusted_fieldgoals_scored	1053	50.055	2.977	39.3	48.1	52	61
adjusted_fieldgoals_allowed	1053	50.202	2.842	41.2	48.3	52.1	60.1
team_turnoverrate	1053	18.832	2.121	13.3	17.3	20.2	26.6
opponent_stealrate	1053	18.773	2.209	12.6	17.3	20.1	27.8
team_freethrowfrequency	1053	32.406	4.792	19.6	28.9	35.7	48.1
opponent_freethrowfrequency	1053	32.632	5.54	19.7	28.8	35.9	55.3
year_played	1053	19.994	0.816	19	19	21	21

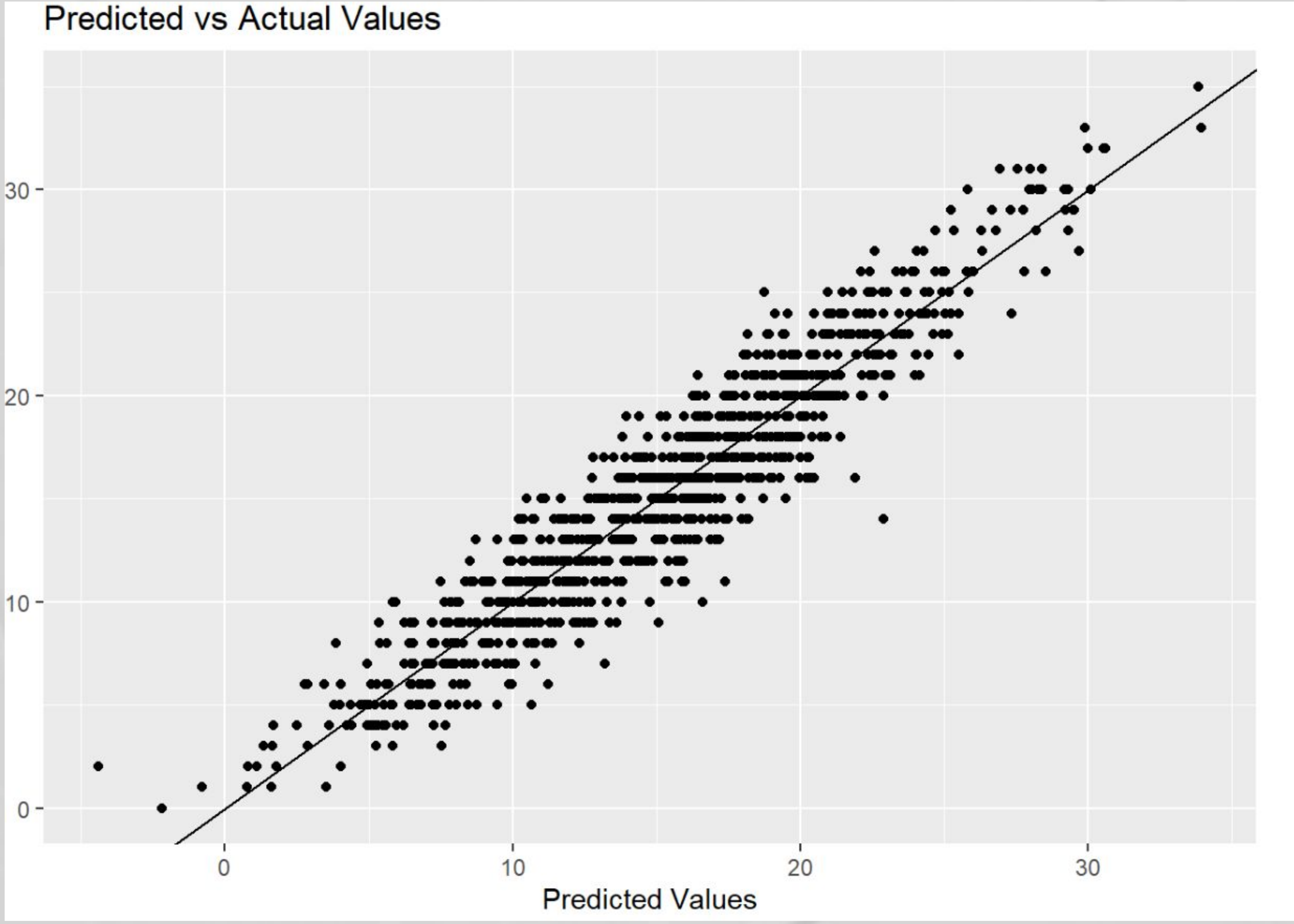
Linear Regression

- Train test split was implemented; 75% training set and 25% testing set
- Predictors used were the selected variables:
  - Adjusted offensive efficiency
  - Adjusted defensive efficiency
  - Adjusted field goals scored
  - Adjusted field goals allowed
  - Team turnover rate
  - Opponent steal rate
  - Team free throw frequency
  - Opponent free throw frequency
  - Name of team
  - Year played

	wins		
Predictors	*Estimates*	*CI*	*p*
(Intercept)	69.60	56.23 – 82.96	**<0.001**
adjusted offensive efficiency	0.45	0.35 – 0.54	**<0.001**
adjusted defensive efficiency	-0.25	-0.35 – -0.15	**<0.001**
adjusted fieldgoals scored	0.17	0.02 – 0.32	**0.025**
adjusted fieldgoals allowed	-0.65	-0.80 – -0.50	**<0.001**
team turnoverrate	-0.18	-0.34 – -0.02	**0.025**
opponent stealrate	0.60	0.44 – 0.75	**<0.001**
team freethrowfrequency	0.13	0.07 – 0.18	**<0.001**
opponent freethrowfrequency	-0.18	-0.24 – -0.13	**<0.001**
name of team [Air Force]	-3.20	-7.40 – 1.00	0.135
name of team [Western Michigan]	-5.10	-9.38 – -0.82	**0.020**
name of team [Wichita St.]	-4.56	-8.85 – -0.27	**0.037**
name of team [William & Mary]	-1.80	-6.05 – 2.46	0.408
name of team [Winthrop]	1.52	-2.58 – 5.62	0.467
name of team [Wisconsin]	-7.88	-12.32 – -3.43	**0.001**
name of team [Wofford]	-0.77	-4.97 – 3.43	0.718
name of team [Wright St.]	-3.53	-7.71 – 0.66	0.099
name of team [Wyoming]	-5.23	-9.37 – -1.08	**0.014**
name of team [Xavier]	-7.91	-12.25 – -3.58	**<0.001**
name of team [Yale]	-3.22	-7.91 – 1.47	0.178
name of team [Youngstown St.]	0.67	-3.56 – 4.90	0.757
year played	-2.62	-2.83 – -2.42	**<0.001**
Observations	1053		
R^2 / R^2 adjusted	0.892 / 0.834		

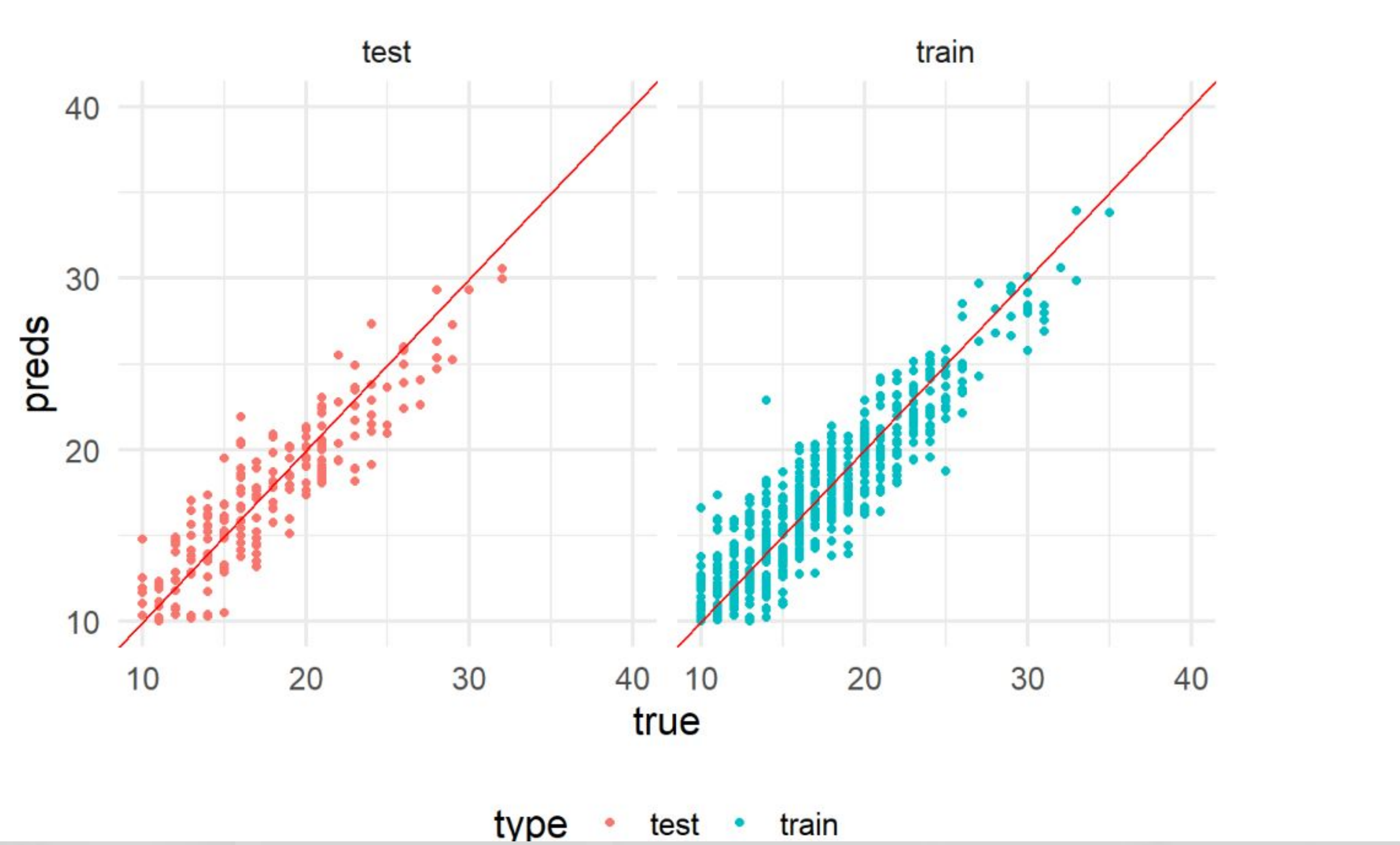
Linear Regression

- The predicted and actual values are both pretty close to the regression line
- Generally, there is minimal variation between the predicted and actual values for our model
  - for every predicted value, the actual value is reasonably close to the predicted value(especially in the ranges of pred values 10-20 and 20-30)
  - The model's R^2 value was 0.89 as mentioned previously, so the performance was relatively strong



Linear Regression

- The graph depicted on the right illustrates the predicted true plot for the test and train sets
  - For both test and train, the data is almost perfectly aligned along the main diagonal
- Calculating the median for both test and train outputted:
  - 1.50328(test)
  - 1.305811(train)
- MedAE informed us about the possible error for median observation
  - Both MedAE's were similarly low, and overall the model doesn't appear to be underfit or overfit




```
get_medae <- function(true, predictions){
  median(abs(true - predictions))
}
```

```
get_medae(results_test$true, results_test$preds)
```

```
## [1] 1.505328
```

```
get_medae(results_train$true, results_train$preds)
```

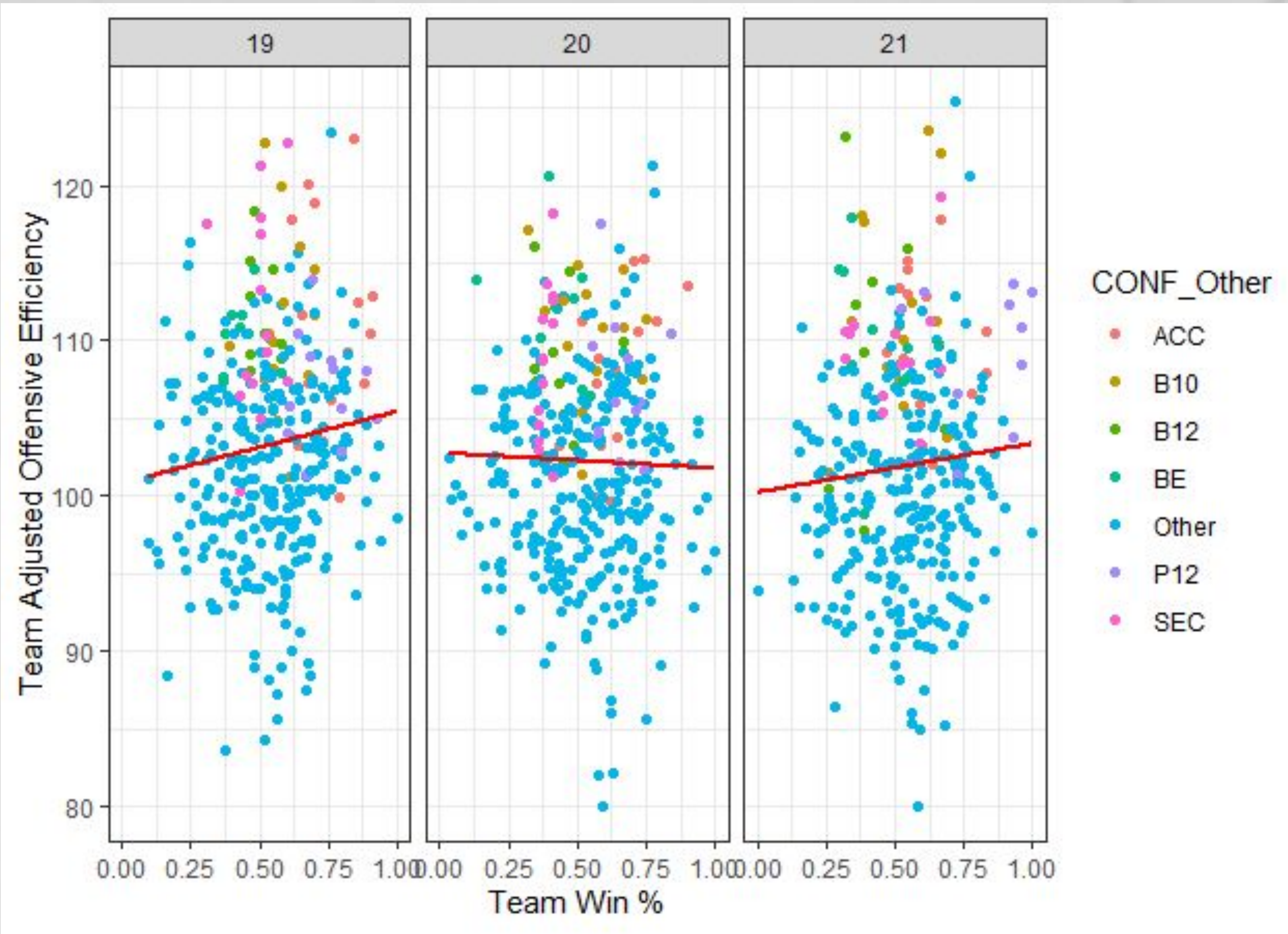
```
## [1] 1.305811
```

Logistic Regression

- Highest coefficient to predict a team’s win percentage was Adjusted Offensive Efficiency
- Graph shows ADJOE vs Win% by year in the dataset

```r
winper_split <- initial_split(CBB, prop = 0.75)
CBB_train <- training(winper_split)
CBB_test <- testing(winper_split)
mod1 <- glm(WINPER ~ ADJOE + ADJDE + EFG_O + EFG_D + TOR +
              TORD + FTR + FTRD, family = quasibinomial, data = CBB_train)
summary(mod1)
ggplot(CBB, aes(x = WINPER, y =  ADJOE, color = CONF_Other)) + geom_point() +
  theme_bw() + facet_wrap(~YR) +
  geom_smooth(method=lm, se=FALSE, col='red') +
  labs(x = "Team Win %", y = "Team Adjusted Offensive Efficiency")
```

```
> summary(mod1)

Call:
glm(formula = WINPER ~ ADJOE + ADJDE + EFG_O + EFG_D + TOR +
    TORD + FTR + FTRD, family = quasibinomial, data = CBB_train)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.14376  -0.25904   0.00594   0.26234   1.15654

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.438799   1.175873   2.074  0.03840 *
ADJOE        0.013679   0.007614   1.797  0.07279 .
ADJDE       -0.014247   0.009112  -1.563  0.11836
EFG_O       -0.037459   0.013140  -2.851  0.00448 **
EFG_D       -0.003049   0.016213  -0.188  0.85087
TOR          0.011761   0.015746   0.747  0.45535
TORD        -0.024778   0.014935  -1.659  0.09752 .
FTR         -0.004374   0.005948  -0.735  0.46238
FTRD         0.005005   0.005643   0.887  0.37535
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for quasibinomial family taken to be 0.1301831)

    Null deviance: 113.72  on 788  degrees of freedom
Residual deviance: 110.12  on 780  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 3
```



Ridge Model

- Effective Field Goal Percentage Allowed most negative impact

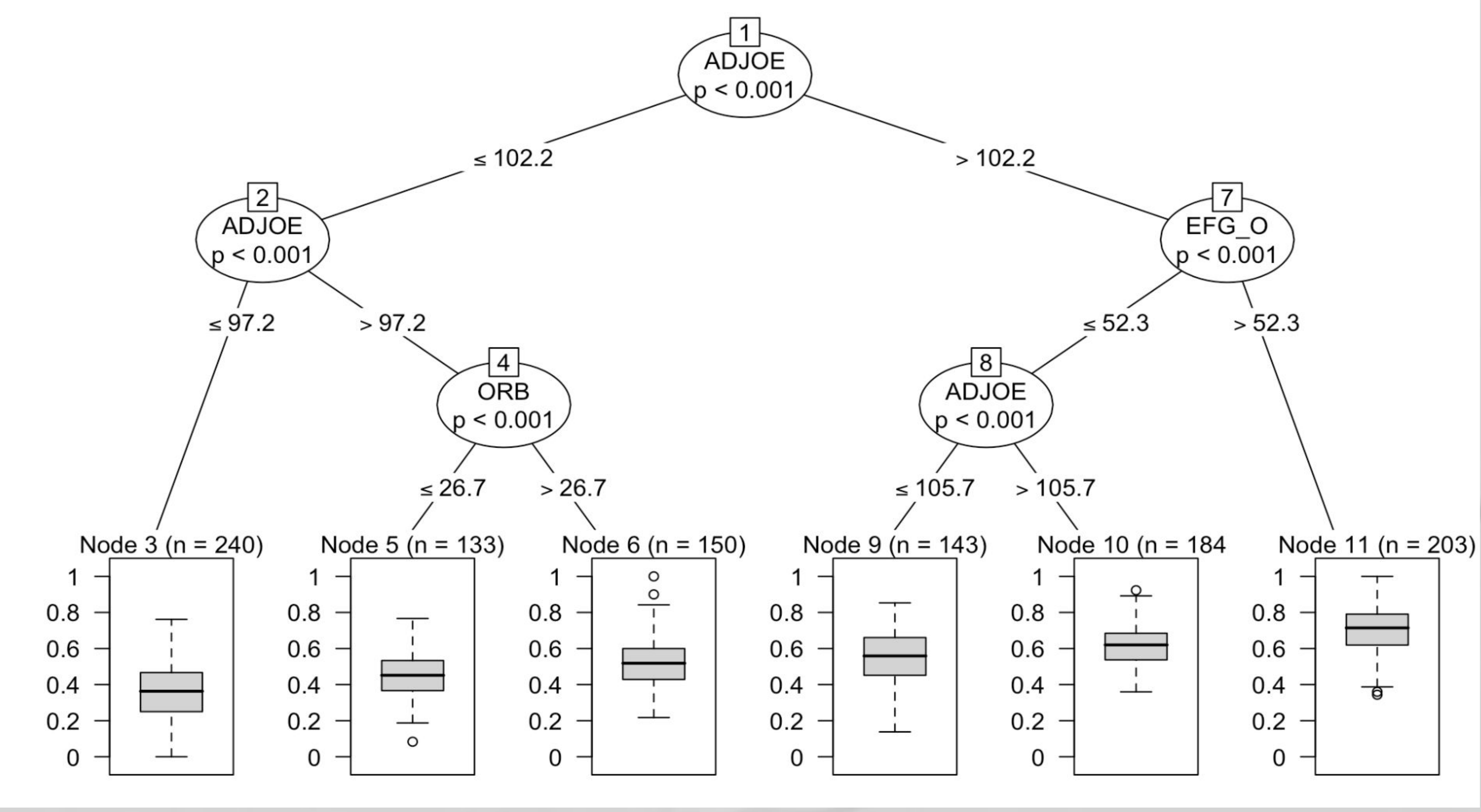
- Effective Field Goal Percentage Taken most positive impact

```
## (Intercept)       58.106
## ADJOE              0.182
## ADJDE             -0.156
## EFG_O              0.377
## EFG_D             -0.494
## TOR               -0.394
## TORD               0.361
## FTR                0.132
## FTRD              -0.079
```

Tree Regression - Offense

- Goal: Predicting team win percentage given the offense variables: ADJOE, EFG_O, TOR, FTR, and ORB
- The first split shows that Adjusted Offensive Efficiency is the most important of the offense variables

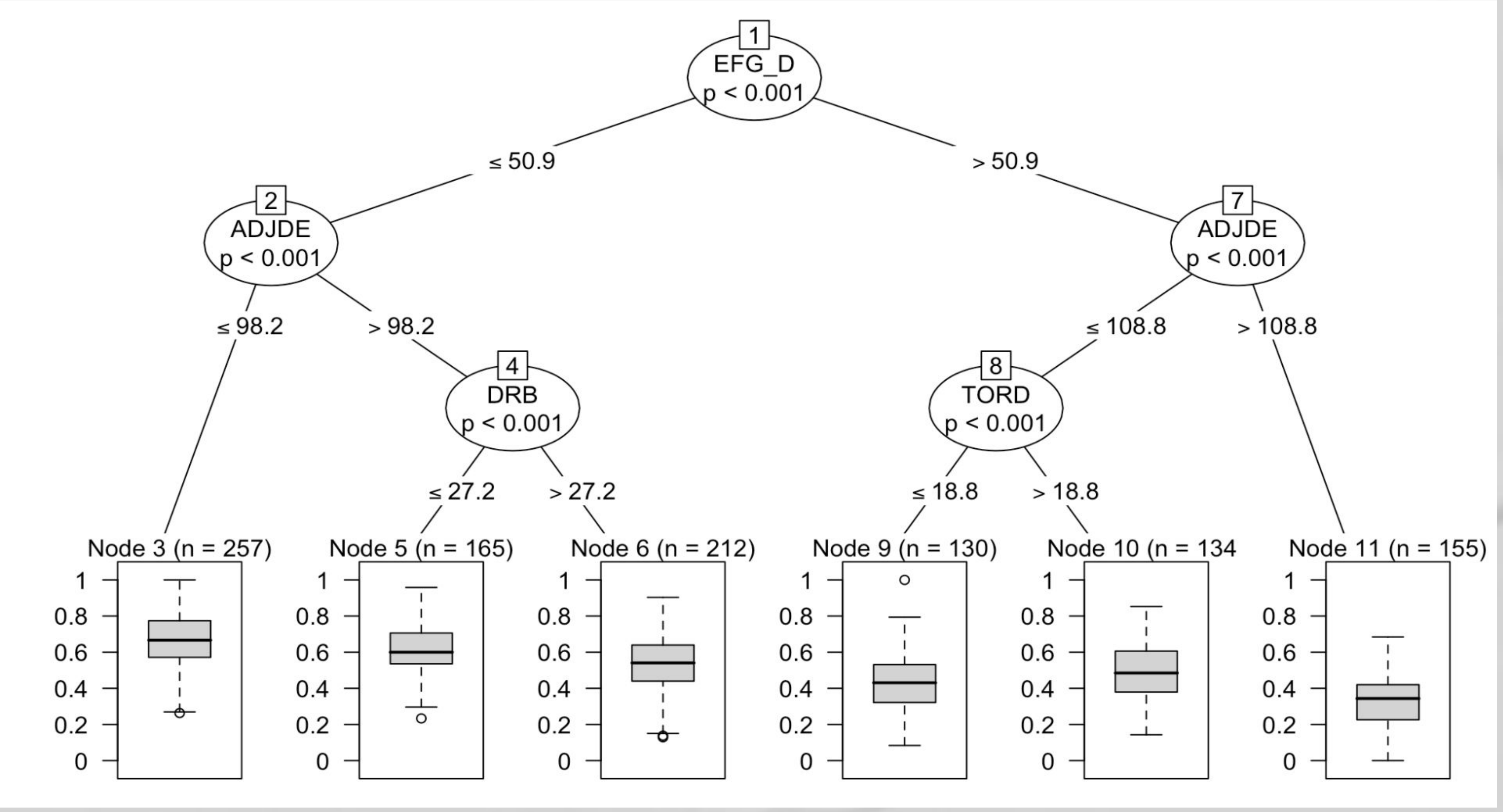
```
Model formula:
WINPER ~ ADJOE + EFG_O + TOR + FTR + ORB

Fitted party:
[1] root
|   [2] ADJOE <= 102.2
|   |   [3] ADJOE <= 97.2: 0.368 (n = 240, err = 5.5)
|   |   [4] ADJOE > 97.2
|   |   |   [5] ORB <= 26.7: 0.457 (n = 133, err = 2.2)
|   |   |   [6] ORB > 26.7: 0.521 (n = 150, err = 3.5)
|   [7] ADJOE > 102.2
|   |   [8] EFG_O <= 52.3
|   |   |   [9] ADJOE <= 105.7: 0.556 (n = 143, err = 3.0)
|   |   |   [10] ADJOE > 105.7: 0.616 (n = 184, err = 2.4)
|   |   [11] EFG_O > 52.3: 0.707 (n = 203, err = 3.8)

Number of inner nodes:    5
Number of terminal nodes: 6
```



Tree Regression - Defense

- Goal: Predicting team win percentage given the offense variables: ADJDE, EFG_D, TORD, FTRD, and DRB
- The first split shows that Effective Field Goal Percentage Allowed is the most important of the defense variables

```
Model formula:
WINPER ~ ADJDE + EFG_D + TORD + FTRD + DRB

Fitted party:
[1] root
|   [2] EFG_D <= 50.9
|   |   [3] ADJDE <= 98.2: 0.672 (n = 257, err = 5.2)
|   |   [4] ADJDE > 98.2
|   |   |   [5] DRB <= 27.2: 0.612 (n = 165, err = 3.2)
|   |   |   [6] DRB > 27.2: 0.537 (n = 212, err = 4.5)
|   [7] EFG_D > 50.9
|   |   [8] ADJDE <= 108.8
|   |   |   [9] TORD <= 18.8: 0.438 (n = 130, err = 3.0)
|   |   |   [10] TORD > 18.8: 0.500 (n = 134, err = 3.0)
|   |   [11] ADJDE > 108.8: 0.337 (n = 155, err = 3.1)

Number of inner nodes:    5
Number of terminal nodes: 6
```



Conclusion

- Model we would recommend:
  - Linear Regression
- This model is applicable in the following situations:
  - Coaching
  - Betting
- Github:

  https://github.com/ryanking916/CBB-Project