Predicting NFL Statistics

Rich Lindberg

March 27, 2017

Predicting NFL Stats

The goal of this capstone project was to set a baseline linear regression for predicting NFL statistics. The use of the analysis would be to project player performance and see if the team I am working with needs to consider making adjustments given various factors of the upcoming game/season.

Where to get the data

I went to the website http://armchairanalysis.com/data.php. I have a subscription to the database, so I connected into it via SQL. I downloaded the historical database onto my hard drive, and mapped it in MySQL.

I then queried the DB to get the fields I would need. This operation took extensive time, so once it ran, I exported to a csv file, then read the csv into R.

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.3.2
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 3.3.2
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.3.2
```

```
library(reshape2)
```

```
## Warning: package 'reshape2' was built under R version 3.3.2
```

```
##
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
library(caTools)
```

```
## Warning: package 'caTools' was built under R version 3.3.3
```

```
nfl_data <- read.csv("NFL_offense.csv")
```

The data is pretty clean from armchairanalyis, fivethirtyeight.com uses this website for its sports data, so it is a pretty reputable site.

I felt there were pieces of data either missing, or needing cleaning up. This brought on the fun process of cleaning and tidying the data.

Weather and field conditions

From a qualitative perspective, we know that field turf and ideal temperatures are the least inhibitive towards speed, according to players themselves. I wanted to identify extremes and hinderences.

I replaced all "NULL" temp fields with a generic "room" temperature assumption.

cold_weather and hot_weather were fields created to identify extreme ends of the temperature spectrum, and see if they have an impact on play

```
#make all null temperatures at game time "room" temperature
nfl_data$temp[nfl_data$temp == "NULL"] <- 70
nfl_data$temp <- as.integer(nfl_data$temp)

#highlight temp extremes
nfl_data <- mutate(nfl_data, cold_weather= ifelse(temp < 45, 1,0))
nfl_data <- mutate(nfl_data, hot_weather=  ifelse(temp > 85, 1,0))

#weather factors
nfl_data <- mutate(nfl_data, grass_1 = ifelse(surf == "DD GrassMaster" | surf == "Grass",
                                              1,0))
nfl_data <- mutate(nfl_data, bad_weather_1 = ifelse(cond == "Light Rain" |
                                                      cond == "Rain" |
                                                      cond == "Flurries" |
                                                      cond == "Snow" |
                                                      cond == "Foggy" |
                                                      cond == "Windy" |
                                                      cond == "Hazy" |
                                                      cond == "Thunderstorms"|
                                                      cond == "Light Snow" |
                                                      cond == "Light Showers" ,1,0))
```

Home field advantage

Do players play better at home?

```
#identify home team
nfl_data$h <- as.character(nfl_data$h)
nfl_data$team <- as.character(nfl_data$team)
nfl_data <- mutate(nfl_data, home_team_1=  ifelse(h == team, 1,0))
```

Positions

Ignoring player stats, does the position matter

```
#identify position
nfl_data <- mutate(nfl_data, is_WR =  ifelse(pos1 == "WR", 1,0))
nfl_data <- mutate(nfl_data, is_TE =  ifelse(pos1 == "TE", 1,0))
```

```
nfl_data <- mutate(nfl_data, is_RB =  ifelse(pos1 == "RB", 1,0))
nfl_data <- mutate(nfl_data, is_QB =  ifelse(pos1 == "QB", 1,0))
```

Age

Every year players get older, so we want to know "Does father time impact player performance?"

```
#age
nfl_data <- mutate(nfl_data, age = year - yob)
```

Combine cleanup

The NFL combine is an event where prospective new players work out for the entire league to see. Their physical measurements are taken, and people find merit in this event. I wanted to see if these stats had any impact on player performance. Not all players attend the combine. For the fields where there are zeroes for the combine stat, I took the average for all non-zero stats for that position. This basically implies if you didn't attend the combine, your stats are middle of the road.

```
#replace 0 forty with avg for position
nfl_data <- nfl_data %>%
  group_by(pos1)%>%
  mutate(forty1 = ifelse(forty == 0, mean(forty[forty>0]), forty))
#replace 0 vertical with average for position
nfl_data <- nfl_data %>%
  group_by(pos1)%>%
  mutate(vertical1 = ifelse(vertical == 0, mean(vertical[vertical>0]), vertical))
#replace 0 arm length with formula for 40% of height is arm
nfl_data$arm <- ifelse(nfl_data$arm == 0, nfl_data$height*0.4, nfl_data$arm)
nfl_data <- nfl_data %>%
  group_by(pos1)%>%
  mutate(shuttle1 = ifelse(shuttle == 0, mean(shuttle[shuttle>0]), shuttle))
nfl_data <- nfl_data %>%
  group_by(pos1)%>%
  mutate(cone1 = ifelse(cone == 0, mean(cone[cone>0]), cone))
```

NFL Teams

I created fields for teams (1 if player plays for that team in the header 0 if it doesn't). I also cleaned up one team: The St Louis/LA Rams. The Rams moved in 2016 to LA, so the conditions of stadium changed. I combined the field into a single field.

```
#clean teams and give each team a field
nfl_data <- mutate(nfl_data, Teams =  ifelse(team == "STL" | team == "LA", "STL/LA",team))
nfl_data <- mutate(nfl_data, ARI = ifelse(Teams == "ARI",1,0))
nfl_data <- mutate(nfl_data, ATL = ifelse(Teams == "ATL",1,0))
nfl_data <- mutate(nfl_data, BAL = ifelse(Teams == "BAL",1,0))
nfl_data <- mutate(nfl_data, BUF = ifelse(Teams == "BUF",1,0))
nfl_data <- mutate(nfl_data, CAR = ifelse(Teams == "CAR",1,0))
nfl_data <- mutate(nfl_data, CHI = ifelse(Teams == "CHI",1,0))
nfl_data <- mutate(nfl_data, CIN = ifelse(Teams == "CIN",1,0))
nfl_data <- mutate(nfl_data, CLE = ifelse(Teams == "CLE",1,0))
nfl_data <- mutate(nfl_data, DAL = ifelse(Teams == "DAL",1,0))
nfl_data <- mutate(nfl_data, DEN = ifelse(Teams == "DEN",1,0))
nfl_data <- mutate(nfl_data, DET = ifelse(Teams == "DET",1,0))
```

```r
nfl_data <- mutate(nfl_data, GB = ifelse(Teams == "GB",1,0))
nfl_data <- mutate(nfl_data, HOU = ifelse(Teams == "HOU",1,0))
nfl_data <- mutate(nfl_data, IND = ifelse(Teams == "IND",1,0))
nfl_data <- mutate(nfl_data, JAC = ifelse(Teams == "JAC",1,0))
nfl_data <- mutate(nfl_data, KC = ifelse(Teams == "KC",1,0))
nfl_data <- mutate(nfl_data, MIA = ifelse(Teams == "MIA",1,0))
nfl_data <- mutate(nfl_data, MINN = ifelse(Teams == "MIN",1,0))
nfl_data <- mutate(nfl_data, NE = ifelse(Teams == "NE",1,0))
nfl_data <- mutate(nfl_data, NOR = ifelse(Teams == "NO",1,0))
nfl_data <- mutate(nfl_data, NYG = ifelse(Teams == "NYG",1,0))
nfl_data <- mutate(nfl_data, NYJ = ifelse(Teams == "NYJ",1,0))
nfl_data <- mutate(nfl_data, OAK = ifelse(Teams == "OAK",1,0))
nfl_data <- mutate(nfl_data, PHI = ifelse(Teams == "PHI",1,0))
nfl_data <- mutate(nfl_data, PIT = ifelse(Teams == "PIT",1,0))
nfl_data <- mutate(nfl_data, SD = ifelse(Teams == "SD",1,0))
nfl_data <- mutate(nfl_data, SEA = ifelse(Teams == "SEA",1,0))
nfl_data <- mutate(nfl_data, SF = ifelse(Teams == "SF",1,0))
nfl_data <- mutate(nfl_data, STL = ifelse(Teams == "STL/LA",1,0))
nfl_data <- mutate(nfl_data, TB = ifelse(Teams == "TB",1,0))
nfl_data <- mutate(nfl_data, TEN = ifelse(Teams == "TEN",1,0))
nfl_data <- mutate(nfl_data, WAS = ifelse(Teams == "WAS",1,0))
```

Receiving Stats

For receiving, I wanted to get every players average: * yards * receptions * targets * touchdowns

I also wanted to get every position average, and average for team. Rationale for at least having that info is this: Compare player to team to league wide position

```r
#calculate the averages by player, position, and team
#receiving
nfl_data <- nfl_data %>%
              group_by(player.1)%>%
                mutate(avg_recy_plyr = mean(recy))
nfl_data <- nfl_data %>%
              group_by(pos1)%>%
                mutate(avg_recy_pos = mean(recy))
nfl_data <- nfl_data %>%
              group_by(Teams)%>%
                mutate(avg_recy_team = mean(recy))
nfl_data <- nfl_data %>%
              group_by(player.1)%>%
                mutate(avg_rec_plyr = mean(rec))
nfl_data <- nfl_data %>%
              group_by(pos1)%>%
                mutate(avg_rec_pos = mean(rec))
nfl_data <- nfl_data %>%
              group_by(Teams)%>%
                mutate(avg_rec_team = mean(rec))
nfl_data <- nfl_data %>%
              group_by(player.1)%>%
                mutate(avg_trg_plyr = mean(trg))
nfl_data <- nfl_data %>%
              group_by(pos1)%>%
```

```
                mutate(avg_trg_pos = mean(trg))
nfl_data <- nfl_data %>%
              group_by(Teams)%>%
                mutate(avg_trg_team = mean(trg))
nfl_data <- nfl_data %>%
              group_by(player.1)%>%
                mutate(avg_rectd_plyr = mean(tdrec))
nfl_data <- nfl_data %>%
              group_by(pos1)%>%
                mutate(avg_rectd_pos = mean(tdrec))
nfl_data <- nfl_data %>%
              group_by(Teams)%>%
                mutate(avg_rectd_team = mean(tdrec))
```

Running Stats

I followed a similar process from up above.
The stats I was looking for the mean for were: * rushing attempts * rushing yards * fumbles

```
#running
nfl_data <- nfl_data %>%
  group_by(player.1)%>%
  mutate(avg_rbra_plyr = mean(ra))

nfl_data <- nfl_data %>%
  group_by(Teams)%>%
  mutate(avg_rbra_team = mean(ra))

nfl_data <- nfl_data %>%
  group_by(pos1)%>%
  mutate(avg_rbra_pos = mean(ra))

nfl_data <- nfl_data %>%
  group_by(player.1)%>%
  mutate(avg_rbry_plyr = mean(ry))

nfl_data <- nfl_data %>%
  group_by(Teams)%>%
  mutate(avg_rbry_team = mean(ry))

nfl_data <- nfl_data %>%
  group_by(pos1)%>%
  mutate(avg_rbry_pos = mean(ry))

nfl_data <- nfl_data %>%
  group_by(player.1)%>%
  mutate(avg_fuml_plyr = mean(fuml))

nfl_data <- nfl_data %>%
  group_by(Teams)%>%
  mutate(avg_fuml_team = mean(fuml))

nfl_data <- nfl_data %>%
```

```
  group_by(pos1)%>%
  mutate(avg_fuml_pos = mean(fuml))

nfl_data <- nfl_data %>%
  group_by(player.1)%>%
  mutate(avg_tdr_plyr = mean(tdr))

nfl_data <- nfl_data %>%
  group_by(pos1)%>%
  mutate(avg_tdr_pos = mean(tdr))

nfl_data <- nfl_data %>%
  group_by(Teams)%>%
  mutate(avg_tdr_team = mean(tdr))
```

Passing

I followed a similar process from up above.
The stats I was looking for the mean for were: * passing yards * passing attempts * passing completions * passing touchdowns * interceptions

```
#passing
nfl_data <- nfl_data %>%
  group_by(player.1)%>%
  mutate(avg_qbpy_plyr = mean(py))

nfl_data <- nfl_data %>%
  group_by(Teams)%>%
  mutate(avg_qbpy_team = mean(py))

nfl_data <- nfl_data %>%
  group_by(pos1)%>%
  mutate(avg_qbpy_pos = mean(py))


nfl_data <- nfl_data %>%
  group_by(player.1)%>%
  mutate(avg_qbpc_plyr = mean(pc))

nfl_data <- nfl_data %>%
  group_by(Teams)%>%
  mutate(avg_qbpc_team = mean(pc))

nfl_data <- nfl_data %>%
  group_by(pos1)%>%
  mutate(avg_qbpc_pos = mean(pc))

nfl_data <- nfl_data %>%
  group_by(player.1)%>%
  mutate(avg_qbints_plyr = mean(ints))

nfl_data <- nfl_data %>%
  group_by(Teams)%>%
```

```
  mutate(avg_qbints_team = mean(ints))

nfl_data <- nfl_data %>%
  group_by(pos1)%>%
  mutate(avg_qbints_pos = mean(ints))

nfl_data <- nfl_data %>%
  group_by(player.1)%>%
  mutate(avg_qbtdp_plyr = mean(tdp))

nfl_data <- nfl_data %>%
  group_by(Teams)%>%
  mutate(avg_qbtdp_team = mean(tdp))

nfl_data <- nfl_data %>%
  group_by(pos1)%>%
  mutate(avg_qbtdp_pos = mean(tdp))

nfl_data <- nfl_data %>%
  group_by(player.1)%>%
  mutate(avg_qbpa_plyr = mean(pa))

nfl_data <- nfl_data %>%
  group_by(Teams)%>%
  mutate(avg_qbpa_team = mean(pa))

nfl_data <- nfl_data %>%
  group_by(pos1)%>%
  mutate(avg_qbpa_pos = mean(pa))
```

Pre-Analysis exploration

I wanted to learn a little about the data. I had approximately 100 fields to choose from, so it is hard to infer if there were any correlations or other patterns off hand. I built a subset of the fields I wanted to explore. I named this subset: nfl_data_fields

I then created a correlation matrix:

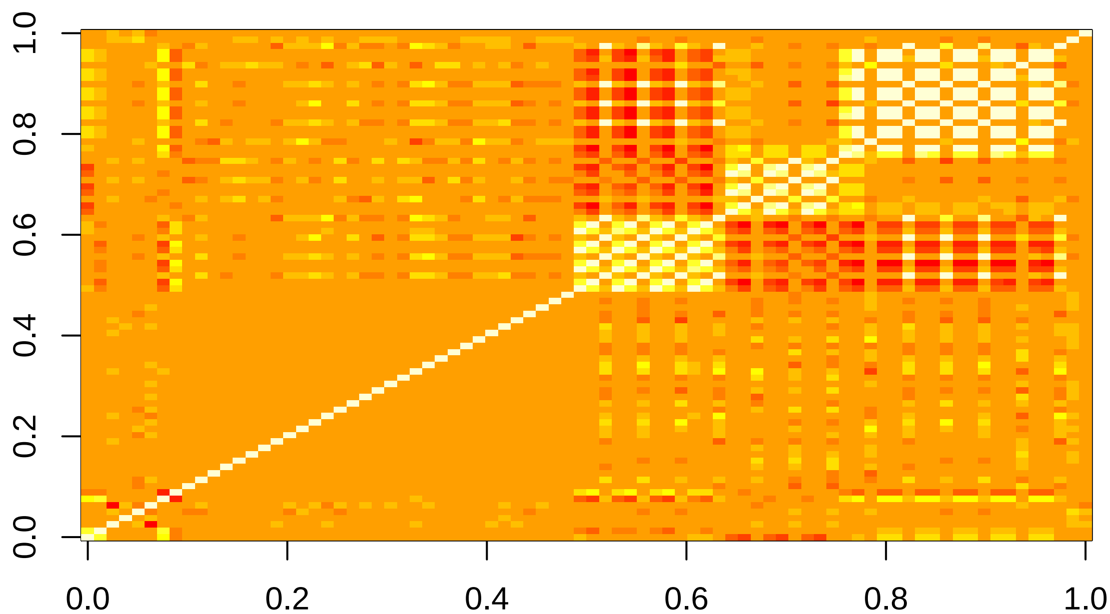
```
nfl_data_fields<- subset(nfl_data, select = c("height", "weight", "cold_weather", "hot_weather",
                         "home_team_1", "temp",
                         "forty1", "vertical1", "ARI", "ATL", "BAL", "BUF",
                         "CAR", "CHI", "CIN", "CLE", "DAL", "DEN", "DET", "GB", "HOU", "IND",
                         "JAC", "KC", "MIA", "MINN", "NE", "NOR", "NYG", "NYJ", "OAK", "PHI", "PIT",
                         "SD", "SEA", "STL", "TB", "TEN", "WAS",
                         "avg_recy_plyr","avg_recy_pos","avg_recy_team","avg_rec_plyr","avg_rec_pos",
                         "avg_rec_team", "avg_trg_plyr","avg_trg_pos","avg_trg_team","avg_rectd_plyr",
                         "avg_rectd_pos","avg_rectd_team","avg_tdr_plyr","avg_tdr_pos","avg_tdr_team",
                         "avg_rbra_plyr","avg_rbra_pos", "avg_rbra_team","avg_rbry_plyr","avg_rbry_pos"
                         "avg_rbry_team","avg_fuml_plyr","avg_fuml_pos", "avg_fuml_team","avg_qbpy_plyr
                         "avg_qbpy_pos", "avg_qbpy_team","avg_qbpa_plyr","avg_qbpa_pos","avg_qbpa_team"
                         "avg_qbpc_plyr","avg_qbpc_pos", "avg_qbpc_team","avg_qbints_plyr", "avg_qbints_
                         "avg_qbints_team","avg_qbtdp_plyr","avg_qbtdp_pos","avg_qbtdp_team","grass_1",
                         "bad_weather_1"))
```

```r
cor_nfl <- cor(nfl_data_fields)
```

If you were to run cor_nfl, the print out is extremely difficult to read and gets cut off because of it's size.

I decided a simpler view should be created, so I created an image of the matrix

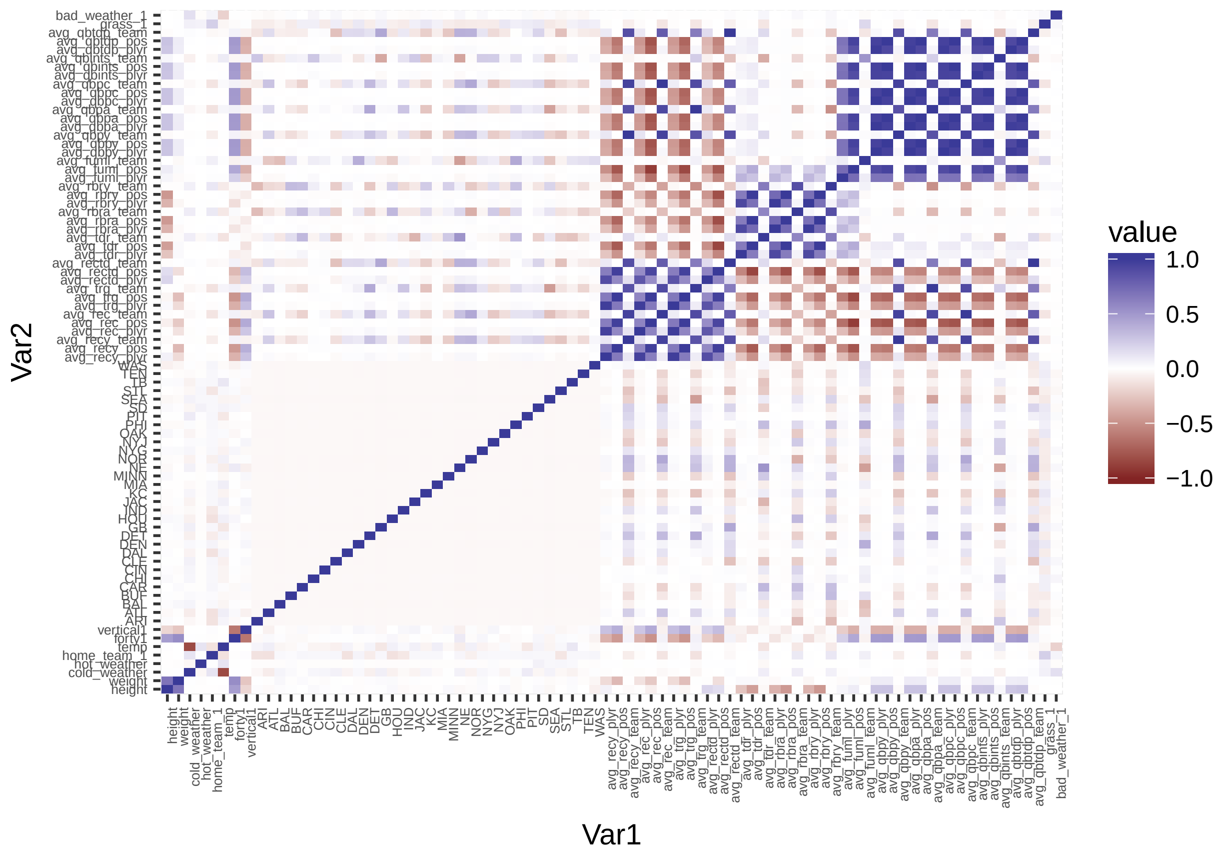
```r
image(cor_nfl)
```



This gives a pretty interesting view of the data, but is still hard to interpret. That being said, it was on the right path.

I then used qplot to get a better view of the data. With qplot I can control the colors, and since correlation matrices range from -1 to 1, setting the bookends of the color spectrum based on the values of the correlation would give me a very indicative heatmap

```r
qplot(x=Var1, y=Var2, data=melt(cor(nfl_data_fields)), fill=value, geom="tile")+
  scale_fill_gradient2(limits=c(-1, 1))+
  theme(axis.text.x = element_text(angle = 90, hjust = 1, size = 5),
        axis.text.y = element_text(size = 5))
```



There are strong relationships where we calculated general averages for the position, teams and players. That is not surprising since a stat like rushing will have strong relationships with how many attempts you make at rushing the ball. Typically, if you are running the ball more, you should see more yards. Obvious, but this confirms the correlation.

Receiving Regressions

We have to set up our test and training data:

```
set.seed(123)
split <- sample.split(nfl_data$recy, SplitRatio = 0.7)
TrainRecy <- subset(nfl_data, split == TRUE)
TestRecy <- subset(nfl_data, split == FALSE)
```

Now that we are ready to train and test the data, let's do it!

Receiving Yards first run

```
linRegrecy <- lm(recy ~ height+ weight + cold_weather + hot_weather + home_team_1+ temp+ forty1 + verti
                    NYJ + OAK + PHI + PIT +SD + SEA + STL + TB + TEN + WAS + avg_recy_plyr+avg_recy_pos +
                    avg_recy_team + avg_rec_plyr +avg_rec_pos + avg_rec_team +avg_trg_plyr + avg_trg_pos
                    avg_trg_team + avg_rectd_plyr + avg_rectd_pos +avg_rectd_team+
                    avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
                    avg_rbra_plyr + avg_rbra_pos +avg_rbra_team +
                    avg_rbry_plyr + avg_rbry_pos +avg_rbry_team +
```

```r
                   avg_fuml_plyr + avg_fuml_pos +avg_fuml_team +
                   avg_qbpy_plyr + avg_qbpy_pos +avg_qbpy_team +
                   avg_qbpa_plyr + avg_qbpa_pos +avg_qbpa_team+
                   avg_qbpc_plyr + avg_qbpc_pos +avg_qbpc_team +
                   avg_qbints_plyr + avg_qbints_pos +avg_qbints_team +
                   avg_qbtdp_plyr + avg_qbtdp_pos +avg_qbtdp_team +
                   grass_1 + bad_weather_1, data = nfl_data)

summary(linRegrecy)
```

```
##
## Call:
## lm(formula = recy ~ height + weight + cold_weather + hot_weather +
##     home_team_1 + temp + forty1 + vertical1 + shuttle1 + cone1 +
##     ARI + ATL + BAL + BUF + CAR + CHI + CIN + CLE + DAL + DEN +
##     DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR +
##     NYG + NYJ + OAK + PHI + PIT + SD + SEA + STL + TB + TEN +
##     WAS + avg_recy_plyr + avg_recy_pos + avg_recy_team + avg_rec_plyr +
##     avg_rec_pos + avg_rec_team + avg_trg_plyr + avg_trg_pos +
##     avg_trg_team + avg_rectd_plyr + avg_rectd_pos + avg_rectd_team +
##     avg_tdr_plyr + avg_tdr_pos + avg_tdr_team + avg_rbra_plyr +
##     avg_rbra_pos + avg_rbra_team + avg_rbry_plyr + avg_rbry_pos +
##     avg_rbry_team + avg_fuml_plyr + avg_fuml_pos + avg_fuml_team +
##     avg_qbpy_plyr + avg_qbpy_pos + avg_qbpy_team + avg_qbpa_plyr +
##     avg_qbpa_pos + avg_qbpa_team + avg_qbpc_plyr + avg_qbpc_pos +
##     avg_qbpc_team + avg_qbints_plyr + avg_qbints_pos + avg_qbints_team +
##     avg_qbtdp_plyr + avg_qbtdp_pos + avg_qbtdp_team + grass_1 +
##     bad_weather_1, data = nfl_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -98.419 -12.000  -1.734   6.465 232.390
##
## Coefficients: (18 not defined because of singularities)
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.057970   9.933129   0.207 0.835868
## height           -0.038325   0.085704  -0.447 0.654750
## weight            0.002289   0.011435   0.200 0.841339
## cold_weather     -0.682754   0.517743  -1.319 0.187273
## hot_weather      -1.647985   2.045951  -0.805 0.420544
## home_team_1       0.218408   0.269888   0.809 0.418373
## temp              0.012260   0.014751   0.831 0.405896
## forty1           -0.210646   1.361048  -0.155 0.877005
## vertical1         0.006676   0.055595   0.120 0.904415
## shuttle1         -0.328503   1.174926  -0.280 0.779790
## cone1             0.052920   0.857337   0.062 0.950781
## ARI               1.926183   0.985713   1.954 0.050696 .
## ATL               2.645292   0.986540   2.681 0.007335 **
## BAL               2.808884   0.957663   2.933 0.003358 **
## BUF               2.583370   0.996958   2.591 0.009566 **
## CAR               1.921807   0.975734   1.970 0.048891 *
## CHI               2.658690   0.996948   2.667 0.007660 **
## CIN               2.431601   0.988469   2.460 0.013899 *
## CLE               3.255099   0.978635   3.326 0.000881 ***
```

```
## DAL              2.071710   0.998979   2.074 0.038102 *
## DEN              2.344790   0.961115   2.440 0.014706 *
## DET              1.950222   0.991697   1.967 0.049242 *
## GB               3.247170   0.966687   3.359 0.000783 ***
## HOU              2.989320   0.981071   3.047 0.002313 **
## IND              1.221505   0.980764   1.245 0.212969
## JAC              2.873811   0.993145   2.894 0.003810 **
## KC               2.628836   0.969487   2.712 0.006699 **
## MIA              1.336018   0.991161   1.348 0.177688
## MINN             1.791633   0.993568   1.803 0.071360 .
## NE               2.795808   0.975699   2.865 0.004166 **
## NOR              2.817817   0.980680   2.873 0.004064 **
## NYG              3.343864   0.981834   3.406 0.000661 ***
## NYJ              1.586744   0.982028   1.616 0.106149
## OAK              3.014011   0.967162   3.116 0.001832 **
## PHI              1.236107   0.983721   1.257 0.208920
## PIT              1.454012   0.972049   1.496 0.134708
## SD               3.029863   0.992367   3.053 0.002266 **
## SEA              0.587397   0.957559   0.613 0.539594
## STL              2.040074   0.990183   2.060 0.039376 *
## TB               2.489462   0.985848   2.525 0.011567 *
## TEN              2.507334   0.972806   2.577 0.009958 **
## WAS              2.353757   0.969766   2.427 0.015223 *
## avg_recy_plyr    1.017963   0.035248  28.880  < 2e-16 ***
## avg_recy_pos    -0.160468   4.531690  -0.035 0.971753
## avg_recy_team          NA         NA      NA       NA
## avg_rec_plyr    -0.179048   0.543808  -0.329 0.741969
## avg_rec_pos      0.042182  10.659064   0.004 0.996842
## avg_rec_team           NA         NA      NA       NA
## avg_trg_plyr    -0.008485   0.398804  -0.021 0.983026
## avg_trg_pos      1.490538  51.274041   0.029 0.976809
## avg_trg_team           NA         NA      NA       NA
## avg_rectd_plyr  -0.620218   1.884906  -0.329 0.742124
## avg_rectd_pos   -7.414988 234.910227  -0.032 0.974819
## avg_rectd_team         NA         NA      NA       NA
## avg_tdr_plyr    -0.033980   2.324859  -0.015 0.988339
## avg_tdr_pos      5.284057 114.995513   0.046 0.963350
## avg_tdr_team           NA         NA      NA       NA
## avg_rbra_plyr   -0.082981   0.224684  -0.369 0.711888
## avg_rbra_pos     0.017673   8.199564   0.002 0.998280
## avg_rbra_team          NA         NA      NA       NA
## avg_rbry_plyr    0.021557   0.052194   0.413 0.679592
## avg_rbry_pos    -0.082653   0.747991  -0.111 0.912013
## avg_rbry_team          NA         NA      NA       NA
## avg_fuml_plyr    0.098736   2.782721   0.035 0.971696
## avg_fuml_pos     1.045914  32.845700   0.032 0.974597
## avg_fuml_team          NA         NA      NA       NA
## avg_qbpy_plyr    0.017355   0.044076   0.394 0.693762
## avg_qbpy_pos           NA         NA      NA       NA
## avg_qbpy_team          NA         NA      NA       NA
## avg_qbpa_plyr    0.024448   0.343151   0.071 0.943204
## avg_qbpa_pos           NA         NA      NA       NA
## avg_qbpa_team          NA         NA      NA       NA
## avg_qbpc_plyr   -0.167068   0.619851  -0.270 0.787524
```

```
## avg_qbpc_pos             NA         NA      NA       NA
## avg_qbpc_team            NA         NA      NA       NA
## avg_qbints_plyr  -0.425143   1.984999  -0.214 0.830409
## avg_qbints_pos           NA         NA      NA       NA
## avg_qbints_team          NA         NA      NA       NA
## avg_qbtdp_plyr   -0.874542   2.173354  -0.402 0.687397
## avg_qbtdp_pos            NA         NA      NA       NA
## avg_qbtdp_team           NA         NA      NA       NA
## grass_1          -0.784849   0.278239  -2.821 0.004793 **
## bad_weather_1    -1.032719   0.553706  -1.865 0.062175 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 23.99 on 39189 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.4333, Adjusted R-squared:  0.4324
## F-statistic: 468.2 on 64 and 39189 DF,  p-value: < 2.2e-16
```

Second run at Receiving Yards

```
linRegrecy2 <- lm(recy ~ avg_recy_plyr+grass_1+bad_weather_1, data = TrainRecy)

summary(linRegrecy2)
```

```
##
## Call:
## lm(formula = recy ~ avg_recy_plyr + grass_1 + bad_weather_1,
##     data = TrainRecy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -89.564 -12.133  -1.731   6.620 232.436
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.49287    0.26966   1.828  0.06759 .
## avg_recy_plyr  0.99938    0.00698 143.183  < 2e-16 ***
## grass_1       -0.81124    0.29265  -2.772  0.00557 **
## bad_weather_1 -1.69067    0.64854  -2.607  0.00914 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 24.23 on 27480 degrees of freedom
## Multiple R-squared:  0.4277, Adjusted R-squared:  0.4276
## F-statistic:  6845 on 3 and 27480 DF,  p-value: < 2.2e-16
```

Modest gains in r-square and residual standard error and we cut the variables down to just 3. R2 is not very strong 0.4277. This is an improvement (slightly) over the historical average only as the explanatory variable.

Testing the data, we see that the training set and the test set are similar. The model seems to hold up through testing

```
RecyPredicted <- predict(linRegrecy2, newdata = TestRecy)

SSErecy <- sum((RecyPredicted - TestRecy$recy)^2)
```

```r
SSTrecy <- sum((mean(nfl_data$recy)-TestRecy$recy)^2)
r2_recy <- 1 - SSErecy/SSTrecy
r2_recy
```

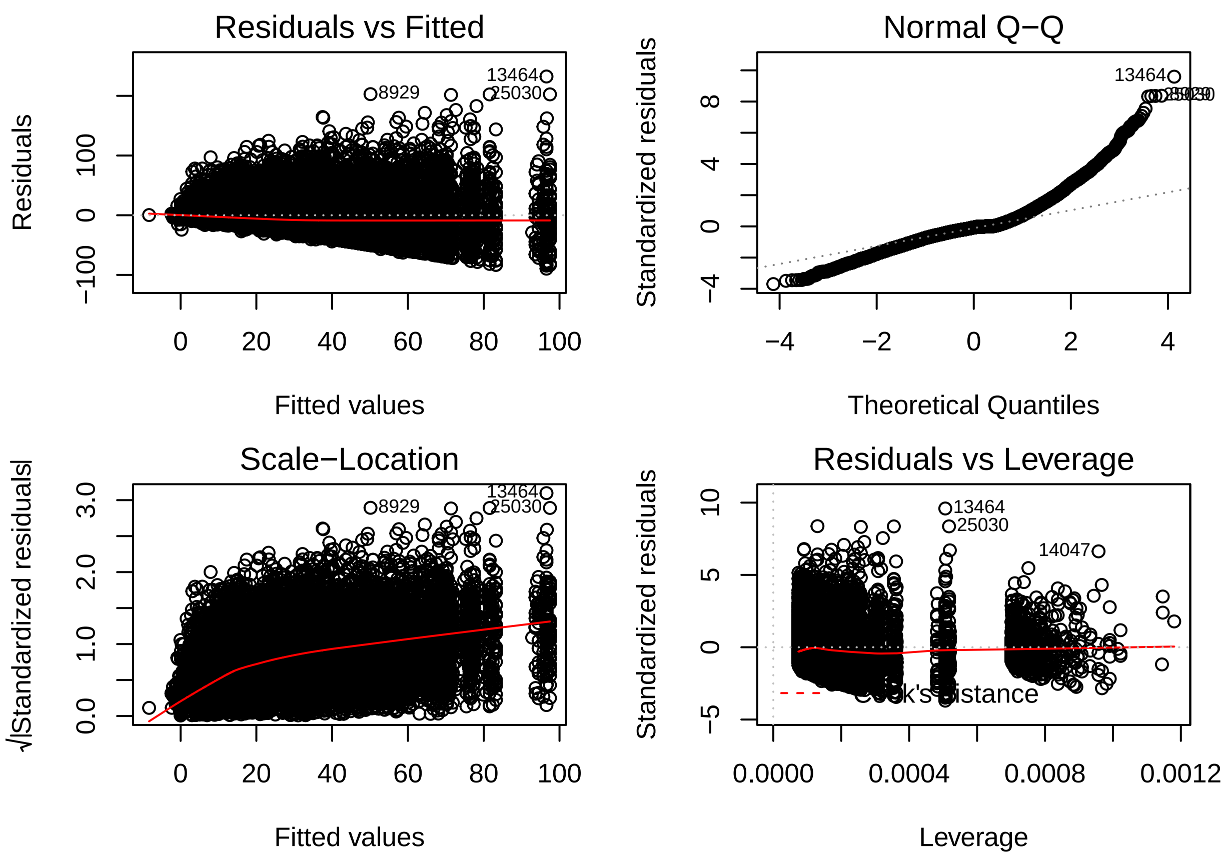
```
## [1] 0.4443393
```

```r
rmse_recy <- sqrt(SSErecy/nrow(TestRecy))
rmse_recy
```

```
## [1] 23.39669
```

Looking at the regression plots:

```r
par(mar = c(4, 4, 2, 2), mfrow = c(2, 2))
plot(linRegrecy2, which = c(1,2,3,5))
```



The residuals vs fitted appears to be okay for the model. The normal Q-Q looks okay, however, it may have some skewness to it. The scale-location does not seem to be ideal. The red line is not smooth, and there appears to be a gap in the data. The residuals vs leverage has some values that seem extreme.

The summary statistics are below.

```r
confint(linRegrecy2)
```

```
##                     2.5 %     97.5 %
## (Intercept)   -0.03566562  1.0214112
## avg_recy_plyr  0.98569940  1.0130606
## grass_1       -1.38484895 -0.2376387
## bad_weather_1 -2.96184739 -0.4194934
```

```r
coef(summary(linRegrecy2))
```

```
##                 Estimate  Std. Error    t value    Pr(>|t|)
## (Intercept)    0.4928728 0.269655529   1.827787 0.067592411
## avg_recy_plyr  0.9993800 0.006979718 143.183444 0.000000000
## grass_1       -0.8112438 0.292648152  -2.772079 0.005573697
## bad_weather_1 -1.6906704 0.648543049  -2.606875 0.009142216
```

```r
anova(linRegrecy2)
```

```
## Analysis of Variance Table
##
## Response: recy
##                  Df   Sum Sq  Mean Sq    F value    Pr(>F)
## avg_recy_plyr     1 12050728 12050728 20519.2219 < 2.2e-16 ***
## grass_1           1     4854     4854     8.2650  0.004045 **
## bad_weather_1     1     3991     3991     6.7958  0.009142 **
## Residuals     27480 16138722      587
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

I would say overall, the model is just okay for predicting. The average yards receiving historical for the player is really the best predictor according to this analysis

Receptions

```r
linRegrec <- lm(rec ~ height+ weight+cold_weather + hot_weather + home_team_1+ temp+ forty1 + vertical1
                   CIN + CLE + DAL + DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR + NY
                   NYJ + OAK + PHI + PIT +SD + SEA + STL + TB + TEN + WAS + avg_recy_plyr+avg_recy_pos
                  avg_recy_team + avg_rec_plyr +avg_rec_pos + avg_rec_team +avg_trg_plyr + avg_trg_pos
                  avg_trg_team + avg_rectd_plyr + avg_rectd_pos +avg_rectd_team+
                  avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
                  avg_rbra_plyr + avg_rbra_pos +avg_rbra_team +
                  avg_rbry_plyr + avg_rbry_pos +avg_rbry_team +
                  avg_fuml_plyr + avg_fuml_pos +avg_fuml_team +
                  avg_qbpy_plyr + avg_qbpy_pos +avg_qbpy_team +
                  avg_qbpa_plyr + avg_qbpa_pos +avg_qbpa_team+
                  avg_qbpc_plyr + avg_qbpc_pos +avg_qbpc_team +
                  avg_qbints_plyr + avg_qbints_pos +avg_qbints_team +
                  avg_qbtdp_plyr + avg_qbtdp_pos +avg_qbtdp_team + grass_1 + bad_weather_1, data = nfl_

summary(linRegrec)
```

```
##
## Call:
## lm(formula = rec ~ height + weight + cold_weather + hot_weather +
##     home_team_1 + temp + forty1 + vertical1 + shuttle1 + cone1 +
##     ARI + ATL + BAL + BUF + CAR + CHI + CIN + CLE + DAL + DEN +
##     DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR +
##     NYG + NYJ + OAK + PHI + PIT + SD + SEA + STL + TB + TEN +
##     WAS + avg_recy_plyr + avg_recy_pos + avg_recy_team + avg_rec_plyr +
##     avg_rec_pos + avg_rec_team + avg_trg_plyr + avg_trg_pos +
##     avg_trg_team + avg_rectd_plyr + avg_rectd_pos + avg_rectd_team +
##     avg_tdr_plyr + avg_tdr_pos + avg_tdr_team + avg_rbra_plyr +
```

```
##     avg_rbra_pos + avg_rbra_team + avg_rbry_plyr + avg_rbry_pos +
##     avg_rbry_team + avg_fuml_plyr + avg_fuml_pos + avg_fuml_team +
##     avg_qbpy_plyr + avg_qbpy_pos + avg_qbpy_team + avg_qbpa_plyr +
##     avg_qbpa_pos + avg_qbpa_team + avg_qbpc_plyr + avg_qbpc_pos +
##     avg_qbpc_team + avg_qbints_plyr + avg_qbints_pos + avg_qbints_team +
##     avg_qbtdp_plyr + avg_qbtdp_pos + avg_qbtdp_team + grass_1 +
##     bad_weather_1, data = nfl_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.4192 -0.9846 -0.0923  0.6720 12.9113
##
## Coefficients: (18 not defined because of singularities)
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.0938807  0.7039114   0.133 0.893901
## height          -0.0042403  0.0060734  -0.698 0.485069
## weight           0.0002323  0.0008104   0.287 0.774409
## cold_weather    -0.0220358  0.0366898  -0.601 0.548112
## hot_weather     -0.1970381  0.1449864  -1.359 0.174151
## home_team_1     -0.0254156  0.0191256  -1.329 0.183895
## temp             0.0017991  0.0010453   1.721 0.085234 .
## forty1           0.0017486  0.0964507   0.018 0.985536
## vertical1        0.0005648  0.0039398   0.143 0.886002
## shuttle1        -0.0271131  0.0832612  -0.326 0.744699
## cone1            0.0060101  0.0607552   0.099 0.921200
## ARI              0.1197501  0.0698525   1.714 0.086476 .
## ATL              0.1953800  0.0699112   2.795 0.005198 **
## BAL              0.2263529  0.0678648   3.335 0.000853 ***
## BUF              0.2195255  0.0706494   3.107 0.001890 **
## CAR              0.1062628  0.0691454   1.537 0.124350
## CHI              0.2570649  0.0706488   3.639 0.000274 ***
## CIN              0.2251935  0.0700479   3.215 0.001306 **
## CLE              0.3016798  0.0693510   4.350 1.36e-05 ***
## DAL              0.1397207  0.0707926   1.974 0.048427 *
## DEN              0.1536972  0.0681094   2.257 0.024037 *
## DET              0.1571431  0.0702766   2.236 0.025353 *
## GB               0.2185524  0.0685043   3.190 0.001422 **
## HOU              0.2088800  0.0695237   3.004 0.002662 **
## IND              0.1201685  0.0695019   1.729 0.083818 .
## JAC              0.2439920  0.0703792   3.467 0.000527 ***
## KC               0.2208369  0.0687027   3.214 0.001308 **
## MIA              0.1037127  0.0702386   1.477 0.139797
## MINN             0.1565144  0.0704092   2.223 0.026227 *
## NE               0.1415596  0.0691429   2.047 0.040631 *
## NOR              0.2432929  0.0694959   3.501 0.000464 ***
## NYG              0.2506883  0.0695777   3.603 0.000315 ***
## NYJ              0.1014262  0.0695915   1.457 0.145000
## OAK              0.2556766  0.0685380   3.730 0.000191 ***
## PHI              0.0569294  0.0697114   0.817 0.414137
## PIT              0.0941025  0.0688842   1.366 0.171917
## SD               0.2398785  0.0703241   3.411 0.000648 ***
## SEA              0.0139364  0.0678575   0.205 0.837278
## STL              0.1932653  0.0701693   2.754 0.005885 **
## TB               0.1944023  0.0698621   2.783 0.005394 **
```

```
## TEN              0.1973571  0.0689379   2.863 0.004201 **
## WAS              0.1976704  0.0687225   2.876 0.004025 **
## avg_recy_plyr    0.0015752  0.0024978   0.631 0.528282
## avg_recy_pos     0.0086644  0.3211383   0.027 0.978476
## avg_recy_team           NA         NA      NA       NA
## avg_rec_plyr     0.9902182  0.0385370  25.695  < 2e-16 ***
## avg_rec_pos      0.0769220  0.7553548   0.102 0.918888
## avg_rec_team            NA         NA      NA       NA
## avg_trg_plyr    -0.0064921  0.0282613  -0.230 0.818313
## avg_trg_pos     -0.1284888  3.6335359  -0.035 0.971791
## avg_trg_team            NA         NA      NA       NA
## avg_rectd_plyr  -0.0009410  0.1335739  -0.007 0.994379
## avg_rectd_pos    0.2897527 16.6469179   0.017 0.986113
## avg_rectd_team          NA         NA      NA       NA
## avg_tdr_plyr     0.0195821  0.1647512   0.119 0.905388
## avg_tdr_pos      0.0477189  8.1491593   0.006 0.995328
## avg_tdr_team            NA         NA      NA       NA
## avg_rbra_plyr   -0.0062248  0.0159222  -0.391 0.695837
## avg_rbra_pos     0.0165374  0.5810623   0.028 0.977295
## avg_rbra_team           NA         NA      NA       NA
## avg_rbry_plyr    0.0015785  0.0036987   0.427 0.669546
## avg_rbry_pos    -0.0042079  0.0530064  -0.079 0.936727
## avg_rbry_team           NA         NA      NA       NA
## avg_fuml_plyr   -0.0096984  0.1971976  -0.049 0.960775
## avg_fuml_pos     0.1110329  2.3276112   0.048 0.961954
## avg_fuml_team           NA         NA      NA       NA
## avg_qbpy_plyr    0.0020097  0.0031234   0.643 0.519948
## avg_qbpy_pos            NA         NA      NA       NA
## avg_qbpy_team           NA         NA      NA       NA
## avg_qbpa_plyr    0.0044174  0.0243174   0.182 0.855854
## avg_qbpa_pos            NA         NA      NA       NA
## avg_qbpa_team           NA         NA      NA       NA
## avg_qbpc_plyr   -0.0258841  0.0439258  -0.589 0.555684
## avg_qbpc_pos            NA         NA      NA       NA
## avg_qbpc_team           NA         NA      NA       NA
## avg_qbints_plyr -0.0655407  0.1406670  -0.466 0.641270
## avg_qbints_pos          NA         NA      NA       NA
## avg_qbints_team         NA         NA      NA       NA
## avg_qbtdp_plyr  -0.0451540  0.1540147  -0.293 0.769386
## avg_qbtdp_pos           NA         NA      NA       NA
## avg_qbtdp_team          NA         NA      NA       NA
## grass_1         -0.0478848  0.0197174  -2.429 0.015164 *
## bad_weather_1   -0.0976921  0.0392384  -2.490 0.012789 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.7 on 39189 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.4514, Adjusted R-squared:  0.4505
## F-statistic: 503.8 on 64 and 39189 DF,  p-value: < 2.2e-16
```

Second run at Receptions

```r
linRegrec2 <- linRegrec2 <- lm(rec ~ temp + CHI+
                   CLE +
                   avg_rec_plyr+ grass_1+
                   bad_weather_1, data = TrainRecy)

summary(linRegrec2)
```

```
##
## Call:
## lm(formula = rec ~ temp + CHI + CLE + avg_rec_plyr + grass_1 +
##     bad_weather_1, data = TrainRecy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.1523 -0.9887 -0.0730  0.6889 12.8809
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -0.0979026  0.0425781  -2.299 0.021492 *
## temp           0.0024694  0.0006762   3.652 0.000261 ***
## CHI            0.1497555  0.0618786   2.420 0.015521 *
## CLE            0.1329733  0.0604090   2.201 0.027729 *
## avg_rec_plyr   0.9935729  0.0066937 148.434  < 2e-16 ***
## grass_1       -0.0558199  0.0207806  -2.686 0.007232 **
## bad_weather_1 -0.1132370  0.0467311  -2.423 0.015392 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.709 on 27477 degrees of freedom
## Multiple R-squared:  0.4458, Adjusted R-squared:  0.4457
## F-statistic:  3684 on 6 and 27477 DF,  p-value: < 2.2e-16
```

We had a modest r-square improvement. The R2 is not very strong (0.4457), and we achieved approximately the same value when we cut the variables down to just 6. Much simpler model with similar results

Testing the data, we see that the training set and the test set are similar. The model seems to hold up through testing

```r
RecPredicted <- predict(linRegrec2, newdata = TestRecy)

SSErec <- sum((RecPredicted - TestRecy$rec)^2)
SSTrec <- sum((mean(nfl_data$rec)-TestRecy$rec)^2)
r2_rec <- 1 - SSErec/SSTrec
r2_rec
```

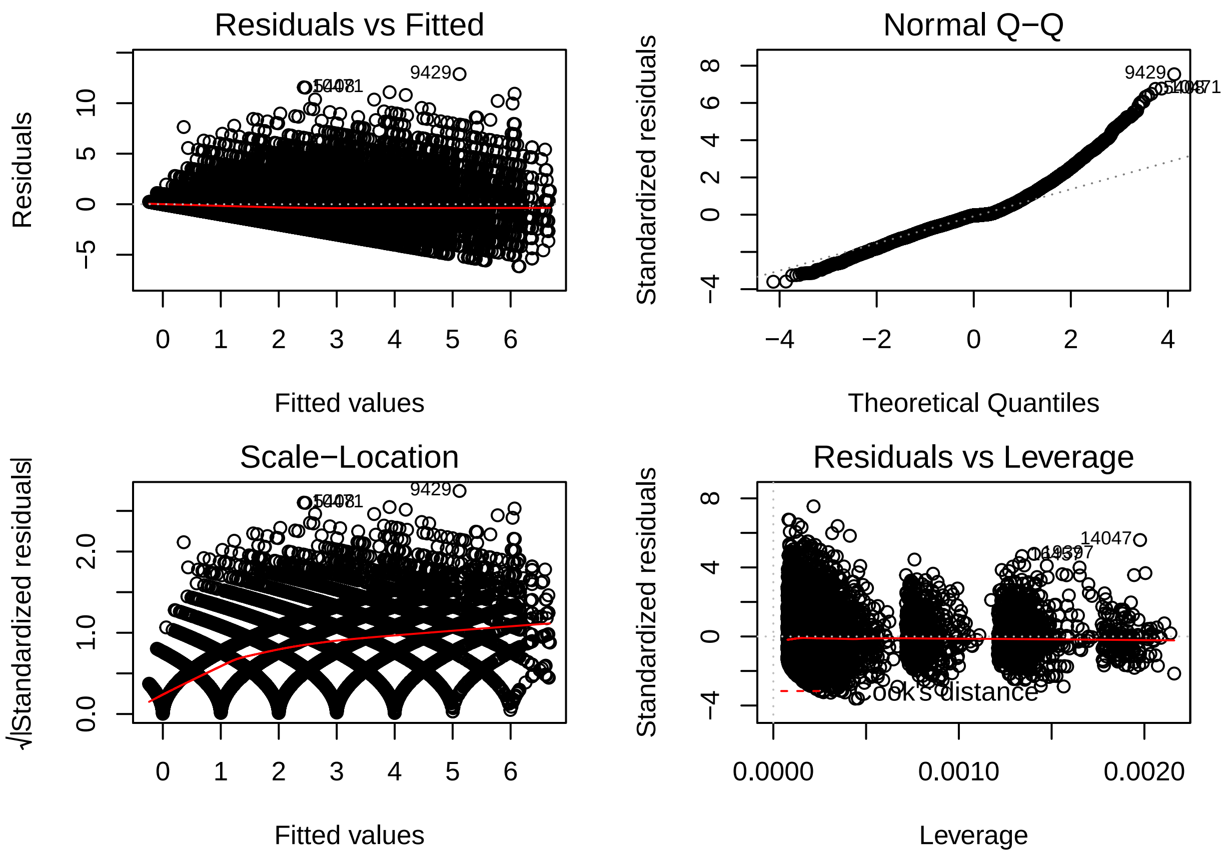
```
## [1] 0.4613523
```

```r
rmse_rec <- sqrt(SSErec/nrow(TestRecy))
rmse_rec
```

```
## [1] 1.679946
```

Looking at the regression plots:

```r
par(mar = c(4, 4, 2, 2), mfrow = c(2, 2))
plot(linRegrec2, which = c(1,2,3,5))
```



The charts show very similarly to what we saw above for receiving yards.

The summary statistics are below:

```
confint(linRegrec2)
```

```
##                       2.5 %       97.5 %
## (Intercept)    -0.181357900 -0.014447355
## temp            0.001143982  0.003794792
## CHI             0.028470358  0.271040729
## CLE             0.014568632  0.251377975
## avg_rec_plyr    0.980452871  1.006692890
## grass_1        -0.096550998 -0.015088862
## bad_weather_1  -0.204832290 -0.021641738
```

```
coef(summary(linRegrec2))
```

```
##                    Estimate   Std. Error    t value     Pr(>|t|)
## (Intercept)    -0.097902627 0.0425781273  -2.299364 0.0214917221
## temp            0.002469387 0.0006762097   3.651806 0.0002608857
## CHI             0.149755543 0.0618786079   2.420150 0.0155205279
## CLE             0.132973303 0.0604089954   2.201217 0.0277289544
## avg_rec_plyr    0.993572881 0.0066937107 148.433796 0.0000000000
## grass_1        -0.055819930 0.0207806235  -2.686153 0.0072323467
## bad_weather_1  -0.113237014 0.0467310839  -2.423163 0.0153924561
```

```
anova(linRegrec2)
```

```
## Analysis of Variance Table
```

```
##
## Response: rec
##                  Df Sum Sq Mean Sq    F value    Pr(>F)
## temp              1     99      99    33.8855  5.91e-09 ***
## CHI               1      2       2     0.8325  0.361565
## CLE               1      7       7     2.4935  0.114332
## avg_rec_plyr      1  64378   64378 22051.6165 < 2.2e-16 ***
## grass_1           1     22      22     7.5773  0.005915 **
## bad_weather_1     1     17      17     5.8717  0.015392 *
## Residuals     27477  80217       3
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Like the model above, linear regression may not be the best predictor for this statistic

Targets

```r
linRegtrg <- lm(trg ~ height+ weight+cold_weather + hot_weather + home_team_1+ temp+forty1 + vertical1
                  CIN + CLE + DAL + DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR + NYG
                  NYJ + OAK + PHI + PIT +SD + SEA + STL + TB + TEN + WAS +avg_recy_plyr+avg_recy_pos +
                  avg_recy_team + avg_rec_plyr +avg_rec_pos + avg_rec_team +avg_trg_plyr + avg_trg_pos
                  avg_trg_team + avg_rectd_plyr + avg_rectd_pos +avg_rectd_team+
                  avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
                  avg_rbra_plyr + avg_rbra_pos +avg_rbra_team +
                  avg_rbry_plyr + avg_rbry_pos +avg_rbry_team +
                  avg_fuml_plyr + avg_fuml_pos +avg_fuml_team +
                  avg_qbpy_plyr + avg_qbpy_pos +avg_qbpy_team +
                  avg_qbpa_plyr + avg_qbpa_pos +avg_qbpa_team+
                  avg_qbpc_plyr + avg_qbpc_pos +avg_qbpc_team +
                  avg_qbints_plyr + avg_qbints_pos +avg_qbints_team +
                  avg_qbtdp_plyr + avg_qbtdp_pos +avg_qbtdp_team + grass_1 + bad_weather_1 , data = nfl

summary(linRegtrg)
```

```
##
## Call:
## lm(formula = trg ~ height + weight + cold_weather + hot_weather +
##     home_team_1 + temp + forty1 + vertical1 + shuttle1 + cone1 +
##     ARI + ATL + BAL + BUF + CAR + CHI + CIN + CLE + DAL + DEN +
##     DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR +
##     NYG + NYJ + OAK + PHI + PIT + SD + SEA + STL + TB + TEN +
##     WAS + avg_recy_plyr + avg_recy_pos + avg_recy_team + avg_rec_plyr +
##     avg_rec_pos + avg_rec_team + avg_trg_plyr + avg_trg_pos +
##     avg_trg_team + avg_rectd_plyr + avg_rectd_pos + avg_rectd_team +
##     avg_tdr_plyr + avg_tdr_pos + avg_tdr_team + avg_rbra_plyr +
##     avg_rbra_pos + avg_rbra_team + avg_rbry_plyr + avg_rbry_pos +
##     avg_rbry_team + avg_fuml_plyr + avg_fuml_pos + avg_fuml_team +
##     avg_qbpy_plyr + avg_qbpy_pos + avg_qbpy_team + avg_qbpa_plyr +
##     avg_qbpa_pos + avg_qbpa_team + avg_qbpc_plyr + avg_qbpc_pos +
##     avg_qbpc_team + avg_qbints_plyr + avg_qbints_pos + avg_qbints_team +
##     avg_qbtdp_plyr + avg_qbtdp_pos + avg_qbtdp_team + grass_1 +
##     bad_weather_1, data = nfl_data)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.8392 -1.2671 -0.1191  0.8976 15.9457
##
## Coefficients: (18 not defined because of singularities)
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.2805420  0.9609969   0.292 0.770343
## height          -0.0050085  0.0082916  -0.604 0.545816
## weight           0.0001882  0.0011063   0.170 0.864933
## cold_weather    -0.0458794  0.0500899  -0.916 0.359703
## hot_weather     -0.2172190  0.1979389  -1.097 0.272472
## home_team_1     -0.0972821  0.0261107  -3.726 0.000195 ***
## temp            -0.0007050  0.0014271  -0.494 0.621291
## forty1          -0.0071155  0.1316769  -0.054 0.956905
## vertical1        0.0006616  0.0053787   0.123 0.902110
## shuttle1        -0.0409339  0.1136702  -0.360 0.718766
## cone1            0.0066546  0.0829445   0.080 0.936055
## ARI              0.2403120  0.0953644   2.520 0.011742 *
## ATL              0.2866087  0.0954444   3.003 0.002676 **
## BAL              0.3775828  0.0926507   4.075 4.60e-05 ***
## BUF              0.3935459  0.0964523   4.080 4.51e-05 ***
## CAR              0.2493833  0.0943990   2.642 0.008250 **
## CHI              0.3678562  0.0964514   3.814 0.000137 ***
## CIN              0.3416093  0.0956311   3.572 0.000354 ***
## CLE              0.5036674  0.0946796   5.320 1.05e-07 ***
## DAL              0.2107326  0.0966478   2.180 0.029232 *
## DEN              0.2696729  0.0929846   2.900 0.003731 **
## DET              0.2720134  0.0959433   2.835 0.004583 **
## GB               0.3200362  0.0935237   3.422 0.000622 ***
## HOU              0.3616955  0.0949154   3.811 0.000139 ***
## IND              0.2467854  0.0948857   2.601 0.009302 **
## JAC              0.4402984  0.0960834   4.582 4.61e-06 ***
## KC               0.3218716  0.0937946   3.432 0.000601 ***
## MIA              0.2334368  0.0958915   2.434 0.014922 *
## MINN             0.2619964  0.0961244   2.726 0.006421 **
## NE               0.2339584  0.0943956   2.478 0.013198 *
## NOR              0.3255428  0.0948775   3.431 0.000602 ***
## NYG              0.4347323  0.0949892   4.577 4.74e-06 ***
## NYJ              0.2333530  0.0950079   2.456 0.014048 *
## OAK              0.4152517  0.0935697   4.438 9.11e-06 ***
## PHI              0.1568084  0.0951717   1.648 0.099435 .
## PIT              0.1139187  0.0940424   1.211 0.225767
## SD               0.3244898  0.0960082   3.380 0.000726 ***
## SEA              0.0376568  0.0926406   0.406 0.684390
## STL              0.3736498  0.0957968   3.900 9.62e-05 ***
## TB               0.3788240  0.0953775   3.972 7.14e-05 ***
## TEN              0.3174885  0.0941157   3.373 0.000743 ***
## WAS              0.2768352  0.0938216   2.951 0.003173 **
## avg_recy_plyr    0.0030372  0.0034101   0.891 0.373123
## avg_recy_pos     0.0318712  0.4384258   0.073 0.942049
## avg_recy_team           NA         NA      NA       NA
## avg_rec_plyr     0.0119431  0.0526116   0.227 0.820421
## avg_rec_pos      0.1448771  1.0312286   0.140 0.888274
## avg_rec_team            NA         NA      NA       NA
```

```
## avg_trg_plyr     0.9667014  0.0385830  25.055  < 2e-16 ***
## avg_trg_pos     -0.4069845  4.9605914  -0.082 0.934612
## avg_trg_team            NA         NA      NA       NA
## avg_rectd_plyr   0.0307513  0.1823584   0.169 0.866088
## avg_rectd_pos    1.6199949 22.7267761   0.071 0.943174
## avg_rectd_team          NA         NA      NA       NA
## avg_tdr_plyr     0.0388035  0.2249223   0.173 0.863030
## avg_tdr_pos     -0.5249072 11.1254300  -0.047 0.962369
## avg_tdr_team            NA         NA      NA       NA
## avg_rbra_plyr   -0.0087116  0.0217374  -0.401 0.688595
## avg_rbra_pos     0.0557440  0.7932803   0.070 0.943979
## avg_rbra_team           NA         NA      NA       NA
## avg_rbry_plyr    0.0020798  0.0050495   0.412 0.680429
## avg_rbry_pos    -0.0052586  0.0723656  -0.073 0.942071
## avg_rbry_team           NA         NA      NA       NA
## avg_fuml_plyr   -0.0109521  0.2692189  -0.041 0.967551
## avg_fuml_pos     0.1504232  3.1777113   0.047 0.962245
## avg_fuml_team           NA         NA      NA       NA
## avg_qbpy_plyr    0.0033452  0.0042642   0.784 0.432757
## avg_qbpy_pos            NA         NA      NA       NA
## avg_qbpy_team           NA         NA      NA       NA
## avg_qbpa_plyr   -0.0092676  0.0331987  -0.279 0.780127
## avg_qbpa_pos            NA         NA      NA       NA
## avg_qbpa_team           NA         NA      NA       NA
## avg_qbpc_plyr   -0.0170934  0.0599685  -0.285 0.775616
## avg_qbpc_pos            NA         NA      NA       NA
## avg_qbpc_team           NA         NA      NA       NA
## avg_qbints_plyr -0.0865833  0.1920420  -0.451 0.652096
## avg_qbints_pos          NA         NA      NA       NA
## avg_qbints_team         NA         NA      NA       NA
## avg_qbtdp_plyr  -0.0799650  0.2102647  -0.380 0.703720
## avg_qbtdp_pos           NA         NA      NA       NA
## avg_qbtdp_team          NA         NA      NA       NA
## grass_1         -0.0415842  0.0269187  -1.545 0.122401
## bad_weather_1    0.0103342  0.0535692   0.193 0.847029
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.321 on 39189 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.5394, Adjusted R-squared:  0.5387
## F-statistic: 717.1 on 64 and 39189 DF,  p-value: < 2.2e-16
```

Second run at targets

```
linRegtrg2 <- lm(trg ~ home_team_1 + BAL + BUF + CHI + CIN + CLE +
                   HOU + JAC + NOR + NYG+
                   OAK + STL + TB + avg_trg_plyr,
                 data = TrainRecy)

summary(linRegtrg2)
```

```
##
## Call:
```

```
## lm(formula = trg ~ home_team_1 + BAL + BUF + CHI + CIN + CLE +
##     HOU + JAC + NOR + NYG + OAK + STL + TB + avg_trg_plyr, data = TrainRecy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.3536 -1.2790 -0.1288  0.8969 15.9835
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.001327   0.028462  -0.047  0.96281
## home_team_1  -0.123797   0.029554  -4.189 2.81e-05 ***
## BAL           0.111353   0.079915   1.393  0.16351
## BUF           0.151280   0.082986   1.823  0.06832 .
## CHI           0.130088   0.084925   1.532  0.12559
## CIN           0.161660   0.082675   1.955  0.05055 .
## CLE           0.238764   0.082756   2.885  0.00392 **
## HOU           0.130947   0.081694   1.603  0.10897
## JAC           0.232303   0.084522   2.748  0.00599 **
## NOR           0.076147   0.077904   0.977  0.32836
## NYG           0.188197   0.081073   2.321  0.02028 *
## OAK           0.170422   0.079925   2.132  0.03299 *
## STL           0.150210   0.084192   1.784  0.07441 .
## TB            0.075675   0.083273   0.909  0.36348
## avg_trg_plyr  0.996502   0.005625 177.168  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.337 on 27469 degrees of freedom
## Multiple R-squared:  0.5339, Adjusted R-squared:  0.5337
## F-statistic:  2248 on 14 and 27469 DF,  p-value: < 2.2e-16
```

Modest gains in second run's R2. It is a much more simple model, and has a little better descriptive stats.

Testing the data, we see that the training set and the test set are similar. The model seems to hold up through testing

```
TrgPredicted <- predict(linRegtrg2, newdata = TestRecy)

SSEtrg <- sum((TrgPredicted - TestRecy$trg)^2)
SSTtrg <- sum((mean(nfl_data$trg)-TestRecy$trg)^2)
r2_trg <- 1 - SSEtrg/SSTtrg
r2_trg
```

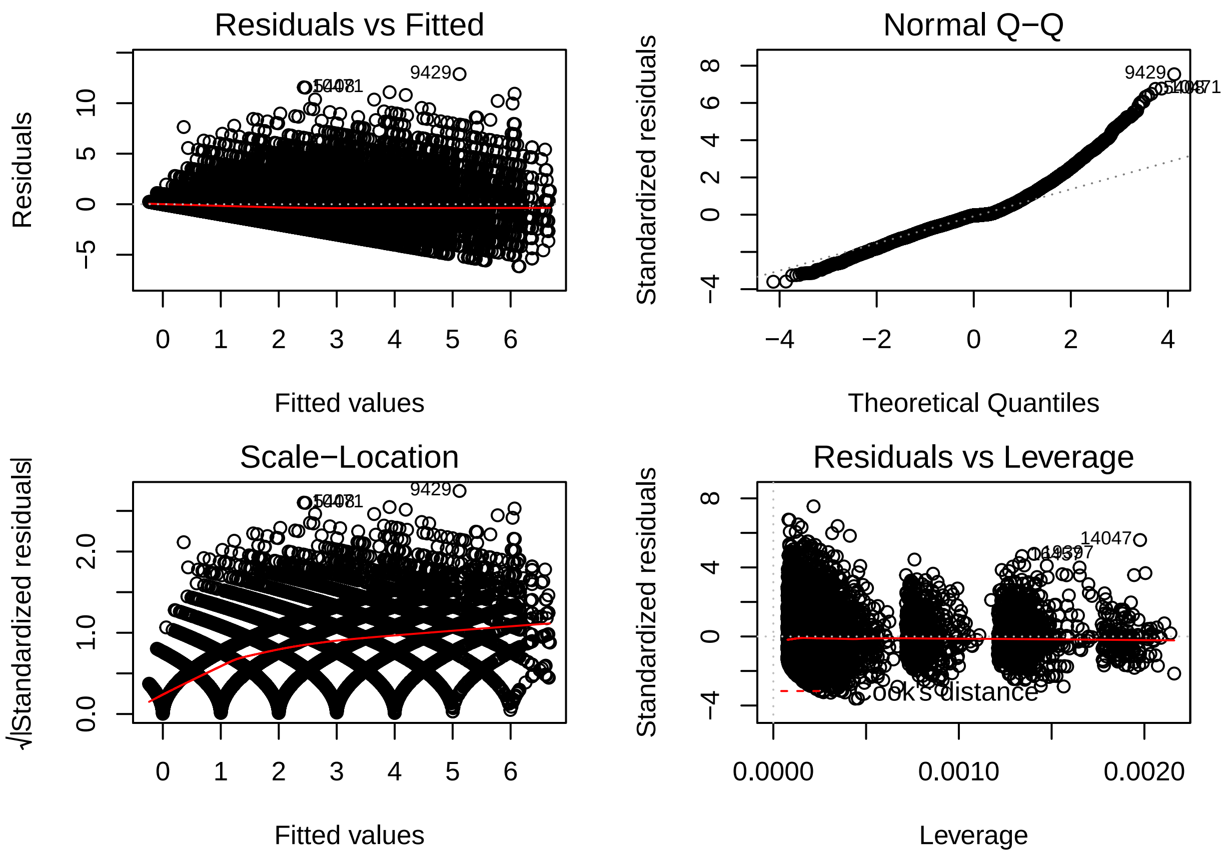
```
## [1] 0.550581
```

```
rmse_trg <- sqrt(SSEtrg/nrow(TestRecy))
rmse_trg
```

```
## [1] 2.282097
```

The regression plots for targets are below:

```
par(mar = c(4, 4, 2, 2), mfrow = c(2, 2))
plot(linRegrec2, which = c(1:3,5))
```



Here are additional summary statistics:

```r
confint(linRegtrg2)
```

```
##                      2.5 %      97.5 %
## (Intercept)  -0.0571143572  0.05445980
## home_team_1  -0.1817242415 -0.06586998
## BAL          -0.0452842381  0.26799077
## BUF          -0.0113759840  0.31393594
## CHI          -0.0363699880  0.29654580
## CIN          -0.0003872561  0.32370822
## CLE           0.0765579240  0.40097058
## HOU          -0.0291765861  0.29107082
## JAC           0.0666356429  0.39797050
## NOR          -0.0765490931  0.22884381
## NYG           0.0292888968  0.34710500
## OAK           0.0137646891  0.32707940
## STL          -0.0148097445  0.31523023
## TB           -0.0875441047  0.23889472
## avg_trg_plyr  0.9854776112  1.00752668
```

```r
coef(summary(linRegtrg2))
```

```
##                 Estimate  Std. Error     t value     Pr(>|t|)
## (Intercept)  -0.00132728 0.028462064 -0.04663329 9.628058e-01
## home_team_1  -0.12379711 0.029553899 -4.18885889 2.812351e-05
## BAL           0.11135327 0.079915042  1.39339560 1.635115e-01
## BUF           0.15127998 0.082985603  1.82296652 6.831933e-02
```

```
## CHI           0.13008790 0.084925314   1.53179185 1.255853e-01
## CIN           0.16166048 0.082675294   1.95536629 5.054998e-02
## CLE           0.23876425 0.082756205   2.88515227 3.915287e-03
## HOU           0.13094712 0.081693669   1.60290412 1.089673e-01
## JAC           0.23230307 0.084522027   2.74843233 5.992008e-03
## NOR           0.07614736 0.077904351   0.97744679 3.283566e-01
## NYG           0.18819695 0.081073453   2.32131406 2.027718e-02
## OAK           0.17042204 0.079925168   2.13227006 3.299350e-02
## STL           0.15021024 0.084191709   1.78414533 7.441109e-02
## TB            0.07567531 0.083273072   0.90876087 3.634843e-01
## avg_trg_plyr  0.99650214 0.005624617 177.16798695 0.000000e+00
```

```r
anova(linRegtrg2)
```

```
## Analysis of Variance Table
##
## Response: trg
##                  Df Sum Sq Mean Sq    F value    Pr(>F)
## home_team_1       1    206     206    37.6532 8.566e-10 ***
## BAL               1      0       0     0.0140 0.9059734
## BUF               1      5       5     0.9332 0.3340331
## CHI               1      0       0     0.0138 0.9065536
## CIN               1      1       1     0.1507 0.6978352
## CLE               1      1       1     0.1065 0.7441778
## HOU               1      0       0     0.0836 0.7725401
## JAC               1      7       7     1.2099 0.2713631
## NOR               1     74      74    13.5635 0.0002311 ***
## NYG               1     17      17     3.0571 0.0803957 .
## OAK               1     24      24     4.3441 0.0371470 *
## STL               1     64      64    11.6353 0.0006480 ***
## TB                1     33      33     6.0404 0.0139881 *
## avg_trg_plyr      1 171416  171416 31388.4956 < 2.2e-16 ***
## Residuals     27469 150011       5
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This one was an improvement over the previous models. It still has its problems, and may need some refinement or other variables to improve the predictions.

Receiving TD's

```r
linRegRecTD <- lm(trg ~ height+ weight+cold_weather + hot_weather + home_team_1+ temp+ forty1 + vertica
                  CIN + CLE + DAL + DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR + NYG
                  NYJ + OAK + PHI + PIT +SD + SEA + STL + TB + TEN + WAS +avg_recy_plyr+avg_recy_pos +
                  avg_recy_team + avg_rec_plyr +avg_rec_pos + avg_rec_team +avg_trg_plyr + avg_trg_pos
                  avg_trg_team + avg_rectd_plyr + avg_rectd_pos +avg_rectd_team+
                  avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
                  avg_rbra_plyr + avg_rbra_pos +avg_rbra_team +
                  avg_rbry_plyr + avg_rbry_pos +avg_rbry_team +
                  avg_fuml_plyr + avg_fuml_pos +avg_fuml_team +
                  avg_qbpy_plyr + avg_qbpy_pos +avg_qbpy_team +
                  avg_qbpa_plyr + avg_qbpa_pos +avg_qbpa_team+
                  avg_qbpc_plyr + avg_qbpc_pos +avg_qbpc_team +
                  avg_qbints_plyr + avg_qbints_pos +avg_qbints_team +
```

```
                  avg_qbtdp_plyr + avg_qbtdp_pos +avg_qbtdp_team + grass_1 + bad_weather_1 , data = nfl_

summary(linRegRecTD)
```

```
##
## Call:
## lm(formula = trg ~ height + weight + cold_weather + hot_weather +
##     home_team_1 + temp + forty1 + vertical1 + shuttle1 + cone1 +
##     ARI + ATL + BAL + BUF + CAR + CHI + CIN + CLE + DAL + DEN +
##     DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR +
##     NYG + NYJ + OAK + PHI + PIT + SD + SEA + STL + TB + TEN +
##     WAS + avg_recy_plyr + avg_recy_pos + avg_recy_team + avg_rec_plyr +
##     avg_rec_pos + avg_rec_team + avg_trg_plyr + avg_trg_pos +
##     avg_trg_team + avg_rectd_plyr + avg_rectd_pos + avg_rectd_team +
##     avg_tdr_plyr + avg_tdr_pos + avg_tdr_team + avg_rbra_plyr +
##     avg_rbra_pos + avg_rbra_team + avg_rbry_plyr + avg_rbry_pos +
##     avg_rbry_team + avg_fuml_plyr + avg_fuml_pos + avg_fuml_team +
##     avg_qbpy_plyr + avg_qbpy_pos + avg_qbpy_team + avg_qbpa_plyr +
##     avg_qbpa_pos + avg_qbpa_team + avg_qbpc_plyr + avg_qbpc_pos +
##     avg_qbpc_team + avg_qbints_plyr + avg_qbints_pos + avg_qbints_team +
##     avg_qbtdp_plyr + avg_qbtdp_pos + avg_qbtdp_team + grass_1 +
##     bad_weather_1, data = nfl_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.8392 -1.2671 -0.1191  0.8976 15.9457
##
## Coefficients: (18 not defined because of singularities)
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.2805420  0.9609969   0.292 0.770343
## height          -0.0050085  0.0082916  -0.604 0.545816
## weight           0.0001882  0.0011063   0.170 0.864933
## cold_weather    -0.0458794  0.0500899  -0.916 0.359703
## hot_weather     -0.2172190  0.1979389  -1.097 0.272472
## home_team_1     -0.0972821  0.0261107  -3.726 0.000195 ***
## temp            -0.0007050  0.0014271  -0.494 0.621291
## forty1          -0.0071155  0.1316769  -0.054 0.956905
## vertical1        0.0006616  0.0053787   0.123 0.902110
## shuttle1        -0.0409339  0.1136702  -0.360 0.718766
## cone1            0.0066546  0.0829445   0.080 0.936055
## ARI              0.2403120  0.0953644   2.520 0.011742 *
## ATL              0.2866087  0.0954444   3.003 0.002676 **
## BAL              0.3775828  0.0926507   4.075 4.60e-05 ***
## BUF              0.3935459  0.0964523   4.080 4.51e-05 ***
## CAR              0.2493833  0.0943990   2.642 0.008250 **
## CHI              0.3678562  0.0964514   3.814 0.000137 ***
## CIN              0.3416093  0.0956311   3.572 0.000354 ***
## CLE              0.5036674  0.0946796   5.320 1.05e-07 ***
## DAL              0.2107326  0.0966478   2.180 0.029232 *
## DEN              0.2696729  0.0929846   2.900 0.003731 **
## DET              0.2720134  0.0959433   2.835 0.004583 **
## GB               0.3200362  0.0935237   3.422 0.000622 ***
## HOU              0.3616955  0.0949154   3.811 0.000139 ***
## IND              0.2467854  0.0948857   2.601 0.009302 **
```

```
## JAC             0.4402984  0.0960834   4.582 4.61e-06 ***
## KC              0.3218716  0.0937946   3.432 0.000601 ***
## MIA             0.2334368  0.0958915   2.434 0.014922 *
## MINN            0.2619964  0.0961244   2.726 0.006421 **
## NE              0.2339584  0.0943956   2.478 0.013198 *
## NOR             0.3255428  0.0948775   3.431 0.000602 ***
## NYG             0.4347323  0.0949892   4.577 4.74e-06 ***
## NYJ             0.2333530  0.0950079   2.456 0.014048 *
## OAK             0.4152517  0.0935697   4.438 9.11e-06 ***
## PHI             0.1568084  0.0951717   1.648 0.099435 .
## PIT             0.1139187  0.0940424   1.211 0.225767
## SD              0.3244898  0.0960082   3.380 0.000726 ***
## SEA             0.0376568  0.0926406   0.406 0.684390
## STL             0.3736498  0.0957968   3.900 9.62e-05 ***
## TB              0.3788240  0.0953775   3.972 7.14e-05 ***
## TEN             0.3174885  0.0941157   3.373 0.000743 ***
## WAS             0.2768352  0.0938216   2.951 0.003173 **
## avg_recy_plyr   0.0030372  0.0034101   0.891 0.373123
## avg_recy_pos    0.0318712  0.4384258   0.073 0.942049
## avg_recy_team          NA         NA      NA       NA
## avg_rec_plyr    0.0119431  0.0526116   0.227 0.820421
## avg_rec_pos     0.1448771  1.0312286   0.140 0.888274
## avg_rec_team           NA         NA      NA       NA
## avg_trg_plyr    0.9667014  0.0385830  25.055  < 2e-16 ***
## avg_trg_pos    -0.4069845  4.9605914  -0.082 0.934612
## avg_trg_team           NA         NA      NA       NA
## avg_rectd_plyr  0.0307513  0.1823584   0.169 0.866088
## avg_rectd_pos   1.6199949 22.7267761   0.071 0.943174
## avg_rectd_team         NA         NA      NA       NA
## avg_tdr_plyr    0.0388035  0.2249223   0.173 0.863030
## avg_tdr_pos    -0.5249072 11.1254300  -0.047 0.962369
## avg_tdr_team           NA         NA      NA       NA
## avg_rbra_plyr  -0.0087116  0.0217374  -0.401 0.688595
## avg_rbra_pos    0.0557440  0.7932803   0.070 0.943979
## avg_rbra_team          NA         NA      NA       NA
## avg_rbry_plyr   0.0020798  0.0050495   0.412 0.680429
## avg_rbry_pos   -0.0052586  0.0723656  -0.073 0.942071
## avg_rbry_team          NA         NA      NA       NA
## avg_fuml_plyr  -0.0109521  0.2692189  -0.041 0.967551
## avg_fuml_pos    0.1504232  3.1777113   0.047 0.962245
## avg_fuml_team          NA         NA      NA       NA
## avg_qbpy_plyr   0.0033452  0.0042642   0.784 0.432757
## avg_qbpy_pos           NA         NA      NA       NA
## avg_qbpy_team          NA         NA      NA       NA
## avg_qbpa_plyr  -0.0092676  0.0331987  -0.279 0.780127
## avg_qbpa_pos           NA         NA      NA       NA
## avg_qbpa_team          NA         NA      NA       NA
## avg_qbpc_plyr  -0.0170934  0.0599685  -0.285 0.775616
## avg_qbpc_pos           NA         NA      NA       NA
## avg_qbpc_team          NA         NA      NA       NA
## avg_qbints_plyr -0.0865833  0.1920420  -0.451 0.652096
## avg_qbints_pos         NA         NA      NA       NA
## avg_qbints_team        NA         NA      NA       NA
## avg_qbtdp_plyr -0.0799650  0.2102647  -0.380 0.703720
```

```
## avg_qbtdp_pos            NA          NA      NA       NA
## avg_qbtdp_team           NA          NA      NA       NA
## grass_1         -0.0415842  0.0269187  -1.545 0.122401
## bad_weather_1    0.0103342  0.0535692   0.193 0.847029
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.321 on 39189 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.5394, Adjusted R-squared:  0.5387
## F-statistic: 717.1 on 64 and 39189 DF,  p-value: < 2.2e-16
```

Second Run

```
linRegRecTD2 <- lm(tdrec ~weight+home_team_1+ ATL+ DAL + DEN + GB + NE + NOR +
                  avg_recy_plyr+ avg_rec_plyr, data = TrainRecy)

summary(linRegRecTD2)
```

```
##
## Call:
## lm(formula = tdrec ~ weight + home_team_1 + ATL + DAL + DEN +
##     GB + NE + NOR + avg_recy_plyr + avg_rec_plyr, data = TrainRecy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.6950 -0.1781 -0.0687 -0.0017  3.6889
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.302e-01  2.147e-02 -10.721  < 2e-16 ***
## weight         9.669e-04  9.438e-05  10.245  < 2e-16 ***
## home_team_1    1.139e-02  4.802e-03   2.371  0.01774 *
## ATL            3.161e-02  1.330e-02   2.377  0.01745 *
## DAL            3.800e-02  1.340e-02   2.835  0.00459 **
## DEN            2.694e-02  1.293e-02   2.084  0.03719 *
## GB             5.758e-02  1.264e-02   4.557 5.21e-06 ***
## NE             3.846e-02  1.275e-02   3.017  0.00256 **
## NOR            5.901e-02  1.260e-02   4.683 2.84e-06 ***
## avg_recy_plyr  8.546e-03  3.794e-04  22.523  < 2e-16 ***
## avg_rec_plyr  -2.518e-02  5.141e-03  -4.898 9.76e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3794 on 27473 degrees of freedom
## Multiple R-squared:  0.1237, Adjusted R-squared:  0.1234
## F-statistic: 387.8 on 10 and 27473 DF,  p-value: < 2.2e-16
```

The R2 is worsened in the second model, and the R2 in general is fairly low.

Testing the data, we see that the training set and the test set are similar. The model seems to hold up through testing

```
RectdPredicted <- predict(linRegRecTD2, newdata = TestRecy)
```

```r
SSErectd <- sum((RectdPredicted - TestRecy$tdrec)^2)
SSTrectd <- sum((mean(nfl_data$tdrec)-TestRecy$tdrec)^2)
r2_rectd <- 1 - SSErectd/SSTrectd
r2_rectd
```

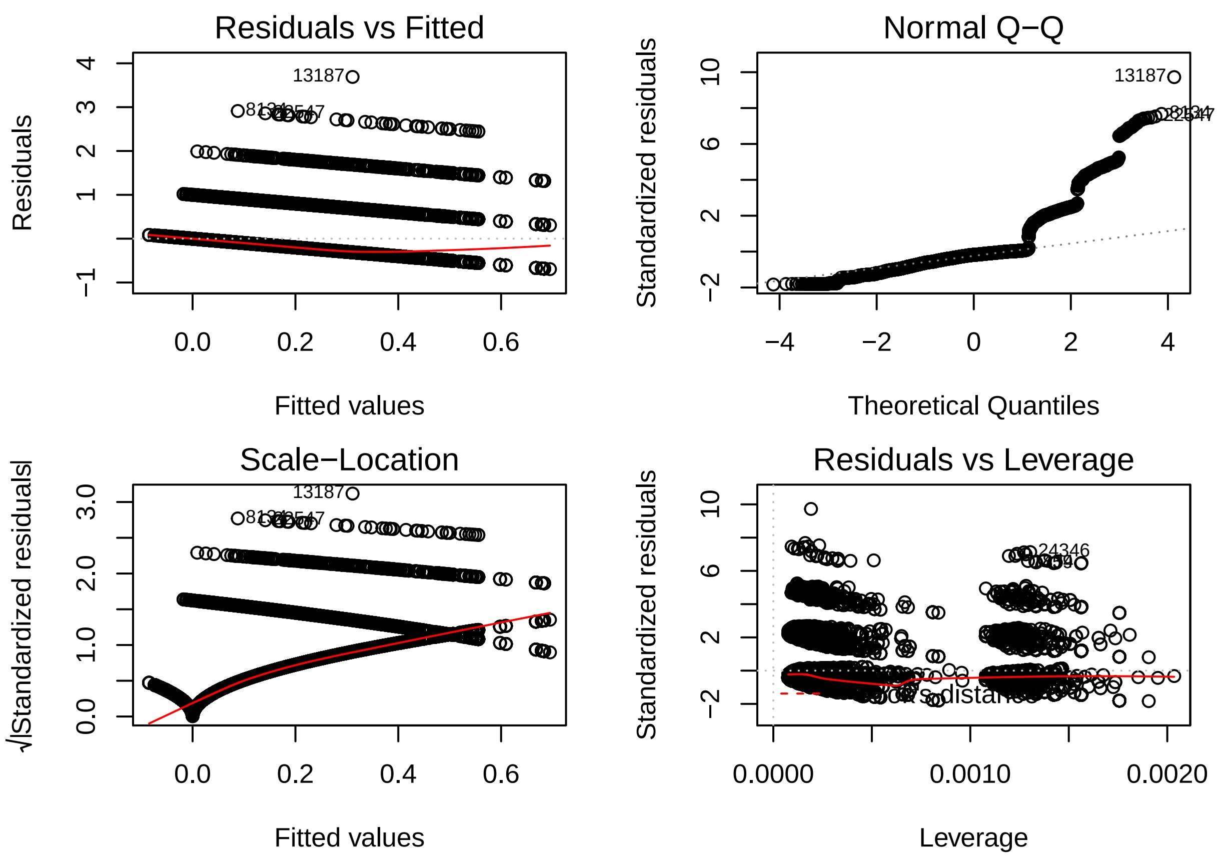
```
## [1] 0.1233666
```

```r
rmse_rectd <- sqrt(SSEtrg/nrow(TestRecy))
rmse_rectd
```

```
## [1] 2.282097
```

Regression plots below

```r
par(mar = c(4, 4, 2, 2), mfrow = c(2, 2))
plot(linRegRecTD2, which = c(1:3,5))
```



I would say that it appears that these charts do not like the prediction. Touchdowns are infrequent and random/unpredictable. If a receiver has an amazing season and gets 100 receptions, if they had 10 Tds it would be an All-pro year for them.

Additional summary statistics

```r
confint(linRegRecTD2)
```

```
##                     2.5 %       97.5 %
## (Intercept)   -0.2722684046 -0.188103591
## weight         0.0007819224  0.001151906
## home_team_1    0.0019738516  0.020798396
## ATL            0.0055473341  0.057670118
```

```
## DAL            0.0117263530  0.064271587
## DEN            0.0015988815  0.052275800
## GB             0.0328132348  0.082345950
## NE             0.0134682723  0.063443445
## NOR            0.0343100272  0.083705824
## avg_recy_plyr  0.0078026048  0.009290105
## avg_rec_plyr  -0.0352539989 -0.015101391
```

```
coef(summary(linRegRecTD2))
```

```
##                    Estimate   Std. Error    t value      Pr(>|t|)
## (Intercept)   -0.2301859980 2.147006e-02 -10.721253  9.139440e-27
## weight         0.0009669144 9.438125e-05  10.244772  1.383176e-24
## home_team_1    0.0113861240 4.802056e-03   2.371093  1.774241e-02
## ATL            0.0316087262 1.329629e-02   2.377260  1.744861e-02
## DAL            0.0379989698 1.340405e-02   2.834887  4.587539e-03
## DEN            0.0269373407 1.292745e-02   2.083731  3.719382e-02
## GB             0.0575795925 1.263557e-02   4.556944  5.212662e-06
## NE             0.0384558588 1.274844e-02   3.016515  2.559328e-03
## NOR            0.0590079255 1.260064e-02   4.682929  2.841556e-06
## avg_recy_plyr  0.0085463547 3.794545e-04  22.522741 2.534688e-111
## avg_rec_plyr  -0.0251776947 5.140839e-03  -4.897584  9.757292e-07
```

```
anova(linRegRecTD2)
```

```
## Analysis of Variance Table
##
## Response: tdrec
##                  Df Sum Sq Mean Sq   F value    Pr(>F)
## weight            1    0.0    0.00    0.0125  0.911033
## home_team_1       1    0.2    0.23    1.5752  0.209464
## ATL               1    1.2    1.22    8.4972  0.003560 **
## DAL               1    1.5    1.49   10.3239  0.001315 **
## DEN               1    0.8    0.77    5.3518  0.020709 *
## GB                1    4.7    4.66   32.3670 1.289e-08 ***
## NE                1    3.6    3.60   24.9949 5.784e-07 ***
## NOR               1    6.6    6.57   45.6328 1.455e-11 ***
## avg_recy_plyr     1  536.2  536.23 3725.7478 < 2.2e-16 ***
## avg_rec_plyr      1    3.5    3.45   23.9863 9.757e-07 ***
## Residuals     27473 3954.0    0.14
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Passing

Passing Yards

```
linRegQBpyds <- lm(py ~ height+ weight+cold_weather + hot_weather + home_team_1+ temp+
                  age+ forty1 + vertical1 + shuttle1+ cone1 + ARI + ATL + BAL + BUF + CAR + CHI+
                  CIN + CLE + DAL + DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR + NYG+
                  NYJ + OAK + PHI + PIT +SD + SEA + STL + TB + TEN + WAS + avg_recy_plyr+avg_recy_pos +
                    avg_recy_team + avg_rec_plyr +avg_rec_pos + avg_rec_team +avg_trg_plyr + avg_trg_pos
                    avg_trg_team + avg_rectd_plyr + avg_rectd_pos +avg_rectd_team+
                    avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
```

```r
                    avg_rbra_plyr + avg_rbra_pos +avg_rbra_team +
                    avg_rbry_plyr + avg_rbry_pos +avg_rbry_team +
                    avg_fuml_plyr + avg_fuml_pos +avg_fuml_team +
                    avg_qbpy_plyr + avg_qbpy_pos +avg_qbpy_team +
                    avg_qbpa_plyr + avg_qbpa_pos +avg_qbpa_team+
                    avg_qbpc_plyr + avg_qbpc_pos +avg_qbpc_team +
                    avg_qbints_plyr + avg_qbints_pos +avg_qbints_team +
                    avg_qbtdp_plyr + avg_qbtdp_pos +avg_qbtdp_team + grass_1 +
                  bad_weather_1, data = TrainRecy)

summary(linRegQBpyds)
```

```
##
## Call:
## lm(formula = py ~ height + weight + cold_weather + hot_weather +
##     home_team_1 + temp + age + forty1 + vertical1 + shuttle1 +
##     cone1 + ARI + ATL + BAL + BUF + CAR + CHI + CIN + CLE + DAL +
##     DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE +
##     NOR + NYG + NYJ + OAK + PHI + PIT + SD + SEA + STL + TB +
##     TEN + WAS + avg_recy_plyr + avg_recy_pos + avg_recy_team +
##     avg_rec_plyr + avg_rec_pos + avg_rec_team + avg_trg_plyr +
##     avg_trg_pos + avg_trg_team + avg_rectd_plyr + avg_rectd_pos +
##     avg_rectd_team + avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
##     avg_rbra_plyr + avg_rbra_pos + avg_rbra_team + avg_rbry_plyr +
##     avg_rbry_pos + avg_rbry_team + avg_fuml_plyr + avg_fuml_pos +
##     avg_fuml_team + avg_qbpy_plyr + avg_qbpy_pos + avg_qbpy_team +
##     avg_qbpa_plyr + avg_qbpa_pos + avg_qbpa_team + avg_qbpc_plyr +
##     avg_qbpc_pos + avg_qbpc_team + avg_qbints_plyr + avg_qbints_pos +
##     avg_qbints_team + avg_qbtdp_plyr + avg_qbtdp_pos + avg_qbtdp_team +
##     grass_1 + bad_weather_1, data = TrainRecy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -260.64   -1.03   -0.05    0.88  390.89
##
## Coefficients: (18 not defined because of singularities)
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       15.13692   13.54921   1.117 0.263926
## height            -0.08491    0.11572  -0.734 0.463101
## weight             0.01020    0.01546   0.660 0.509515
## cold_weather      -2.41912    0.69969  -3.457 0.000546 ***
## hot_weather       -1.65113    2.77533  -0.595 0.551893
## home_team_1        0.39521    0.36578   1.080 0.279938
## temp              -0.03234    0.01994  -1.622 0.104823
## age               -0.05744    0.05487  -1.047 0.295204
## forty1            -0.27164    1.85027  -0.147 0.883281
## vertical1          0.02653    0.07561   0.351 0.725648
## shuttle1          -1.60269    1.60489  -0.999 0.317983
## cone1             -0.08114    1.16066  -0.070 0.944267
## ARI                0.03664    1.33626   0.027 0.978125
## ATL               -0.60768    1.33837  -0.454 0.649801
## BAL               -0.70680    1.30469  -0.542 0.588004
## BUF                1.97060    1.33635   1.475 0.140327
## CAR               -0.11776    1.31592  -0.089 0.928696
```

```
## CHI               0.82380    1.34564   0.612 0.540411
## CIN              -0.88860    1.33723  -0.665 0.506373
## CLE               1.29849    1.32479   0.980 0.327024
## DAL              -0.23069    1.34476  -0.172 0.863797
## DEN               1.88545    1.30231   1.448 0.147692
## DET               0.10678    1.34206   0.080 0.936584
## GB                0.40471    1.30660   0.310 0.756760
## HOU               1.50148    1.32333   1.135 0.256545
## IND              -0.84919    1.32953  -0.639 0.523014
## JAC              -1.00420    1.35074  -0.743 0.457221
## KC                1.91120    1.30979   1.459 0.144531
## MIA              -0.05454    1.33532  -0.041 0.967418
## MINN             -0.07343    1.35123  -0.054 0.956663
## NE               -1.42241    1.31822  -1.079 0.280580
## NOR              -0.02537    1.31894  -0.019 0.984651
## NYG              -0.35659    1.32273  -0.270 0.787480
## NYJ              -0.24539    1.32453  -0.185 0.853024
## OAK               1.15714    1.30425   0.887 0.374978
## PHI               2.79899    1.33963   2.089 0.036683 *
## PIT               0.60844    1.31561   0.462 0.643740
## SD               -0.09593    1.34387  -0.071 0.943095
## SEA               1.05901    1.30659   0.811 0.417651
## STL              -0.72234    1.34721  -0.536 0.591843
## TB                0.29693    1.33293   0.223 0.823718
## TEN               1.00585    1.31563   0.765 0.444552
## WAS               0.53530    1.31296   0.408 0.683493
## avg_recy_plyr    -0.01215    0.04798  -0.253 0.800062
## avg_recy_pos     -0.05208    7.07043  -0.007 0.994123
## avg_recy_team          NA         NA      NA       NA
## avg_rec_plyr      0.12927    0.73673   0.175 0.860715
## avg_rec_pos       0.96047   15.68746   0.061 0.951180
## avg_rec_team           NA         NA      NA       NA
## avg_trg_plyr     -0.01591    0.54414  -0.029 0.976675
## avg_trg_pos      -0.16303   79.79666  -0.002 0.998370
## avg_trg_team           NA         NA      NA       NA
## avg_rectd_plyr    1.08899    2.56903   0.424 0.671648
## avg_rectd_pos    -1.08432  370.13686  -0.003 0.997663
## avg_rectd_team         NA         NA      NA       NA
## avg_tdr_plyr      3.86210    3.11221   1.241 0.214636
## avg_tdr_pos      -6.54813  181.63111  -0.036 0.971241
## avg_tdr_team           NA         NA      NA       NA
## avg_rbra_plyr     0.24904    0.30554   0.815 0.415028
## avg_rbra_pos     -0.39056   12.73078  -0.031 0.975526
## avg_rbra_team          NA         NA      NA       NA
## avg_rbry_plyr    -0.08135    0.07095  -1.146 0.251610
## avg_rbry_pos      0.08936    1.00148   0.089 0.928898
## avg_rbry_team          NA         NA      NA       NA
## avg_fuml_plyr    -0.12731    3.69815  -0.034 0.972538
## avg_fuml_pos     15.46517   50.90814   0.304 0.761293
## avg_fuml_team          NA         NA      NA       NA
## avg_qbpy_plyr     0.99908    0.06041  16.539  < 2e-16 ***
## avg_qbpy_pos           NA         NA      NA       NA
## avg_qbpy_team          NA         NA      NA       NA
## avg_qbpa_plyr     0.17620    0.46252   0.381 0.703240
```

```
## avg_qbpa_pos             NA         NA      NA       NA
## avg_qbpa_team            NA         NA      NA       NA
## avg_qbpc_plyr      -0.43099    0.82815  -0.520 0.602772
## avg_qbpc_pos             NA         NA      NA       NA
## avg_qbpc_team            NA         NA      NA       NA
## avg_qbints_plyr     0.36318    2.77863   0.131 0.896010
## avg_qbints_pos           NA         NA      NA       NA
## avg_qbints_team          NA         NA      NA       NA
## avg_qbtdp_plyr      0.44007    2.97792   0.148 0.882519
## avg_qbtdp_pos            NA         NA      NA       NA
## avg_qbtdp_team           NA         NA      NA       NA
## grass_1            -0.44260    0.37748  -1.172 0.241010
## bad_weather_1      -2.19968    0.74636  -2.947 0.003209 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 27.17 on 27417 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.8702, Adjusted R-squared:  0.8699
## F-statistic:  2827 on 65 and 27417 DF,  p-value: < 2.2e-16
```

Second run

```r
linRegQBpyds2 <- lm(py ~ cold_weather + BUF + PHI+ avg_qbpy_plyr+
                     bad_weather_1, data = TrainRecy)

summary(linRegQBpyds2)
```

```
##
## Call:
## lm(formula = py ~ cold_weather + BUF + PHI + avg_qbpy_plyr +
##     bad_weather_1, data = TrainRecy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -261.33   -0.32   -0.32    1.01  392.79
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.321034   0.201705   1.592  0.11149
## cold_weather  -1.330646   0.385069  -3.456  0.00055 ***
## BUF            1.892014   0.957019   1.977  0.04805 *
## PHI            2.645224   0.977462   2.706  0.00681 **
## avg_qbpy_plyr  0.997199   0.002326 428.788  < 2e-16 ***
## bad_weather_1 -1.971847   0.734475  -2.685  0.00726 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 27.16 on 27478 degrees of freedom
## Multiple R-squared:   0.87,  Adjusted R-squared:   0.87
## F-statistic: 3.678e+04 on 5 and 27478 DF,  p-value: < 2.2e-16
```

This prediction actually turned out to be pretty strong (R2 0.87) and we did it with only 5 variables! This makes sense to me. QB is a 1 player position, with a significant investment in the player. A QB will get time to mature and prove himself good or bad.

Testing the data, we see that the training set and the test set are similar. The model seems to hold up through testing

```
PydsdPredicted <- predict(linRegQBpyds2, newdata = TestRecy)

SSEpyds <- sum((PydsdPredicted - TestRecy$py)^2)
SSTpyds <- sum((mean(nfl_data$py)-TestRecy$py)^2)
r2_pyds <- 1 - SSEpyds/SSTpyds
r2_pyds
```

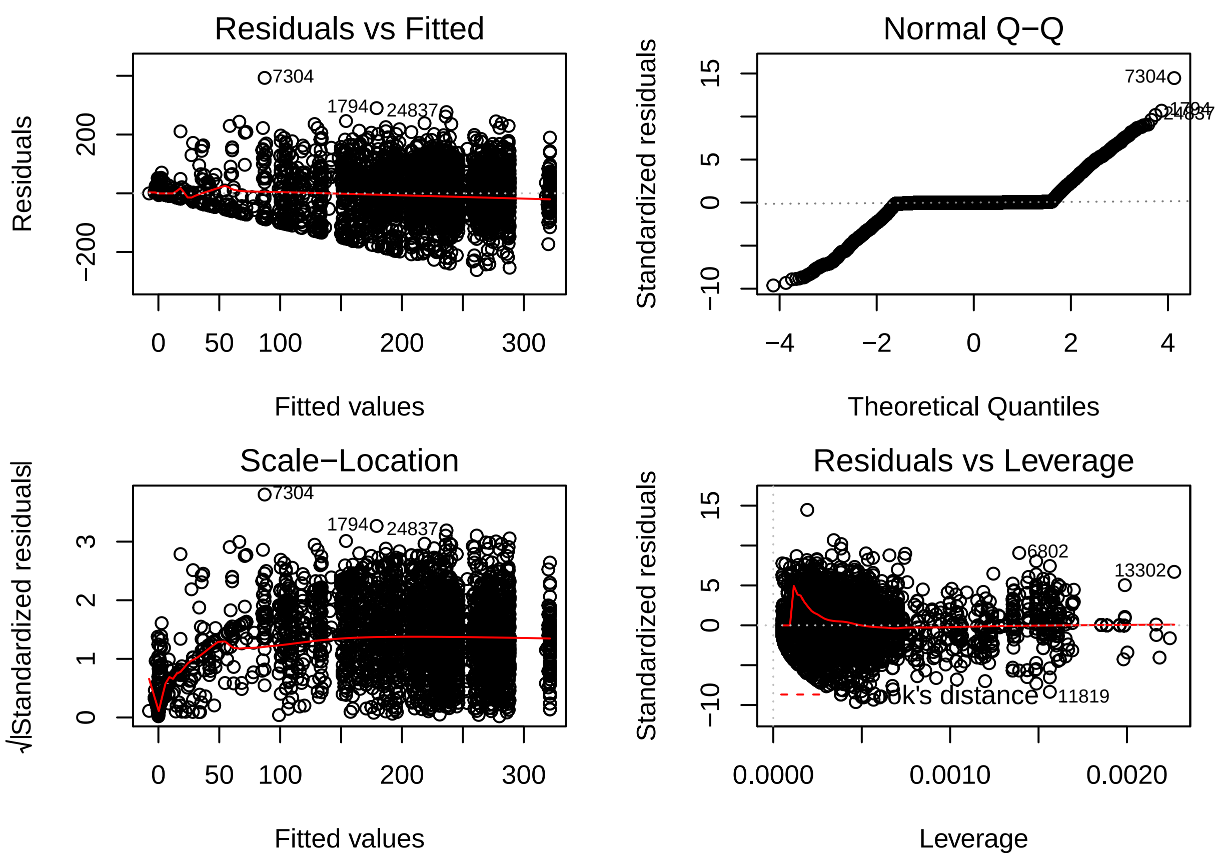
```
## [1] 0.8641837
```

```
rmse_pyds <- sqrt(SSEpyds/nrow(TestRecy))
rmse_pyds
```

```
## [1] 28.27021
```

Below are the regression plots:

```
par(mar = c(4, 4, 2, 2), mfrow = c(2, 2))
plot(linRegQBpyds2, which = c(1:3,5))
```



The normal Q-Q shows are data has extreme values, and means the data is probably not normally distributed

Some more summary statistics:

```
confint(linRegQBpyds2)
```

```
##                    2.5 %     97.5 %
## (Intercept)   -0.07431867  0.7163872
## cold_weather  -2.08540030 -0.5758918
```

```
## BUF            0.01620734  3.7678198
## PHI            0.72934893  4.5610991
## avg_qbpy_plyr  0.99264115  1.0017578
## bad_weather_1 -3.41145479 -0.5322399
```

```r
coef(summary(linRegQBpyds2))
```

```
##                 Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)    0.3210342 0.201705491   1.591599 0.1114863527
## cold_weather  -1.3306460 0.385068819  -3.455606 0.0005498868
## BUF            1.8920136 0.957019434   1.976986 0.0480532443
## PHI            2.6452240 0.977462192   2.706216 0.0068096732
## avg_qbpy_plyr  0.9971995 0.002325623 428.788127 0.0000000000
## bad_weather_1 -1.9718473 0.734474740  -2.684704 0.0072637593
```

```r
anova(linRegQBpyds2)
```

```
## Analysis of Variance Table
##
## Response: py
##                  Df    Sum Sq   Mean Sq    F value    Pr(>F)
## cold_weather      1     10884     10884 1.4760e+01 0.0001224 ***
## BUF               1      7100      7100 9.6283e+00 0.0019179 **
## PHI               1     11271     11271 1.5284e+01 9.272e-05 ***
## avg_qbpy_plyr     1 135581122 135581122 1.8385e+05 < 2.2e-16 ***
## bad_weather_1     1      5315      5315 7.2076e+00 0.0072638 **
## Residuals     27478  20263287       737
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Conclusion, this predicts well, but it seems like this could be driven by some chance. QB statistics remind me of baseball statistics. One batter vs One pitcher. A QB is really like a pitcher or a batter. The opponent is more complex, however one side of the equation is “controlled”.

Pass completions

```r
linRegQBpc <- lm(pc ~ height+ weight+cold_weather + hot_weather + home_team_1+ temp+
                   age+ forty1 + vertical1 + shuttle1+ cone1 + ARI + ATL + BAL + BUF + CAR + CHI+
                   CIN + CLE + DAL + DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR + NY
                   NYJ + OAK + PHI + PIT +SD + SEA + STL + TB + TEN + WAS +avg_recy_plyr+avg_recy_pos +
                   avg_recy_team + avg_rec_plyr +avg_rec_pos + avg_rec_team +avg_trg_plyr + avg_trg_pos
                   avg_trg_team + avg_rectd_plyr + avg_rectd_pos +avg_rectd_team+
                   avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
                   avg_rbra_plyr + avg_rbra_pos +avg_rbra_team +
                   avg_rbry_plyr + avg_rbry_pos +avg_rbry_team +
                   avg_fuml_plyr + avg_fuml_pos +avg_fuml_team +
                   avg_qbpy_plyr + avg_qbpy_pos +avg_qbpy_team +
                   avg_qbpa_plyr + avg_qbpa_pos +avg_qbpa_team+
                   avg_qbpc_plyr + avg_qbpc_pos +avg_qbpc_team +
                   avg_qbints_plyr + avg_qbints_pos +avg_qbints_team +
                   avg_qbtdp_plyr + avg_qbtdp_pos +avg_qbtdp_team + grass_1 +
                   bad_weather_1 , data = TrainRecy)
summary(linRegQBpc)
```

```
##
```

```
## Call:
## lm(formula = pc ~ height + weight + cold_weather + hot_weather +
##     home_team_1 + temp + age + forty1 + vertical1 + shuttle1 +
##     cone1 + ARI + ATL + BAL + BUF + CAR + CHI + CIN + CLE + DAL +
##     DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE +
##     NOR + NYG + NYJ + OAK + PHI + PIT + SD + SEA + STL + TB +
##     TEN + WAS + avg_recy_plyr + avg_recy_pos + avg_recy_team +
##     avg_rec_plyr + avg_rec_pos + avg_rec_team + avg_trg_plyr +
##     avg_trg_pos + avg_trg_team + avg_rectd_plyr + avg_rectd_pos +
##     avg_rectd_team + avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
##     avg_rbra_plyr + avg_rbra_pos + avg_rbra_team + avg_rbry_plyr +
##     avg_rbry_pos + avg_rbry_team + avg_fuml_plyr + avg_fuml_pos +
##     avg_fuml_team + avg_qbpy_plyr + avg_qbpy_pos + avg_qbpy_team +
##     avg_qbpa_plyr + avg_qbpa_pos + avg_qbpa_team + avg_qbpc_plyr +
##     avg_qbpc_pos + avg_qbpc_team + avg_qbints_plyr + avg_qbints_pos +
##     avg_qbints_team + avg_qbtdp_plyr + avg_qbtdp_pos + avg_qbtdp_team +
##     grass_1 + bad_weather_1, data = TrainRecy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -22.5115  -0.0746  -0.0060   0.0696  24.5659
##
## Coefficients: (18 not defined because of singularities)
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.8178704  1.1123275   1.634  0.10221
## height          -0.0131247  0.0094999  -1.382  0.16712
## weight           0.0010891  0.0012694   0.858  0.39092
## cold_weather    -0.1764290  0.0574409  -3.071  0.00213 **
## hot_weather     -0.1060755  0.2278417  -0.466  0.64153
## home_team_1     -0.0067672  0.0300285  -0.225  0.82170
## temp            -0.0020194  0.0016367  -1.234  0.21727
## age             -0.0007263  0.0045044  -0.161  0.87190
## forty1          -0.0651133  0.1518988  -0.429  0.66817
## vertical1        0.0006099  0.0062074   0.098  0.92173
## shuttle1        -0.1084613  0.1317538  -0.823  0.41039
## cone1           -0.0364611  0.0952852  -0.383  0.70198
## ARI              0.0259059  0.1097006   0.236  0.81332
## ATL             -0.0689069  0.1098737  -0.627  0.53057
## BAL             -0.0505477  0.1071094  -0.472  0.63698
## BUF              0.2141346  0.1097079   1.952  0.05096 .
## CAR             -0.0166184  0.1080311  -0.154  0.87775
## CHI              0.1104085  0.1104706   0.999  0.31759
## CIN              0.0659332  0.1097803   0.601  0.54812
## CLE              0.1770124  0.1087595   1.628  0.10363
## DAL              0.0213817  0.1103987   0.194  0.84643
## DEN              0.0983087  0.1069138   0.920  0.35783
## DET             -0.0115361  0.1101769  -0.105  0.91661
## GB               0.0703694  0.1072661   0.656  0.51181
## HOU              0.1474648  0.1086394   1.357  0.17467
## IND             -0.0260639  0.1091487  -0.239  0.81127
## JAC             -0.0473196  0.1108894  -0.427  0.66958
## KC               0.2061026  0.1075281   1.917  0.05528 .
## MIA              0.0481666  0.1096238   0.439  0.66039
## MINN             0.0346913  0.1109298   0.313  0.75449
```

```
## NE             -0.1143017  0.1082196  -1.056  0.29089
## NOR             0.0147959  0.1082786   0.137  0.89131
## NYG             0.0281654  0.1085901   0.259  0.79535
## NYJ            -0.0007443  0.1087375  -0.007  0.99454
## OAK             0.0770211  0.1070732   0.719  0.47194
## PHI             0.1950537  0.1099773   1.774  0.07614 .
## PIT             0.0817121  0.1080053   0.757  0.44932
## SD             -0.0005202  0.1103258  -0.005  0.99624
## SEA             0.1399500  0.1072648   1.305  0.19200
## STL            -0.0094197  0.1105994  -0.085  0.93213
## TB              0.0237595  0.1094272   0.217  0.82811
## TEN             0.0594506  0.1080072   0.550  0.58203
## WAS             0.0543927  0.1077881   0.505  0.61383
## avg_recy_plyr  -0.0003054  0.0039388  -0.078  0.93819
## avg_recy_pos   -0.0035540  0.5804501  -0.006  0.99511
## avg_recy_team          NA         NA      NA       NA
## avg_rec_plyr   -0.0057142  0.0604825  -0.094  0.92473
## avg_rec_pos     0.1988573  1.2878684   0.154  0.87729
## avg_rec_team           NA         NA      NA       NA
## avg_trg_plyr    0.0006451  0.0446712   0.014  0.98848
## avg_trg_pos    -0.0973516  6.5509386  -0.015  0.98814
## avg_trg_team           NA         NA      NA       NA
## avg_rectd_plyr  0.1352430  0.2109055   0.641  0.52137
## avg_rectd_pos   0.1058621 30.3865349   0.003  0.99722
## avg_rectd_team         NA         NA      NA       NA
## avg_tdr_plyr    0.1487660  0.2554984   0.582  0.56040
## avg_tdr_pos    -0.4510668 14.9110794  -0.030  0.97587
## avg_tdr_team           NA         NA      NA       NA
## avg_rbra_plyr   0.0045750  0.0250836   0.182  0.85528
## avg_rbra_pos   -0.0470401  1.0451383  -0.045  0.96410
## avg_rbra_team          NA         NA      NA       NA
## avg_rbry_plyr  -0.0015158  0.0058250  -0.260  0.79469
## avg_rbry_pos    0.0086982  0.0822166   0.106  0.91575
## avg_rbry_team          NA         NA      NA       NA
## avg_fuml_plyr  -0.1218559  0.3036009  -0.401  0.68815
## avg_fuml_pos    2.0153211  4.1793246   0.482  0.62966
## avg_fuml_team          NA         NA      NA       NA
## avg_qbpy_plyr  -0.0054096  0.0049592  -1.091  0.27536
## avg_qbpy_pos           NA         NA      NA       NA
## avg_qbpy_team          NA         NA      NA       NA
## avg_qbpa_plyr   0.0270185  0.0379710   0.712  0.47675
## avg_qbpa_pos           NA         NA      NA       NA
## avg_qbpa_team          NA         NA      NA       NA
## avg_qbpc_plyr   0.9933952  0.0679872  14.612  < 2e-16 ***
## avg_qbpc_pos           NA         NA      NA       NA
## avg_qbpc_team          NA         NA      NA       NA
## avg_qbints_plyr -0.1951935  0.2281125  -0.856  0.39218
## avg_qbints_pos         NA         NA      NA       NA
## avg_qbints_team        NA         NA      NA       NA
## avg_qbtdp_plyr  0.3165858  0.2444736   1.295  0.19534
## avg_qbtdp_pos          NA         NA      NA       NA
## avg_qbtdp_team         NA         NA      NA       NA
## grass_1        -0.0330546  0.0309896  -1.067  0.28615
## bad_weather_1  -0.1828291  0.0612726  -2.984  0.00285 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.23 on 27417 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.8813, Adjusted R-squared:  0.8811
## F-statistic:  3133 on 65 and 27417 DF,  p-value: < 2.2e-16
```

Second run

```r
linRegQBpc2 <- lm(pc ~ cold_weather + BUF + CLE+ HOU + KC+ PHI+ avg_qbpc_plyr+
                    bad_weather_1, data = TrainRecy)

summary(linRegQBpc2)
```

```
##
## Call:
## lm(formula = pc ~ cold_weather + BUF + CLE + HOU + KC + PHI +
##     avg_qbpc_plyr + bad_weather_1, data = TrainRecy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -22.4586  -0.0119  -0.0119   0.0813  24.5277
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.011935   0.017163   0.695 0.486793
## cold_weather  -0.106883   0.031750  -3.366 0.000763 ***
## BUF            0.195798   0.078698   2.488 0.012854 *
## CLE            0.140747   0.078607   1.791 0.073383 .
## HOU            0.120292   0.076938   1.564 0.117945
## KC             0.178582   0.077008   2.319 0.020402 *
## PHI            0.167861   0.080368   2.089 0.036749 *
## avg_qbpc_plyr  0.999110   0.002213 451.382  < 2e-16 ***
## bad_weather_1 -0.170054   0.060327  -2.819 0.004822 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.229 on 27475 degrees of freedom
## Multiple R-squared:  0.8812, Adjusted R-squared:  0.8812
## F-statistic: 2.548e+04 on 8 and 27475 DF,  p-value: < 2.2e-16
```

Again, R2 is very strong, model stays strong when model is shrunk to 8 variables. Not surprising (to me) bad weather and cold weather hurt completions. It is harder to throw a ball in bad weather and catch a ball in bad weather.

Testing the data, we see that the training set and the test set are similar. The model seems to hold up through testing

```r
PcPredicted <- predict(linRegQBpc2, newdata = TestRecy)

SSEpc <- sum((PcPredicted - TestRecy$pc)^2)
SSTpc <- sum((mean(nfl_data$pc)-TestRecy$pc)^2)
r2_pc <- 1 - SSEpc/SSTpc
r2_pc
```

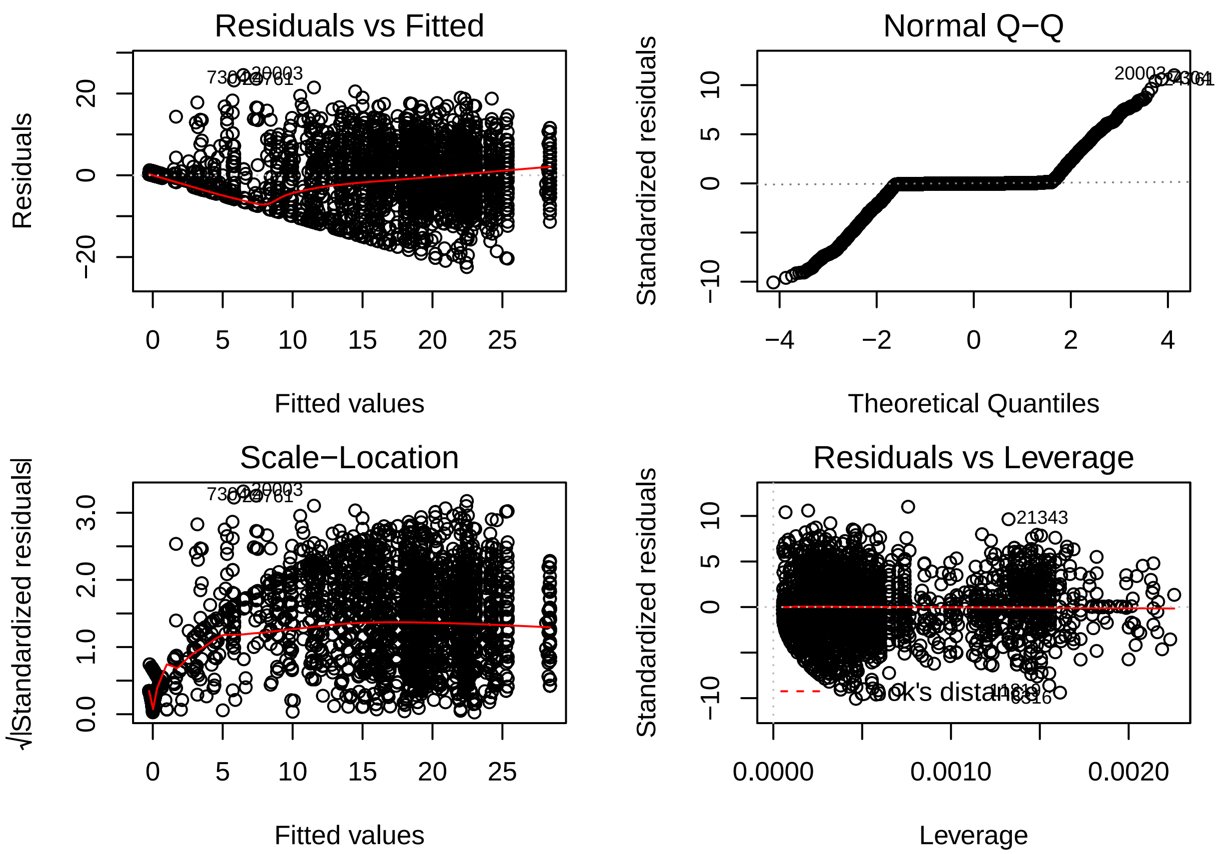
```
## [1] 0.8742303
```

```
rmse_pc <- sqrt(SSEpc/nrow(TestRecy))
rmse_pc
```

```
## [1] 2.317271
```

Regression plots:

```
par(mar = c(4, 4, 2, 2), mfrow = c(2, 2))
plot(linRegQBpc2, which = c(1:3,5))
```



Very similar problems/characteristics as we saw in yards

Summary statistics:

```
confint(linRegQBpc2)
```

```
##                     2.5 %      97.5 %
## (Intercept)   -0.02170414  0.04557480
## cold_weather  -0.16911548 -0.04465100
## BUF            0.04154486  0.35005017
## CLE           -0.01332681  0.29481985
## HOU           -0.03050916  0.27109361
## KC             0.02764192  0.32952201
## PHI            0.01033444  0.32538662
## avg_qbpc_plyr  0.99477183  1.00344877
## bad_weather_1 -0.28829698 -0.05181085
```

```r
coef(summary(linRegQBpc2))
```

```
##                  Estimate  Std. Error     t value     Pr(>|t|)
## (Intercept)    0.01193533 0.017162555   0.6954287 0.4867925880
## cold_weather  -0.10688324 0.031750328  -3.3663664 0.0007626983
## BUF            0.19579751 0.078698313   2.4879506 0.0128540388
## CLE            0.14074652 0.078606824   1.7905128 0.0733825164
## HOU            0.12029222 0.076937506   1.5635056 0.1179452256
## KC             0.17858197 0.077008249   2.3189979 0.0204024484
## PHI            0.16786053 0.080368390   2.0886387 0.0367494383
## avg_qbpc_plyr  0.99911030 0.002213448 451.3819314 0.0000000000
## bad_weather_1 -0.17005392 0.060326546  -2.8188903 0.0048224440
```

```r
anova(linRegQBpc2)
```

```
## Analysis of Variance Table
##
## Response: pc
##                  Df  Sum Sq Mean Sq    F value    Pr(>F)
## cold_weather      1      73      73 1.4634e+01 0.0001308 ***
## BUF               1      36      36 7.1527e+00 0.0074897 **
## CLE               1      62      62 1.2484e+01 0.0004110 ***
## HOU               1      44      44 8.8515e+00 0.0029311 **
## KC                1      98      98 1.9789e+01 8.681e-06 ***
## PHI               1      51      51 1.0267e+01 0.0013558 **
## avg_qbpc_plyr     1 1012335 1012335 2.0374e+05 < 2.2e-16 ***
## bad_weather_1     1      39      39 7.9461e+00 0.0048224 **
## Residuals     27475  136517       5
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Ints

```r
linRegQBInts <- lm(ints ~ height+ weight+cold_weather + hot_weather + home_team_1+ temp+
                     age+ forty1 + vertical1 + shuttle1+ cone1 + ARI + ATL + BAL + BUF + CAR + CHI+
                     CIN + CLE + DAL + DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR + N
                     NYJ + OAK + PHI + PIT +SD + SEA + STL + TB + TEN + WAS +avg_recy_plyr+avg_recy_pos
                     avg_recy_team + avg_rec_plyr +avg_rec_pos + avg_rec_team +avg_trg_plyr + avg_trg_po
                     avg_trg_team + avg_rectd_plyr + avg_rectd_pos +avg_rectd_team+
                     avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
                     avg_rbra_plyr + avg_rbra_pos +avg_rbra_team +
                     avg_rbry_plyr + avg_rbry_pos +avg_rbry_team +
                     avg_fuml_plyr + avg_fuml_pos +avg_fuml_team +
                     avg_qbpy_plyr + avg_qbpy_pos +avg_qbpy_team +
                     avg_qbpa_plyr + avg_qbpa_pos +avg_qbpa_team+
                     avg_qbpc_plyr + avg_qbpc_pos +avg_qbpc_team +
                     avg_qbints_plyr + avg_qbints_pos +avg_qbints_team +
                     avg_qbtdp_plyr + avg_qbtdp_pos +avg_qbtdp_team + grass_1 +
                     bad_weather_1 , data = TrainRecy)
summary(linRegQBInts)
```

```
##
## Call:
```

```
## lm(formula = ints ~ height + weight + cold_weather + hot_weather +
##     home_team_1 + temp + age + forty1 + vertical1 + shuttle1 +
##     cone1 + ARI + ATL + BAL + BUF + CAR + CHI + CIN + CLE + DAL +
##     DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE +
##     NOR + NYG + NYJ + OAK + PHI + PIT + SD + SEA + STL + TB +
##     TEN + WAS + avg_recy_plyr + avg_recy_pos + avg_recy_team +
##     avg_rec_plyr + avg_rec_pos + avg_rec_team + avg_trg_plyr +
##     avg_trg_pos + avg_trg_team + avg_rectd_plyr + avg_rectd_pos +
##     avg_rectd_team + avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
##     avg_rbra_plyr + avg_rbra_pos + avg_rbra_team + avg_rbry_plyr +
##     avg_rbry_pos + avg_rbry_team + avg_fuml_plyr + avg_fuml_pos +
##     avg_fuml_team + avg_qbpy_plyr + avg_qbpy_pos + avg_qbpy_team +
##     avg_qbpa_plyr + avg_qbpa_pos + avg_qbpa_team + avg_qbpc_plyr +
##     avg_qbpc_pos + avg_qbpc_team + avg_qbints_plyr + avg_qbints_pos +
##     avg_qbints_team + avg_qbtdp_plyr + avg_qbtdp_pos + avg_qbtdp_team +
##     grass_1 + bad_weather_1, data = TrainRecy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.5189 -0.0070  0.0004  0.0072  4.3767
##
## Coefficients: (18 not defined because of singularities)
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.341e-01  1.495e-01   0.897   0.3696
## height          -6.517e-04  1.277e-03  -0.510   0.6097
## weight           9.973e-05  1.706e-04   0.585   0.5588
## cold_weather     1.421e-03  7.719e-03   0.184   0.8539
## hot_weather      1.627e-02  3.062e-02   0.531   0.5951
## home_team_1     -4.474e-03  4.035e-03  -1.109   0.2676
## temp            -3.581e-05  2.199e-04  -0.163   0.8707
## age             -8.304e-04  6.053e-04  -1.372   0.1701
## forty1          -1.142e-02  2.041e-02  -0.560   0.5757
## vertical1       -3.146e-05  8.341e-04  -0.038   0.9699
## shuttle1         6.214e-03  1.770e-02   0.351   0.7256
## cone1           -9.495e-03  1.280e-02  -0.742   0.4583
## ARI             -1.073e-02  1.474e-02  -0.728   0.4666
## ATL             -7.613e-03  1.476e-02  -0.516   0.6061
## BAL              7.861e-03  1.439e-02   0.546   0.5850
## BUF              3.229e-03  1.474e-02   0.219   0.8266
## CAR              7.727e-03  1.452e-02   0.532   0.5946
## CHI             -1.540e-02  1.484e-02  -1.037   0.2996
## CIN              2.851e-03  1.475e-02   0.193   0.8468
## CLE              1.387e-02  1.461e-02   0.949   0.3428
## DAL             -2.633e-03  1.484e-02  -0.178   0.8591
## DEN              3.242e-03  1.437e-02   0.226   0.8215
## DET             -3.390e-03  1.481e-02  -0.229   0.8189
## GB              -1.303e-03  1.441e-02  -0.090   0.9280
## HOU             -8.604e-04  1.460e-02  -0.059   0.9530
## IND             -6.149e-03  1.467e-02  -0.419   0.6751
## JAC             -6.388e-03  1.490e-02  -0.429   0.6681
## KC              -4.528e-03  1.445e-02  -0.313   0.7540
## MIA             -1.006e-02  1.473e-02  -0.683   0.4946
## MINN            -4.039e-03  1.491e-02  -0.271   0.7864
## NE              -1.166e-02  1.454e-02  -0.802   0.4226
```

```
## NOR              2.247e-03  1.455e-02   0.154   0.8773
## NYG             -8.899e-03  1.459e-02  -0.610   0.5420
## NYJ             -6.918e-03  1.461e-02  -0.473   0.6359
## OAK              1.423e-02  1.439e-02   0.989   0.3225
## PHI              1.534e-02  1.478e-02   1.038   0.2993
## PIT             -1.056e-02  1.451e-02  -0.728   0.4667
## SD              -6.547e-03  1.483e-02  -0.442   0.6588
## SEA             -2.716e-03  1.441e-02  -0.188   0.8506
## STL              1.317e-03  1.486e-02   0.089   0.9294
## TB              -8.072e-03  1.470e-02  -0.549   0.5830
## TEN             -1.153e-02  1.451e-02  -0.794   0.4270
## WAS              3.514e-03  1.448e-02   0.243   0.8083
## avg_recy_plyr   -1.057e-04  5.293e-04  -0.200   0.8418
## avg_recy_pos     1.029e-02  7.800e-02   0.132   0.8951
## avg_recy_team           NA         NA      NA       NA
## avg_rec_plyr     8.228e-04  8.127e-03   0.101   0.9194
## avg_rec_pos      4.258e-02  1.731e-01   0.246   0.8057
## avg_rec_team            NA         NA      NA       NA
## avg_trg_plyr     4.301e-04  6.003e-03   0.072   0.9429
## avg_trg_pos     -1.290e-01  8.803e-01  -0.147   0.8835
## avg_trg_team            NA         NA      NA       NA
## avg_rectd_plyr   3.304e-03  2.834e-02   0.117   0.9072
## avg_rectd_pos    5.343e-01  4.083e+00   0.131   0.8959
## avg_rectd_team          NA         NA      NA       NA
## avg_tdr_plyr     1.769e-02  3.433e-02   0.515   0.6065
## avg_tdr_pos     -2.640e-01  2.004e+00  -0.132   0.8952
## avg_tdr_team            NA         NA      NA       NA
## avg_rbra_plyr    2.997e-03  3.371e-03   0.889   0.3739
## avg_rbra_pos     4.072e-03  1.404e-01   0.029   0.9769
## avg_rbra_team           NA         NA      NA       NA
## avg_rbry_plyr   -8.259e-04  7.827e-04  -1.055   0.2914
## avg_rbry_pos     2.294e-03  1.105e-02   0.208   0.8355
## avg_rbry_team           NA         NA      NA       NA
## avg_fuml_plyr    6.131e-03  4.080e-02   0.150   0.8805
## avg_fuml_pos     2.089e-01  5.616e-01   0.372   0.7099
## avg_fuml_team           NA         NA      NA       NA
## avg_qbpy_plyr   -2.294e-04  6.664e-04  -0.344   0.7306
## avg_qbpy_pos            NA         NA      NA       NA
## avg_qbpy_team           NA         NA      NA       NA
## avg_qbpa_plyr    5.281e-03  5.102e-03   1.035   0.3007
## avg_qbpa_pos            NA         NA      NA       NA
## avg_qbpa_team           NA         NA      NA       NA
## avg_qbpc_plyr   -7.359e-03  9.136e-03  -0.805   0.4206
## avg_qbpc_pos            NA         NA      NA       NA
## avg_qbpc_team           NA         NA      NA       NA
## avg_qbints_plyr  8.796e-01  3.065e-02  28.695   <2e-16 ***
## avg_qbints_pos          NA         NA      NA       NA
## avg_qbints_team         NA         NA      NA       NA
## avg_qbtdp_plyr   5.736e-02  3.285e-02   1.746   0.0808 .
## avg_qbtdp_pos           NA         NA      NA       NA
## avg_qbtdp_team          NA         NA      NA       NA
## grass_1         -3.639e-04  4.164e-03  -0.087   0.9304
## bad_weather_1    1.181e-02  8.234e-03   1.435   0.1513
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2997 on 27417 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.418,  Adjusted R-squared:  0.4166
## F-statistic: 302.9 on 65 and 27417 DF,  p-value: < 2.2e-16
```

Second run

```
linRegQBInts2 <- lm(ints ~ avg_qbints_plyr, data = TrainRecy)

summary(linRegQBInts2)
```

```
##
## Call:
## lm(formula = ints ~ avg_qbints_plyr, data = TrainRecy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.5989 -0.0007 -0.0007 -0.0007  4.4048
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     0.0007212  0.0019048   0.379    0.705
## avg_qbints_plyr 0.9766662  0.0069667 140.190   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2996 on 27482 degrees of freedom
## Multiple R-squared:  0.417,  Adjusted R-squared:  0.4169
## F-statistic: 1.965e+04 on 1 and 27482 DF,  p-value: < 2.2e-16
```

R2 of 0.4169, and really the only meaningful predictor was "how many interceptions have you thrown in the past". Not an entirely amazing model, but at least we have something here.

Testing the data, we see that the training set and the test set are similar. The model seems to hold up through testing

```
PintPredicted <- predict(linRegQBInts2, newdata = TestRecy)

SSEint <- sum((PintPredicted - TestRecy$ints)^2)
SSTint <- sum((mean(nfl_data$ints)-TestRecy$ints)^2)
r2_int <- 1 - SSEint/SSTint
r2_int
```

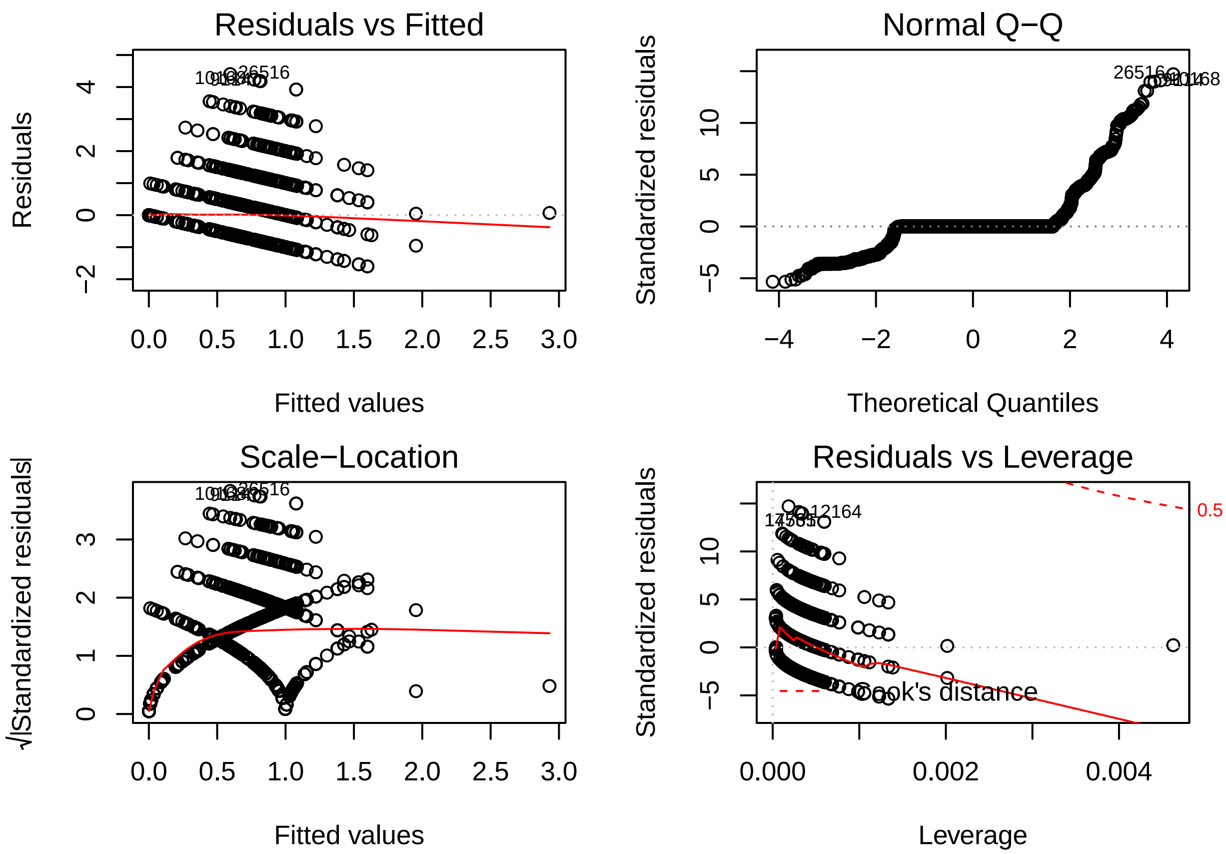
```
## [1] 0.4479224
```

```
rmse_int <- sqrt(SSEint/nrow(TestRecy))
rmse_int
```

```
## [1] 0.309185
```

Regression plots:

```
par(mar = c(4, 4, 2, 2), mfrow = c(2, 2))
plot(linRegQBInts2, which = c(1:3,5))
```



Summary statistics:

```r
confint(linRegQBInts2)
```

```
##                        2.5 %      97.5 %
## (Intercept)     -0.003012216 0.004454602
## avg_qbints_plyr  0.963011073 0.990321292
```

```r
coef(summary(linRegQBInts2))
```

```
##                     Estimate  Std. Error     t value  Pr(>|t|)
## (Intercept)     0.0007211932 0.001904752   0.3786285 0.7049667
## avg_qbints_plyr 0.9766661826 0.006966714 140.1903697 0.0000000
```

```r
anova(linRegQBInts2)
```

```
## Analysis of Variance Table
##
## Response: ints
##                    Df Sum Sq Mean Sq F value    Pr(>F)
## avg_qbints_plyr     1 1764.2 1764.19   19653 < 2.2e-16 ***
## Residuals       27482 2466.9    0.09
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Pass Attempts

```
linRegQBpa <- lm(pa ~ height+ weight+cold_weather + hot_weather + home_team_1+ temp+
                     age+ forty1 + vertical1 + shuttle1+ cone1 + ARI + ATL + BAL + BUF + CAR + CHI+
                     CIN + CLE + DAL + DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR +
                     NYJ + OAK + PHI + PIT +SD + SEA + STL + TB + TEN + WAS +avg_recy_plyr+avg_recy_pos
                   avg_recy_team + avg_rec_plyr +avg_rec_pos + avg_rec_team +avg_trg_plyr + avg_trg_pos
                   avg_trg_team + avg_rectd_plyr + avg_rectd_pos +avg_rectd_team+
                   avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
                   avg_rbra_plyr + avg_rbra_pos +avg_rbra_team +
                   avg_rbry_plyr + avg_rbry_pos +avg_rbry_team +
                   avg_fuml_plyr + avg_fuml_pos +avg_fuml_team +
                   avg_qbpy_plyr + avg_qbpy_pos +avg_qbpy_team +
                   avg_qbpa_plyr + avg_qbpa_pos +avg_qbpa_team+
                   avg_qbpc_plyr + avg_qbpc_pos +avg_qbpc_team +
                   avg_qbints_plyr + avg_qbints_pos +avg_qbints_team +
                   avg_qbtdp_plyr + avg_qbtdp_pos +avg_qbtdp_team + grass_1 +
                     bad_weather_1 , data = nfl_data)

summary(linRegQBpa)
```

```
##
## Call:
## lm(formula = pa ~ height + weight + cold_weather + hot_weather +
##     home_team_1 + temp + age + forty1 + vertical1 + shuttle1 +
##     cone1 + ARI + ATL + BAL + BUF + CAR + CHI + CIN + CLE + DAL +
##     DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE +
##     NOR + NYG + NYJ + OAK + PHI + PIT + SD + SEA + STL + TB +
##     TEN + WAS + avg_recy_plyr + avg_recy_pos + avg_recy_team +
##     avg_rec_plyr + avg_rec_pos + avg_rec_team + avg_trg_plyr +
##     avg_trg_pos + avg_trg_team + avg_rectd_plyr + avg_rectd_pos +
##     avg_rectd_team + avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
##     avg_rbra_plyr + avg_rbra_pos + avg_rbra_team + avg_rbry_plyr +
##     avg_rbry_pos + avg_rbry_team + avg_fuml_plyr + avg_fuml_pos +
##     avg_fuml_team + avg_qbpy_plyr + avg_qbpy_pos + avg_qbpy_team +
##     avg_qbpa_plyr + avg_qbpa_pos + avg_qbpa_team + avg_qbpc_plyr +
##     avg_qbpc_pos + avg_qbpc_team + avg_qbints_plyr + avg_qbints_pos +
##     avg_qbints_team + avg_qbtdp_plyr + avg_qbtdp_pos + avg_qbtdp_team +
##     grass_1 + bad_weather_1, data = nfl_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.688  -0.119   0.013   0.121  38.506
##
## Coefficients: (18 not defined because of singularities)
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.1210957  1.4574365   0.083  0.93378
## height          -0.0041903  0.0125466  -0.334  0.73840
## weight           0.0002808  0.0016793   0.167  0.86721
## cold_weather    -0.2040206  0.0757346  -2.694  0.00707 **
## hot_weather      0.1424447  0.2992858   0.476  0.63411
## home_team_1     -0.0959559  0.0394788  -2.431  0.01508 *
## temp            -0.0046610  0.0021578  -2.160  0.03077 *
## age             -0.0003251  0.0059139  -0.055  0.95616
```

```
## forty1           0.0709123  0.1994720   0.355  0.72222
## vertical1        0.0011589  0.0081367   0.142  0.88675
## shuttle1        -0.0269783  0.1731461  -0.156  0.87618
## cone1            0.0154795  0.1259713   0.123  0.90220
## ARI              0.2208477  0.1442256   1.531  0.12571
## ATL             -0.0517539  0.1443144  -0.359  0.71988
## BAL              0.0195390  0.1400899   0.139  0.88908
## BUF              0.3770709  0.1458488   2.585  0.00973 **
## CAR             -0.0097683  0.1427295  -0.068  0.94544
## CHI              0.0677798  0.1458321   0.465  0.64209
## CIN              0.0172179  0.1447971   0.119  0.90535
## CLE              0.4224136  0.1432844   2.948  0.00320 **
## DAL             -0.1209422  0.1461838  -0.827  0.40806
## DEN              0.1038554  0.1405913   0.739  0.46009
## DET              0.1349612  0.1451048   0.930  0.35233
## GB               0.0998653  0.1414408   0.706  0.48016
## HOU              0.3693154  0.1435572   2.573  0.01010 *
## IND              0.0307118  0.1435663   0.214  0.83061
## JAC              0.1495312  0.1454814   1.028  0.30403
## KC               0.2890362  0.1418348   2.038  0.04157 *
## MIA              0.1744456  0.1453265   1.200  0.23000
## MINN             0.0422303  0.1453624   0.291  0.77142
## NE              -0.1003492  0.1427301  -0.703  0.48202
## NOR              0.0401211  0.1434631   0.280  0.77974
## NYG              0.0559795  0.1437887   0.389  0.69704
## NYJ             -0.0712821  0.1436768  -0.496  0.61981
## OAK              0.3080584  0.1415754   2.176  0.02957 *
## PHI              0.2891731  0.1440383   2.008  0.04469 *
## PIT              0.0617740  0.1422070   0.434  0.66400
## SD               0.0402093  0.1452057   0.277  0.78185
## SEA              0.1546337  0.1400911   1.104  0.26968
## STL              0.1255232  0.1452014   0.864  0.38733
## TB               0.1306612  0.1444262   0.905  0.36563
## TEN              0.1541571  0.1423026   1.083  0.27868
## WAS              0.1387607  0.1418664   0.978  0.32803
## avg_recy_plyr    0.0012730  0.0051595   0.247  0.80511
## avg_recy_pos     0.0455069  0.6629656   0.069  0.94528
## avg_recy_team           NA         NA      NA       NA
## avg_rec_plyr     0.0407983  0.0795490   0.513  0.60804
## avg_rec_pos      0.0578795  1.5592560   0.037  0.97039
## avg_rec_team            NA         NA      NA       NA
## avg_trg_plyr    -0.0444539  0.0584234  -0.761  0.44673
## avg_trg_pos     -0.4727549  7.5009835  -0.063  0.94975
## avg_trg_team            NA         NA      NA       NA
## avg_rectd_plyr   0.1949186  0.2759456   0.706  0.47996
## avg_rectd_pos    2.0900415 34.3634901   0.061  0.95150
## avg_rectd_team          NA         NA      NA       NA
## avg_tdr_plyr     0.0655688  0.3401369   0.193  0.84714
## avg_tdr_pos     -1.4234774 16.8225062  -0.085  0.93257
## avg_tdr_team            NA         NA      NA       NA
## avg_rbra_plyr    0.0121186  0.0328966   0.368  0.71259
## avg_rbra_pos     0.0686059  1.1998429   0.057  0.95440
## avg_rbra_team           NA         NA      NA       NA
## avg_rbry_plyr   -0.0028241  0.0076429  -0.370  0.71175
```

```
## avg_rbry_pos     0.0016399  0.1095271   0.015  0.98805
## avg_rbry_team           NA         NA      NA       NA
## avg_fuml_plyr   -0.0296078  0.4070626  -0.073  0.94202
## avg_fuml_pos    -0.1299245  4.8053254  -0.027  0.97843
## avg_fuml_team           NA         NA      NA       NA
## avg_qbpy_plyr    0.0016497  0.0064475   0.256  0.79806
## avg_qbpy_pos            NA         NA      NA       NA
## avg_qbpy_team           NA         NA      NA       NA
## avg_qbpa_plyr    0.9930627  0.0504354  19.690  < 2e-16 ***
## avg_qbpa_pos            NA         NA      NA       NA
## avg_qbpa_team           NA         NA      NA       NA
## avg_qbpc_plyr   -0.0151864  0.0908514  -0.167  0.86725
## avg_qbpc_pos            NA         NA      NA       NA
## avg_qbpc_team           NA         NA      NA       NA
## avg_qbints_plyr -0.0046062  0.2910631  -0.016  0.98737
## avg_qbints_pos          NA         NA      NA       NA
## avg_qbints_team         NA         NA      NA       NA
## avg_qbtdp_plyr   0.1045517  0.3182860   0.328  0.74255
## avg_qbtdp_pos           NA         NA      NA       NA
## avg_qbtdp_team          NA         NA      NA       NA
## grass_1         -0.0571957  0.0407010  -1.405  0.15995
## bad_weather_1   -0.1133690  0.0809962  -1.400  0.16162
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.509 on 39188 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.8862, Adjusted R-squared:  0.886
## F-statistic:  4693 on 65 and 39188 DF,  p-value: < 2.2e-16
```

Second Run

```r
linRegQBpa2 <- lm(pa~ + cold_weather+ home_team_1+ age+ BUF + CLE+ DAL+ HOU + OAK + PHI+
                    avg_qbpc_plyr+avg_qbints_plyr+avg_qbtdp_plyr, data = nfl_data)

summary(linRegQBpa2)
```

```
##
## Call:
## lm(formula = pa ~ +cold_weather + home_team_1 + age + BUF + CLE +
##     DAL + HOU + OAK + PHI + avg_qbpc_plyr + avg_qbints_plyr +
##     avg_qbtdp_plyr, data = nfl_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -34.033  -0.090   0.004   0.084  39.462
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.473487   0.149471   3.168  0.00154 **
## cold_weather    -0.077541   0.042001  -1.846  0.06487 .
## home_team_1     -0.096787   0.037452  -2.584  0.00976 **
## age             -0.016662   0.005603  -2.974  0.00294 **
## BUF              0.338962   0.106254   3.190  0.00142 **
```

```
## CLE              0.339455   0.104471   3.249  0.00116 **
## DAL             -0.306799   0.105571  -2.906  0.00366 **
## HOU              0.261948   0.103574   2.529  0.01144 *
## OAK              0.262125   0.101125   2.592  0.00954 **
## PHI              0.239905   0.105634   2.271  0.02315 *
## avg_qbpc_plyr    1.591223   0.022710  70.068  < 2e-16 ***
## avg_qbints_plyr  3.387676   0.236246  14.340  < 2e-16 ***
## avg_qbtdp_plyr  -1.657334   0.230755  -7.182 6.98e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.53 on 39242 degrees of freedom
## Multiple R-squared:  0.8846, Adjusted R-squared:  0.8846
## F-statistic: 2.507e+04 on 12 and 39242 DF,  p-value: < 2.2e-16
```

R2 is strong. This stat is the catalyst Ints, completions, TD's. You have to attempt a pass in order to achieve a stat in any of these three categories.

Testing the data, we see that the training set and the test set are similar. The model seems to hold up through testing

```r
PaPredicted <- predict(linRegQBpa2, newdata = TestRecy)

SSEpa <- sum((PaPredicted - TestRecy$pa)^2)
SSTpa <- sum((mean(nfl_data$pa)-TestRecy$pa)^2)
r2_pa <- 1 - SSEpa/SSTpa
r2_pa
```

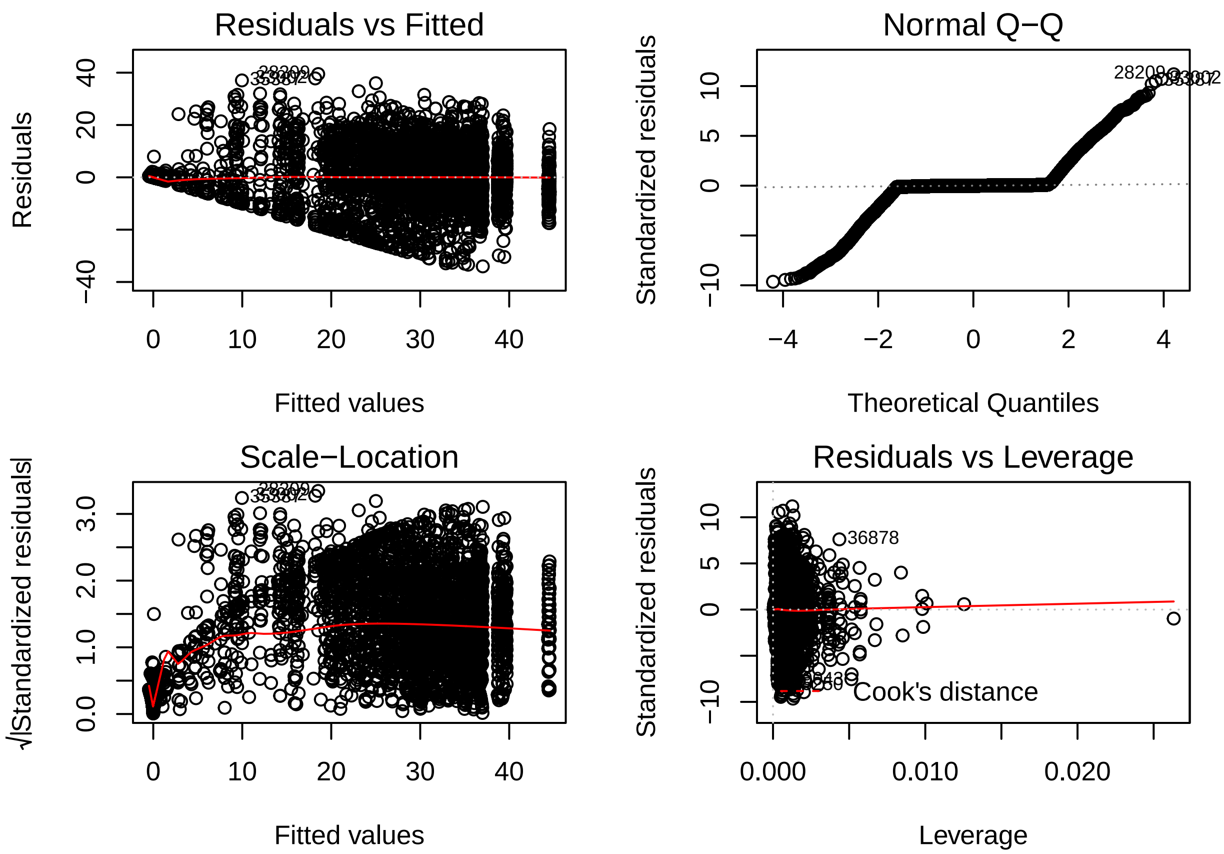
```
## [1] 0.8802769
```

```r
rmse_pa <- sqrt(SSEpa/nrow(TestRecy))
rmse_pa
```

```
## [1] 3.616816
```

Regression plots:

```r
par(mar = c(4, 4, 2, 2), mfrow = c(2, 2))
plot(linRegQBpa2, which = c(1:3,5))
```



Very similar patterns to completions and yards. Some strong tails, so the data may not be normally distributed

Summary statistics:

```
confint(linRegQBpa2)
```

```
##                      2.5 %       97.5 %
## (Intercept)       0.18052104  0.766453936
## cold_weather     -0.15986402  0.004781125
## home_team_1      -0.17019348 -0.023380897
## age              -0.02764332 -0.005679819
## BUF               0.13070135  0.547222924
## CLE               0.13469044  0.544220500
## DAL              -0.51372112 -0.099876632
## HOU               0.05894151  0.464955045
## OAK               0.06391646  0.460332555
## PHI               0.03285997  0.446949481
## avg_qbpc_plyr     1.54671169  1.635735238
## avg_qbints_plyr   2.92462831  3.850722757
## avg_qbtdp_plyr   -2.10961984 -1.205049136
```

```
coef(summary(linRegQBpa2))
```

```
##                    Estimate  Std. Error   t value     Pr(>|t|)
## (Intercept)       0.47348749 0.149470813  3.167759 1.537366e-03
## cold_weather     -0.07754145 0.042000790 -1.846190 6.487211e-02
## home_team_1      -0.09678719 0.037451724 -2.584319 9.760735e-03
## age              -0.01666157 0.005602863 -2.973760 2.943543e-03
```

```
## BUF              0.33896214 0.106254180  3.190106 1.423322e-03
## CLE              0.33945547 0.104470651  3.249290 1.157907e-03
## DAL             -0.30679887 0.105571256 -2.906083 3.661885e-03
## HOU              0.26194828 0.103573590  2.529103 1.143933e-02
## OAK              0.26212451 0.101125295  2.592077 9.543389e-03
## PHI              0.23990473 0.105633761  2.271099 2.314638e-02
## avg_qbpc_plyr    1.59122347 0.022709805 70.067686 0.000000e+00
## avg_qbints_plyr  3.38767553 0.236245639 14.339632 1.622229e-46
## avg_qbtdp_plyr  -1.65733449 0.230754954 -7.182227 6.980186e-13
```

```r
anova(linRegQBpa2)
```

```
## Analysis of Variance Table
##
## Response: pa
##                    Df  Sum Sq Mean Sq    F value    Pr(>F)
## cold_weather        1      89      89 7.1608e+00 0.0074544 **
## home_team_1         1     235     235 1.8839e+01 1.426e-05 ***
## age                 1  215231  215231 1.7271e+04 < 2.2e-16 ***
## BUF                 1       3       3 2.0560e-01 0.6502132
## CLE                 1     168     168 1.3463e+01 0.0002436 ***
## DAL                 1       0       0 3.5000e-03 0.9530714
## HOU                 1       1       1 9.6400e-02 0.7562267
## OAK                 1     143     143 1.1498e+01 0.0006976 ***
## PHI                 1     703     703 5.6443e+01 5.908e-14 ***
## avg_qbpc_plyr       1 3527871 3527871 2.8308e+05 < 2.2e-16 ***
## avg_qbints_plyr     1    4482    4482 3.5967e+02 < 2.2e-16 ***
## avg_qbtdp_plyr      1     643     643 5.1584e+01 6.980e-13 ***
## Residuals       39242  489047      12
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Rushing Stats

Rushing Yards

```r
linRegRushYd <- lm(ry ~ height+ weight+cold_weather + hot_weather + home_team_1+ temp+
                   age+ forty1 + vertical1 + shuttle1+ cone1 + ARI + ATL + BAL + BUF + CAR + CHI+
                   CIN + CLE + DAL + DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR + NYC
                   NYJ + OAK + PHI + PIT +SD + SEA + STL + TB + TEN + WAS + avg_recy_plyr+avg_recy_pos +
                   avg_recy_team + avg_rec_plyr +avg_rec_pos + avg_rec_team +avg_trg_plyr + avg_trg_pos
                   avg_trg_team + avg_rectd_plyr + avg_rectd_pos +avg_rectd_team+
                   avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
                   avg_rbra_plyr + avg_rbra_pos +avg_rbra_team +
                   avg_rbry_plyr + avg_rbry_pos +avg_rbry_team +
                   avg_fuml_plyr + avg_fuml_pos +avg_fuml_team +
                   avg_qbpy_plyr + avg_qbpy_pos +avg_qbpy_team +
                   avg_qbpa_plyr + avg_qbpa_pos +avg_qbpa_team+
                   avg_qbpc_plyr + avg_qbpc_pos +avg_qbpc_team +
                   avg_qbints_plyr + avg_qbints_pos +avg_qbints_team +
                   avg_qbtdp_plyr + avg_qbtdp_pos +avg_qbtdp_team +
                   grass_1 + bad_weather_1 , data = nfl_data)
```

```r
summary(linRegRushYd)
```

```
##
## Call:
## lm(formula = ry ~ height + weight + cold_weather + hot_weather +
##     home_team_1 + temp + age + forty1 + vertical1 + shuttle1 +
##     cone1 + ARI + ATL + BAL + BUF + CAR + CHI + CIN + CLE + DAL +
##     DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE +
##     NOR + NYG + NYJ + OAK + PHI + PIT + SD + SEA + STL + TB +
##     TEN + WAS + avg_recy_plyr + avg_recy_pos + avg_recy_team +
##     avg_rec_plyr + avg_rec_pos + avg_rec_team + avg_trg_plyr +
##     avg_trg_pos + avg_trg_team + avg_rectd_plyr + avg_rectd_pos +
##     avg_rectd_team + avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
##     avg_rbra_plyr + avg_rbra_pos + avg_rbra_team + avg_rbry_plyr +
##     avg_rbry_pos + avg_rbry_team + avg_fuml_plyr + avg_fuml_pos +
##     avg_fuml_team + avg_qbpy_plyr + avg_qbpy_pos + avg_qbpy_team +
##     avg_qbpa_plyr + avg_qbpa_pos + avg_qbpa_team + avg_qbpc_plyr +
##     avg_qbpc_pos + avg_qbpc_team + avg_qbints_plyr + avg_qbints_pos +
##     avg_qbints_team + avg_qbtdp_plyr + avg_qbtdp_pos + avg_qbtdp_team +
##     grass_1 + bad_weather_1, data = nfl_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -87.364  -2.115  -0.380   0.720 180.182
##
## Coefficients: (18 not defined because of singularities)
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.578484   6.764050   0.529  0.59678
## height           -0.017689   0.058230  -0.304  0.76130
## weight            0.003632   0.007794   0.466  0.64125
## cold_weather      0.491723   0.351489   1.399  0.16183
## hot_weather      -0.306673   1.389003  -0.221  0.82526
## home_team_1       0.564301   0.183223   3.080  0.00207 **
## temp             -0.005313   0.010014  -0.530  0.59577
## age              -0.142589   0.027447  -5.195 2.06e-07 ***
## forty1           -0.008062   0.925761  -0.009  0.99305
## vertical1         0.010557   0.037763   0.280  0.77982
## shuttle1         -0.469846   0.803581  -0.585  0.55876
## cone1             0.398631   0.584641   0.682  0.49534
## ARI              -0.263148   0.669360  -0.393  0.69422
## ATL              -0.272282   0.669772  -0.407  0.68436
## BAL              -0.050796   0.650166  -0.078  0.93773
## BUF               0.420754   0.676893   0.622  0.53421
## CAR              -0.448979   0.662416  -0.678  0.49791
## CHI              -0.394026   0.676815  -0.582  0.56045
## CIN              -0.338785   0.672012  -0.504  0.61417
## CLE              -0.070252   0.664992  -0.106  0.91587
## DAL               0.187743   0.678448   0.277  0.78199
## DEN               0.822825   0.652493   1.261  0.20730
## DET               0.326290   0.673440   0.485  0.62802
## GB               -0.316695   0.656435  -0.482  0.62949
## HOU               0.847586   0.666258   1.272  0.20332
## IND              -0.875338   0.666300  -1.314  0.18894
## JAC              -0.354739   0.675188  -0.525  0.59931
```

```
## KC               -0.022533   0.658264  -0.034  0.97269
## MIA               0.316886   0.674469   0.470  0.63848
## MINN             -0.305456   0.674636  -0.453  0.65072
## NE                0.278863   0.662419   0.421  0.67377
## NOR              -0.425794   0.665821  -0.640  0.52250
## NYG              -0.187180   0.667332  -0.280  0.77910
## NYJ              -1.071021   0.666813  -1.606  0.10824
## OAK               0.092736   0.657060   0.141  0.88776
## PHI              -0.825934   0.668490  -1.236  0.21664
## PIT              -0.615320   0.659991  -0.932  0.35118
## SD                0.330104   0.673908   0.490  0.62425
## SEA              -1.043542   0.650171  -1.605  0.10850
## STL               0.674220   0.673888   1.000  0.31708
## TB                0.284724   0.670290   0.425  0.67100
## TEN              -0.064741   0.660435  -0.098  0.92191
## WAS               0.462936   0.658410   0.703  0.48199
## avg_recy_plyr     0.001110   0.023946   0.046  0.96304
## avg_recy_pos      0.354956   3.076863   0.115  0.90816
## avg_recy_team           NA         NA      NA       NA
## avg_rec_plyr      0.039312   0.369192   0.106  0.91520
## avg_rec_pos       0.537955   7.236601   0.074  0.94074
## avg_rec_team            NA         NA      NA       NA
## avg_trg_plyr      0.025126   0.271147   0.093  0.92617
## avg_trg_pos      -3.805546  34.812515  -0.109  0.91295
## avg_trg_team            NA         NA      NA       NA
## avg_rectd_plyr   -0.388475   1.280680  -0.303  0.76164
## avg_rectd_pos    14.486851 159.483024   0.091  0.92762
## avg_rectd_team          NA         NA      NA       NA
## avg_tdr_plyr     -0.124584   1.578596  -0.079  0.93710
## avg_tdr_pos      -7.834022  78.074263  -0.100  0.92007
## avg_tdr_team            NA         NA      NA       NA
## avg_rbra_plyr     0.020334   0.152675   0.133  0.89405
## avg_rbra_pos      1.036269   5.568543   0.186  0.85237
## avg_rbra_team           NA         NA      NA       NA
## avg_rbry_plyr     0.995195   0.035471  28.057  < 2e-16 ***
## avg_rbry_pos     -0.138350   0.508322  -0.272  0.78549
## avg_rbry_team           NA         NA      NA       NA
## avg_fuml_plyr    -0.261073   1.889202  -0.138  0.89009
## avg_fuml_pos     -2.956660  22.301804  -0.133  0.89453
## avg_fuml_team           NA         NA      NA       NA
## avg_qbpy_plyr     0.005984   0.029923   0.200  0.84150
## avg_qbpy_pos            NA         NA      NA       NA
## avg_qbpy_team           NA         NA      NA       NA
## avg_qbpa_plyr    -0.139593   0.234074  -0.596  0.55094
## avg_qbpa_pos            NA         NA      NA       NA
## avg_qbpa_team           NA         NA      NA       NA
## avg_qbpc_plyr     0.092511   0.421647   0.219  0.82634
## avg_qbpc_pos            NA         NA      NA       NA
## avg_qbpc_team           NA         NA      NA       NA
## avg_qbints_plyr   0.880240   1.350841   0.652  0.51465
## avg_qbints_pos          NA         NA      NA       NA
## avg_qbints_team         NA         NA      NA       NA
## avg_qbtdp_plyr    0.553864   1.477185   0.375  0.70770
## avg_qbtdp_pos           NA         NA      NA       NA
```

```
## avg_qbtdp_team          NA         NA      NA       NA
## grass_1           -0.282648   0.188896  -1.496  0.13458
## bad_weather_1      0.471768   0.375908   1.255  0.20948
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.29 on 39188 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.5818, Adjusted R-squared:  0.5811
## F-statistic: 838.8 on 65 and 39188 DF,  p-value: < 2.2e-16
```

Second Run

```r
linRegRushYd2 <- lm(ry ~ home_team_1 + age +avg_rbry_plyr, data = nfl_data)

summary(linRegRushYd2)
```

```
##
## Call:
## lm(formula = ry ~ home_team_1 + age + avg_rbry_plyr, data = nfl_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -86.566  -1.983  -0.307   0.403 179.914
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.056638   0.672507   4.545 5.51e-06 ***
## home_team_1    0.533574   0.168384   3.169  0.00153 **
## age           -0.122796   0.024924  -4.927 8.39e-07 ***
## avg_rbry_plyr  0.998139   0.004301 232.091  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.28 on 39251 degrees of freedom
## Multiple R-squared:  0.5813, Adjusted R-squared:  0.5812
## F-statistic: 1.816e+04 on 3 and 39251 DF,  p-value: < 2.2e-16
```

Testing the data, we see that the training set and the test set are similar. The model seems to hold up through testing

```r
RushydsPredicted <- predict(linRegRushYd2, newdata = TestRecy)

SSEruyd <- sum((RushydsPredicted - TestRecy$ry)^2)
SSTruyd <- sum((mean(nfl_data$ry)-TestRecy$ry)^2)
r2_ruyd <- 1 - SSEruyd/SSTruyd
r2_ruyd
```

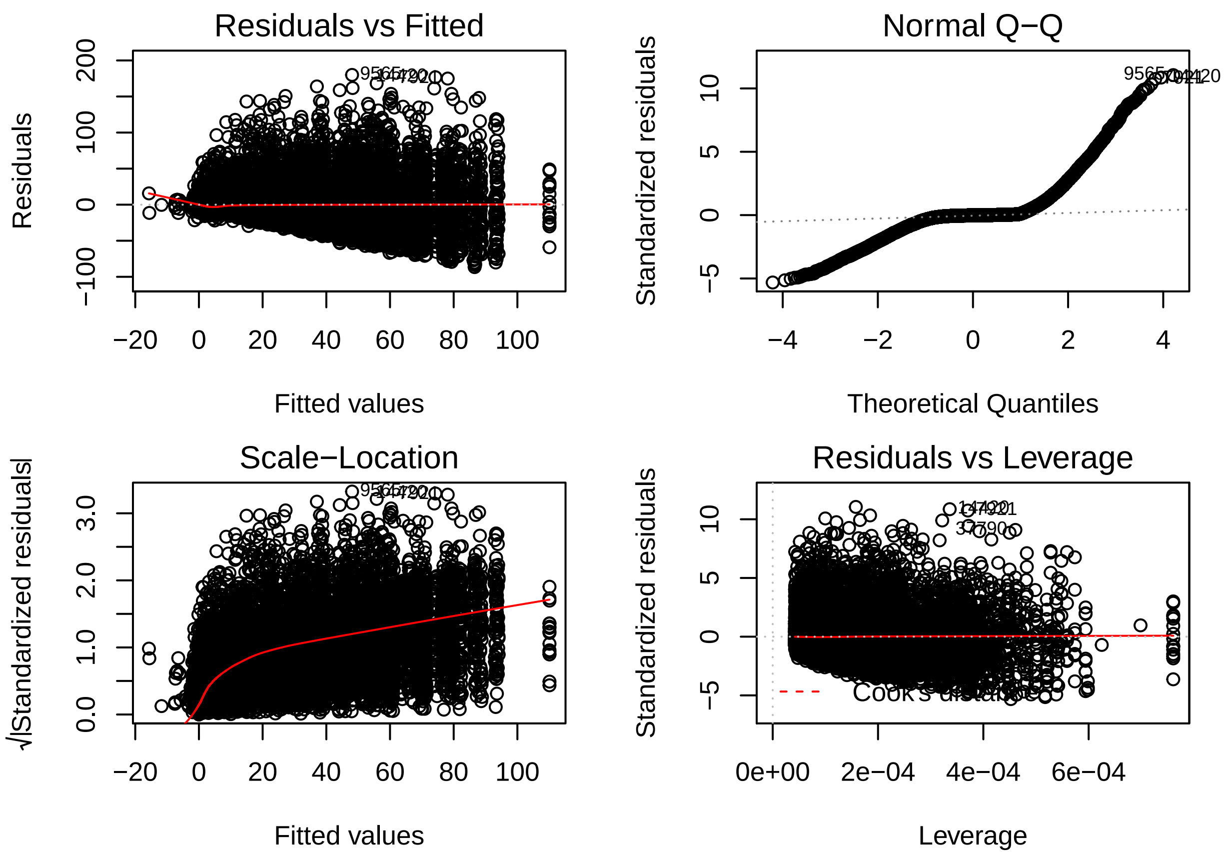
```
## [1] 0.5855079
```

```r
rmse_ruyd <- sqrt(SSEruyd/nrow(TestRecy))
rmse_ruyd
```

```
## [1] 16.24423
```

Regression plots:

```r
par(mar = c(4, 4, 2, 2), mfrow = c(2, 2))
plot(linRegRushYd2, which = c(1:3,5))
```



Summary statistics:

```r
confint(linRegRushYd2)
```

```
##                     2.5 %      97.5 %
## (Intercept)     1.7385074  4.37476874
## home_team_1     0.2035377  0.86361128
## age            -0.1716488 -0.07394389
## avg_rbry_plyr   0.9897099  1.00656857
```

```r
coef(summary(linRegRushYd2))
```

```
##                  Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)     3.0566380 0.672507279   4.545137 5.506302e-06
## home_team_1     0.5335745 0.168384017   3.168795 1.531895e-03
## age            -0.1227964 0.024924425  -4.926748 8.394909e-07
## avg_rbry_plyr   0.9981392 0.004300635 232.091137 0.000000e+00
```

```r
anova(linRegRushYd2)
```

```
## Analysis of Variance Table
##
## Response: ry
##                  Df    Sum Sq  Mean Sq   F value    Pr(>F)
## home_team_1       1      5687     5687    21.451 3.641e-06 ***
```

```
## age                1   158667   158667   598.451 < 2.2e-16 ***
## avg_rbry_plyr      1 14281561 14281561 53866.296 < 2.2e-16 ***
## Residuals      39251 10406610      265
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Rushing attempts

```
linRegRushAtt <- lm(ra ~ height+ weight+cold_weather + hot_weather + home_team_1+ temp+
                     age+ forty1 + vertical1 + shuttle1+ cone1 + ARI + ATL + BAL + BUF + CAR + CHI+
                     CIN + CLE + DAL + DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR + N
                     NYJ + OAK + PHI + PIT +SD + SEA + STL + TB + TEN + WAS + avg_recy_plyr+avg_recy_pos
                     avg_recy_team + avg_rec_plyr +avg_rec_pos + avg_rec_team +avg_trg_plyr + avg_trg_po
                     avg_trg_team + avg_rectd_plyr + avg_rectd_pos +avg_rectd_team+
                      avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
                     avg_rbra_plyr + avg_rbra_pos +avg_rbra_team +
                     avg_rbry_plyr + avg_rbry_pos +avg_rbry_team +
                     avg_fuml_plyr + avg_fuml_pos +avg_fuml_team +
                     avg_qbpy_plyr + avg_qbpy_pos +avg_qbpy_team +
                     avg_qbpa_plyr + avg_qbpa_pos +avg_qbpa_team+
                     avg_qbpc_plyr + avg_qbpc_pos +avg_qbpc_team +
                     avg_qbints_plyr + avg_qbints_pos +avg_qbints_team +
                     avg_qbtdp_plyr + avg_qbtdp_pos +avg_qbtdp_team +
                     grass_1 + bad_weather_1 , data = nfl_data)

summary(linRegRushAtt)
```

```
##
## Call:
## lm(formula = ra ~ height + weight + cold_weather + hot_weather +
##     home_team_1 + temp + age + forty1 + vertical1 + shuttle1 +
##     cone1 + ARI + ATL + BAL + BUF + CAR + CHI + CIN + CLE + DAL +
##     DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE +
##     NOR + NYG + NYJ + OAK + PHI + PIT + SD + SEA + STL + TB +
##     TEN + WAS + avg_recy_plyr + avg_recy_pos + avg_recy_team +
##     avg_rec_plyr + avg_rec_pos + avg_rec_team + avg_trg_plyr +
##     avg_trg_pos + avg_trg_team + avg_rectd_plyr + avg_rectd_pos +
##     avg_rectd_team + avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
##     avg_rbra_plyr + avg_rbra_pos + avg_rbra_team + avg_rbry_plyr +
##     avg_rbry_pos + avg_rbry_team + avg_fuml_plyr + avg_fuml_pos +
##     avg_fuml_team + avg_qbpy_plyr + avg_qbpy_pos + avg_qbpy_team +
##     avg_qbpa_plyr + avg_qbpa_pos + avg_qbpa_team + avg_qbpc_plyr +
##     avg_qbpc_pos + avg_qbpc_team + avg_qbints_plyr + avg_qbints_pos +
##     avg_qbints_team + avg_qbtdp_plyr + avg_qbtdp_pos + avg_qbtdp_team +
##     grass_1 + bad_weather_1, data = nfl_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.6660  -0.3156  -0.0595   0.1608  28.7883
##
## Coefficients: (18 not defined because of singularities)
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.5872029  1.1857743   0.495   0.6205
```

```
## height          -0.0019453  0.0102080  -0.191   0.8489
## weight           0.0004563  0.0013663   0.334   0.7384
## cold_weather     0.0703223  0.0616179   1.141   0.2538
## hot_weather     -0.0420763  0.2434997  -0.173   0.8628
## home_team_1      0.0863209  0.0321200   2.687   0.0072 **
## temp            -0.0016505  0.0017556  -0.940   0.3472
## age             -0.0190196  0.0048116  -3.953 7.74e-05 ***
## forty1          -0.0440550  0.1622909  -0.271   0.7860
## vertical1        0.0012397  0.0066201   0.187   0.8514
## shuttle1        -0.0580941  0.1408721  -0.412   0.6801
## cone1            0.0639659  0.1024906   0.624   0.5326
## ARI             -0.0499362  0.1173423  -0.426   0.6704
## ATL             -0.0686928  0.1174146  -0.585   0.5585
## BAL              0.0093193  0.1139775   0.082   0.9348
## BUF              0.0911284  0.1186630   0.768   0.4425
## CAR             -0.1076917  0.1161251  -0.927   0.3537
## CHI             -0.0790013  0.1186494  -0.666   0.5055
## CIN              0.0091122  0.1178073   0.077   0.9383
## CLE              0.0862227  0.1165766   0.740   0.4595
## DAL              0.0063560  0.1189355   0.053   0.9574
## DEN              0.1610148  0.1143854   1.408   0.1592
## DET              0.0943077  0.1180577   0.799   0.4244
## GB              -0.0529443  0.1150766  -0.460   0.6455
## HOU              0.1787931  0.1167985   1.531   0.1258
## IND             -0.1639191  0.1168060  -1.403   0.1605
## JAC             -0.0063967  0.1183641  -0.054   0.9569
## KC              -0.0294860  0.1153972  -0.256   0.7983
## MIA              0.0954888  0.1182380   0.808   0.4193
## MINN            -0.0754425  0.1182673  -0.638   0.5235
## NE               0.0748271  0.1161256   0.644   0.5193
## NOR             -0.1137753  0.1167220  -0.975   0.3297
## NYG              0.0340775  0.1169869   0.291   0.7708
## NYJ             -0.1570734  0.1168959  -1.344   0.1791
## OAK             -0.0001271  0.1151861  -0.001   0.9991
## PHI             -0.1655449  0.1171899  -1.413   0.1578
## PIT             -0.1349978  0.1157000  -1.167   0.2433
## SD               0.1038228  0.1181397   0.879   0.3795
## SEA             -0.1672506  0.1139785  -1.467   0.1423
## STL              0.1643070  0.1181362   1.391   0.1643
## TB               0.0757634  0.1175055   0.645   0.5191
## TEN             -0.0283452  0.1157778  -0.245   0.8066
## WAS              0.1134681  0.1154229   0.983   0.3256
## avg_recy_plyr    0.0005919  0.0041978   0.141   0.8879
## avg_recy_pos     0.0591962  0.5393906   0.110   0.9126
## avg_recy_team           NA         NA      NA       NA
## avg_rec_plyr     0.0195484  0.0647213   0.302   0.7626
## avg_rec_pos      0.0852950  1.2686150   0.067   0.9464
## avg_rec_team            NA         NA      NA       NA
## avg_trg_plyr    -0.0088855  0.0475335  -0.187   0.8517
## avg_trg_pos     -0.6436794  6.1028208  -0.105   0.9160
## avg_trg_team            NA         NA      NA       NA
## avg_rectd_plyr  -0.0506339  0.2245101  -0.226   0.8216
## avg_rectd_pos    2.6983924 27.9582302   0.097   0.9231
## avg_rectd_team          NA         NA      NA       NA
```

```
## avg_tdr_plyr     0.0195341  0.2767363   0.071   0.9437
## avg_tdr_pos     -1.4282243 13.6868374  -0.104   0.9169
## avg_tdr_team            NA         NA      NA       NA
## avg_rbra_plyr    0.9983045  0.0267648  37.299  < 2e-16 ***
## avg_rbra_pos     0.1671713  0.9761955   0.171   0.8640
## avg_rbra_team           NA         NA      NA       NA
## avg_rbry_plyr    0.0001258  0.0062182   0.020   0.9839
## avg_rbry_pos    -0.0202284  0.0891115  -0.227   0.8204
## avg_rbry_team           NA         NA      NA       NA
## avg_fuml_plyr   -0.0632465  0.3311872  -0.191   0.8486
## avg_fuml_pos    -0.4044266  3.9096260  -0.103   0.9176
## avg_fuml_team           NA         NA      NA       NA
## avg_qbpy_plyr    0.0006534  0.0052457   0.125   0.9009
## avg_qbpy_pos            NA         NA      NA       NA
## avg_qbpy_team           NA         NA      NA       NA
## avg_qbpa_plyr   -0.0241980  0.0410344  -0.590   0.5554
## avg_qbpa_pos            NA         NA      NA       NA
## avg_qbpa_team           NA         NA      NA       NA
## avg_qbpc_plyr    0.0192677  0.0739170   0.261   0.7944
## avg_qbpc_pos            NA         NA      NA       NA
## avg_qbpc_team           NA         NA      NA       NA
## avg_qbints_plyr  0.1512533  0.2368097   0.639   0.5230
## avg_qbints_pos          NA         NA      NA       NA
## avg_qbints_team         NA         NA      NA       NA
## avg_qbtdp_plyr   0.1031010  0.2589584   0.398   0.6905
## avg_qbtdp_pos           NA         NA      NA       NA
## avg_qbtdp_team          NA         NA      NA       NA
## grass_1         -0.0103108  0.0331144  -0.311   0.7555
## bad_weather_1    0.0139671  0.0658987   0.212   0.8321
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.855 on 39188 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.7084, Adjusted R-squared:  0.708
## F-statistic:  1465 on 65 and 39188 DF,  p-value: < 2.2e-16
```

Second run

```r
linRegRushAtt2 <- lm(ra ~ home_team_1 + age +avg_rbra_plyr, data = nfl_data)

summary(linRegRushAtt2)
```

```
##
## Call:
## lm(formula = ra ~ home_team_1 + age + avg_rbra_plyr, data = nfl_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.5145  -0.2716  -0.0490   0.0784  28.7009
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     0.415712   0.117712   3.532 0.000414 ***
```

```
## home_team_1    0.094509   0.029522   3.201 0.001369 **
## age           -0.017039   0.004365  -3.903  9.5e-05 ***
## avg_rbra_plyr  0.999076   0.003250 307.402  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.855 on 39251 degrees of freedom
## Multiple R-squared:  0.708,  Adjusted R-squared:  0.708
## F-statistic: 3.172e+04 on 3 and 39251 DF,  p-value: < 2.2e-16
```

Testing the data, we see that the training set and the test set are similar. The model seems to hold up through testing

```
RushattPredicted <- predict(linRegRushAtt2, newdata = TestRecy)

SSEruatt <- sum((RushattPredicted - TestRecy$ra)^2)
SSTruatt <- sum((mean(nfl_data$ra)-TestRecy$ra)^2)
r2_ruatt <- 1 - SSEruatt/SSTruatt
r2_ruatt
```

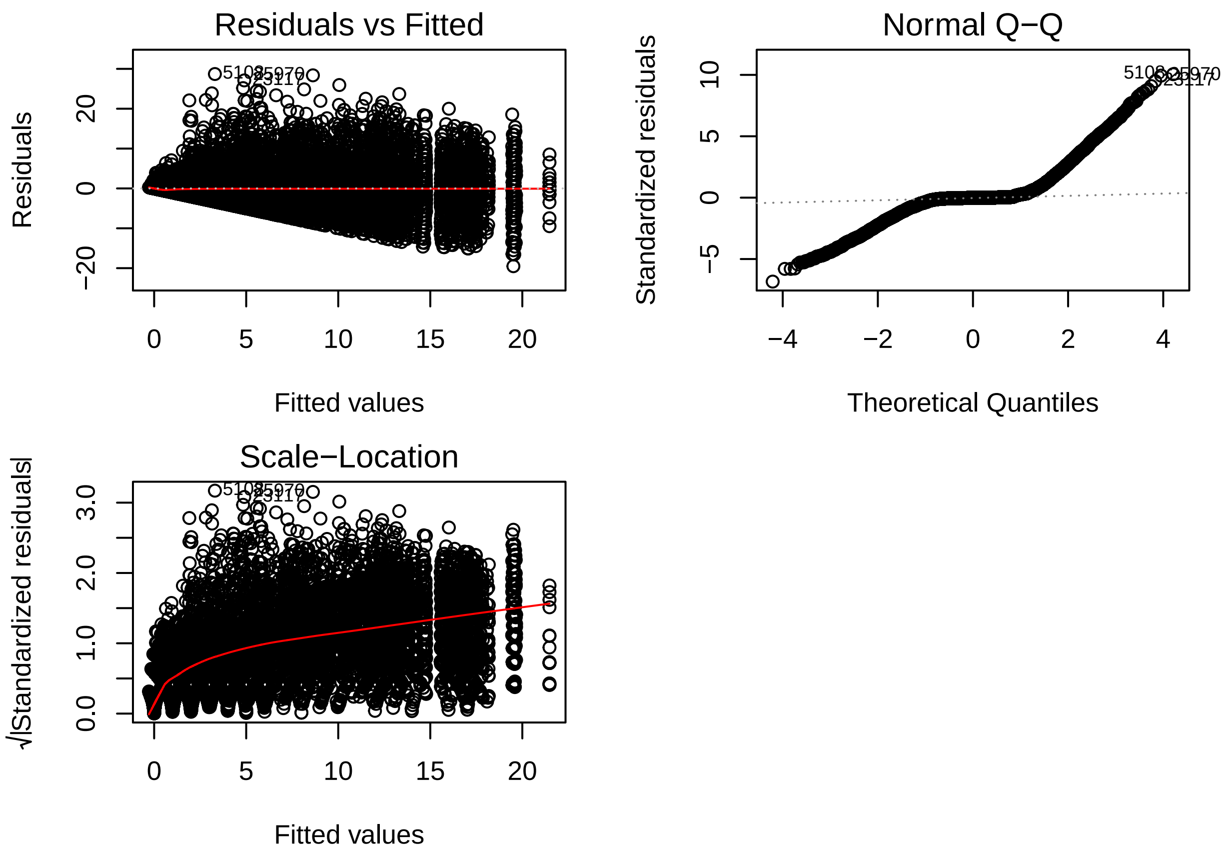
```
## [1] 0.7093207
```

```
rmse_ruatt <- sqrt(SSEruatt/nrow(TestRecy))
rmse_ruatt
```

```
## [1] 2.880426
```

Regression plots:

```
par(mar = c(4, 4, 2, 2), mfrow = c(2, 2))
plot(linRegRushAtt2, which = c(1:3,2))
```



Summary statistics:

```r
confint(linRegRushAtt2)
```

```
##                      2.5 %       97.5 %
## (Intercept)     0.18499368  0.646429890
## home_team_1     0.03664507  0.152372636
## age            -0.02559510 -0.008483082
## avg_rbra_plyr   0.99270599  1.005446400
```

```r
coef(summary(linRegRushAtt2))
```

```
##                   Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)     0.41571178 0.117711852   3.531605 4.135191e-04
## home_team_1     0.09450885 0.029521970   3.201306 1.369152e-03
## age            -0.01703909 0.004365257  -3.903342 9.503174e-05
## avg_rbra_plyr   0.99907620 0.003250063 307.402080 0.000000e+00
```

```r
anova(linRegRushAtt2)
```

```
## Analysis of Variance Table
##
## Response: ra
##                  Df Sum Sq Mean Sq   F value    Pr(>F)
## home_team_1       1    150     150    18.383 1.811e-05 ***
## age               1   5285    5285   648.516 < 2.2e-16 ***
## avg_rbra_plyr     1 770145  770145 94496.039 < 2.2e-16 ***
## Residuals     39251 319897       8
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Fumbles

```
linRegFumble <- lm(fuml ~ height+ weight+cold_weather + hot_weather + home_team_1+ temp+
                     age+ forty1 + vertical1 + shuttle1+ cone1 + ARI + ATL + BAL + BUF + CAR + CHI+
                     CIN + CLE + DAL + DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE + NOR + N
                     NYJ + OAK + PHI + PIT +SD + SEA + STL + TB + TEN + WAS + avg_recy_plyr+avg_recy_pos
                     avg_recy_team + avg_rec_plyr +avg_rec_pos + avg_rec_team +avg_trg_plyr + avg_trg_po
                     avg_trg_team + avg_rectd_plyr + avg_rectd_pos +avg_rectd_team+
                     avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
                     avg_rbra_plyr + avg_rbra_pos +avg_rbra_team +
                     avg_rbry_plyr + avg_rbry_pos +avg_rbry_team +
                     avg_fuml_plyr + avg_fuml_pos +avg_fuml_team +
                     avg_qbpy_plyr + avg_qbpy_pos +avg_qbpy_team +
                     avg_qbpa_plyr + avg_qbpa_pos +avg_qbpa_team+
                     avg_qbpc_plyr + avg_qbpc_pos +avg_qbpc_team +
                     avg_qbints_plyr + avg_qbints_pos +avg_qbints_team +
                     avg_qbtdp_plyr + avg_qbtdp_pos +avg_qbtdp_team +
                     grass_1 + bad_weather_1 , data = TrainRecy)

summary(linRegFumble)
```

```
##
## Call:
## lm(formula = fuml ~ height + weight + cold_weather + hot_weather +
##     home_team_1 + temp + age + forty1 + vertical1 + shuttle1 +
##     cone1 + ARI + ATL + BAL + BUF + CAR + CHI + CIN + CLE + DAL +
##     DEN + DET + GB + HOU + IND + JAC + KC + MIA + MINN + NE +
##     NOR + NYG + NYJ + OAK + PHI + PIT + SD + SEA + STL + TB +
##     TEN + WAS + avg_recy_plyr + avg_recy_pos + avg_recy_team +
##     avg_rec_plyr + avg_rec_pos + avg_rec_team + avg_trg_plyr +
##     avg_trg_pos + avg_trg_team + avg_rectd_plyr + avg_rectd_pos +
##     avg_rectd_team + avg_tdr_plyr + avg_tdr_pos + avg_tdr_team +
##     avg_rbra_plyr + avg_rbra_pos + avg_rbra_team + avg_rbry_plyr +
##     avg_rbry_pos + avg_rbry_team + avg_fuml_plyr + avg_fuml_pos +
##     avg_fuml_team + avg_qbpy_plyr + avg_qbpy_pos + avg_qbpy_team +
##     avg_qbpa_plyr + avg_qbpa_pos + avg_qbpa_team + avg_qbpc_plyr +
##     avg_qbpc_pos + avg_qbpc_team + avg_qbints_plyr + avg_qbints_pos +
##     avg_qbints_team + avg_qbtdp_plyr + avg_qbtdp_pos + avg_qbtdp_team +
##     grass_1 + bad_weather_1, data = TrainRecy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5517 -0.0637 -0.0271 -0.0030  3.8444
##
## Coefficients: (18 not defined because of singularities)
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       1.023e-01  1.144e-01   0.894   0.3712
## height            1.163e-04  9.772e-04   0.119   0.9053
## weight            6.497e-05  1.306e-04   0.498   0.6188
## cold_weather      2.225e-03  5.909e-03   0.376   0.7066
```

```
## hot_weather      7.362e-03  2.344e-02   0.314   0.7535
## home_team_1     -1.809e-03  3.089e-03  -0.586   0.5582
## temp            -1.467e-04  1.684e-04  -0.872   0.3835
## age             -9.220e-04  4.634e-04  -1.990   0.0466 *
## forty1          -6.068e-03  1.563e-02  -0.388   0.6978
## vertical1       -3.229e-04  6.385e-04  -0.506   0.6130
## shuttle1        -5.425e-03  1.355e-02  -0.400   0.6890
## cone1           -3.116e-03  9.802e-03  -0.318   0.7506
## ARI             -8.459e-03  1.128e-02  -0.750   0.4535
## ATL             -5.509e-03  1.130e-02  -0.487   0.6260
## BAL             -9.345e-03  1.102e-02  -0.848   0.3964
## BUF             -5.035e-03  1.129e-02  -0.446   0.6555
## CAR             -6.367e-03  1.111e-02  -0.573   0.5667
## CHI             -1.757e-02  1.136e-02  -1.546   0.1221
## CIN             -1.366e-02  1.129e-02  -1.209   0.2266
## CLE             -5.898e-03  1.119e-02  -0.527   0.5981
## DAL             -5.371e-03  1.136e-02  -0.473   0.6362
## DEN             -1.917e-03  1.100e-02  -0.174   0.8616
## DET             -5.591e-03  1.133e-02  -0.493   0.6218
## GB              -3.286e-03  1.103e-02  -0.298   0.7659
## HOU             -3.013e-03  1.118e-02  -0.270   0.7875
## IND             -1.546e-02  1.123e-02  -1.377   0.1686
## JAC             -2.725e-03  1.141e-02  -0.239   0.8112
## KC              -6.731e-04  1.106e-02  -0.061   0.9515
## MIA             -7.989e-04  1.128e-02  -0.071   0.9435
## MINN            -4.886e-03  1.141e-02  -0.428   0.6685
## NE              -1.365e-02  1.113e-02  -1.226   0.2202
## NOR             -1.225e-02  1.114e-02  -1.100   0.2715
## NYG              1.830e-03  1.117e-02   0.164   0.8699
## NYJ             -1.660e-02  1.119e-02  -1.484   0.1378
## OAK             -1.834e-04  1.101e-02  -0.017   0.9867
## PHI              1.883e-03  1.131e-02   0.166   0.8678
## PIT             -9.405e-03  1.111e-02  -0.846   0.3973
## SD              -9.772e-04  1.135e-02  -0.086   0.9314
## SEA              1.396e-03  1.103e-02   0.126   0.8993
## STL             -3.221e-03  1.138e-02  -0.283   0.7771
## TB              -3.668e-03  1.126e-02  -0.326   0.7445
## TEN             -1.051e-02  1.111e-02  -0.946   0.3441
## WAS             -7.122e-03  1.109e-02  -0.642   0.5207
## avg_recy_plyr   -4.532e-05  4.052e-04  -0.112   0.9109
## avg_recy_pos     1.479e-02  5.971e-02   0.248   0.8043
## avg_recy_team           NA         NA      NA       NA
## avg_rec_plyr     2.026e-03  6.222e-03   0.326   0.7447
## avg_rec_pos      4.965e-02  1.325e-01   0.375   0.7078
## avg_rec_team            NA         NA      NA       NA
## avg_trg_plyr    -3.538e-04  4.595e-03  -0.077   0.9386
## avg_trg_pos     -1.676e-01  6.739e-01  -0.249   0.8036
## avg_trg_team            NA         NA      NA       NA
## avg_rectd_plyr  -6.322e-03  2.170e-02  -0.291   0.7707
## avg_rectd_pos    5.204e-01  3.126e+00   0.166   0.8678
## avg_rectd_team          NA         NA      NA       NA
## avg_tdr_plyr    -1.511e-03  2.628e-02  -0.057   0.9542
## avg_tdr_pos     -2.137e-01  1.534e+00  -0.139   0.8892
## avg_tdr_team            NA         NA      NA       NA
```

```
## avg_rbra_plyr    4.571e-03  2.580e-03   1.772   0.0765 .
## avg_rbra_pos     2.474e-02  1.075e-01   0.230   0.8180
## avg_rbra_team           NA         NA      NA       NA
## avg_rbry_plyr   -1.067e-03  5.992e-04  -1.781   0.0749 .
## avg_rbry_pos    -2.188e-03  8.458e-03  -0.259   0.7958
## avg_rbry_team           NA         NA      NA       NA
## avg_fuml_plyr    1.020e+00  3.123e-02  32.673   <2e-16 ***
## avg_fuml_pos    -1.160e-01  4.299e-01  -0.270   0.7874
## avg_fuml_team           NA         NA      NA       NA
## avg_qbpy_plyr    8.927e-04  5.102e-04   1.750   0.0801 .
## avg_qbpy_pos            NA         NA      NA       NA
## avg_qbpy_team           NA         NA      NA       NA
## avg_qbpa_plyr   -3.649e-03  3.906e-03  -0.934   0.3502
## avg_qbpa_pos            NA         NA      NA       NA
## avg_qbpa_team           NA         NA      NA       NA
## avg_qbpc_plyr    4.888e-04  6.994e-03   0.070   0.9443
## avg_qbpc_pos            NA         NA      NA       NA
## avg_qbpc_team           NA         NA      NA       NA
## avg_qbints_plyr -1.231e-02  2.347e-02  -0.525   0.5998
## avg_qbints_pos          NA         NA      NA       NA
## avg_qbints_team         NA         NA      NA       NA
## avg_qbtdp_plyr  -6.108e-02  2.515e-02  -2.429   0.0152 *
## avg_qbtdp_pos           NA         NA      NA       NA
## avg_qbtdp_team          NA         NA      NA       NA
## grass_1         -4.034e-03  3.188e-03  -1.265   0.2057
## bad_weather_1    1.233e-03  6.303e-03   0.196   0.8450
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2294 on 27417 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.09319,    Adjusted R-squared:  0.09104
## F-statistic: 43.35 on 65 and 27417 DF,  p-value: < 2.2e-16
```

Second run

```r
linRegFumble2 <- lm(fuml ~ avg_fuml_plyr + grass_1, data = TrainRecy)

summary(linRegFumble2)
```

```
##
## Call:
## lm(formula = fuml ~ avg_fuml_plyr + grass_1, data = TrainRecy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5379 -0.0648 -0.0266 -0.0013  3.8565
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.001310   0.002230   0.588    0.557
## avg_fuml_plyr  1.013512   0.019207  52.768   <2e-16 ***
## grass_1       -0.002610   0.002767  -0.943    0.346
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2293 on 27481 degrees of freedom
## Multiple R-squared:  0.092,  Adjusted R-squared:  0.09194
## F-statistic:  1392 on 2 and 27481 DF,  p-value: < 2.2e-16
```

Definitely does not seem to be a good predictor

Testing the data, we see that the training set and the test set are similar. The model seems to hold up through testing

```
FumblePredicted <- predict(linRegFumble2, newdata = TestRecy)

SSEfum <- sum((FumblePredicted - TestRecy$fuml)^2)
SSTfum <- sum((mean(nfl_data$fuml)-TestRecy$fuml)^2)
r2_fum <- 1 - SSEfum/SSTfum
r2_fum
```

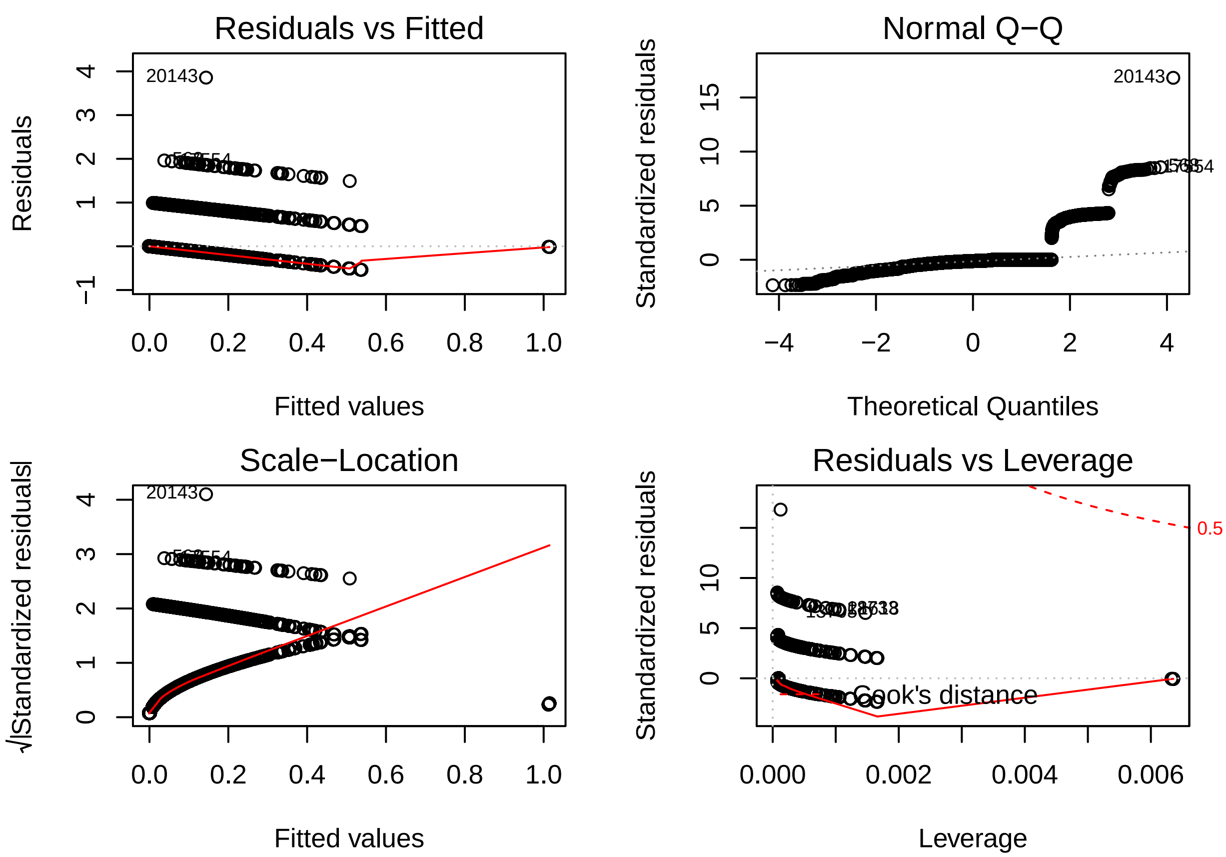
```
## [1] 0.08244378
```

```
rmse_fum <- sqrt(SSEfum/nrow(TestRecy))
rmse_fum
```

```
## [1] 0.2293017
```

Regression plots:

```
par(mar = c(4, 4, 2, 2), mfrow = c(2, 2))
plot(linRegFumble2, which = c(1:3,5))
```



The plots don't seem to confirm that this is a good model for the data

Summary statistics:

```
confint(linRegFumble2)
```

```
##                      2.5 %      97.5 %
## (Intercept)   -0.003061317 0.005682210
## avg_fuml_plyr  0.975864893 1.051158137
## grass_1       -0.008033472 0.002814257
```

```
coef(summary(linRegFumble2))
```

```
##                   Estimate  Std. Error    t value  Pr(>|t|)
## (Intercept)    0.001310447 0.002230434  0.5875298 0.5568528
## avg_fuml_plyr  1.013511515 0.019206967 52.7679099 0.0000000
## grass_1       -0.002609608 0.002767207 -0.9430475 0.3456649
```

```
anova(linRegFumble2)
```

```
## Analysis of Variance Table
##
## Response: fuml
##                  Df  Sum Sq Mean Sq   F value Pr(>F)
## avg_fuml_plyr     1  146.38 146.376 2783.6475 <2e-16 ***
## grass_1           1    0.05   0.047    0.8893 0.3457
## Residuals     27481 1445.07   0.053
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Conclusions

Rushing and passing data seems to be something that linear regression has better predictions for that receiving. This makes sense. RB's and QB's are generally singular players on the field. Both are heavily invested in by teams and are given a lot of touches every game. Receiving is a little more spread out. There are generally at minimum, 3 players in a receiving capacity (excluding the RB), 2 WR and a TE. There can be up to 4 WR on the field, so trying to predict who gets the ball will be harder because it is more uncertain.

RB's are negatively effected by age, the data supports this well known fact, I was glad to see that relationship.

Fumbles and INTs are also going to be hard to predict because they are generally random, but highly dependent upon the player carrying the ball and the defense they are playing against.

The data seems to have some skewness, so I may have to explore other options for predicting.

Going forward, I would like to explore more fields, I have completely ignored the opponents in this analysis, and would like to add them in for the future.

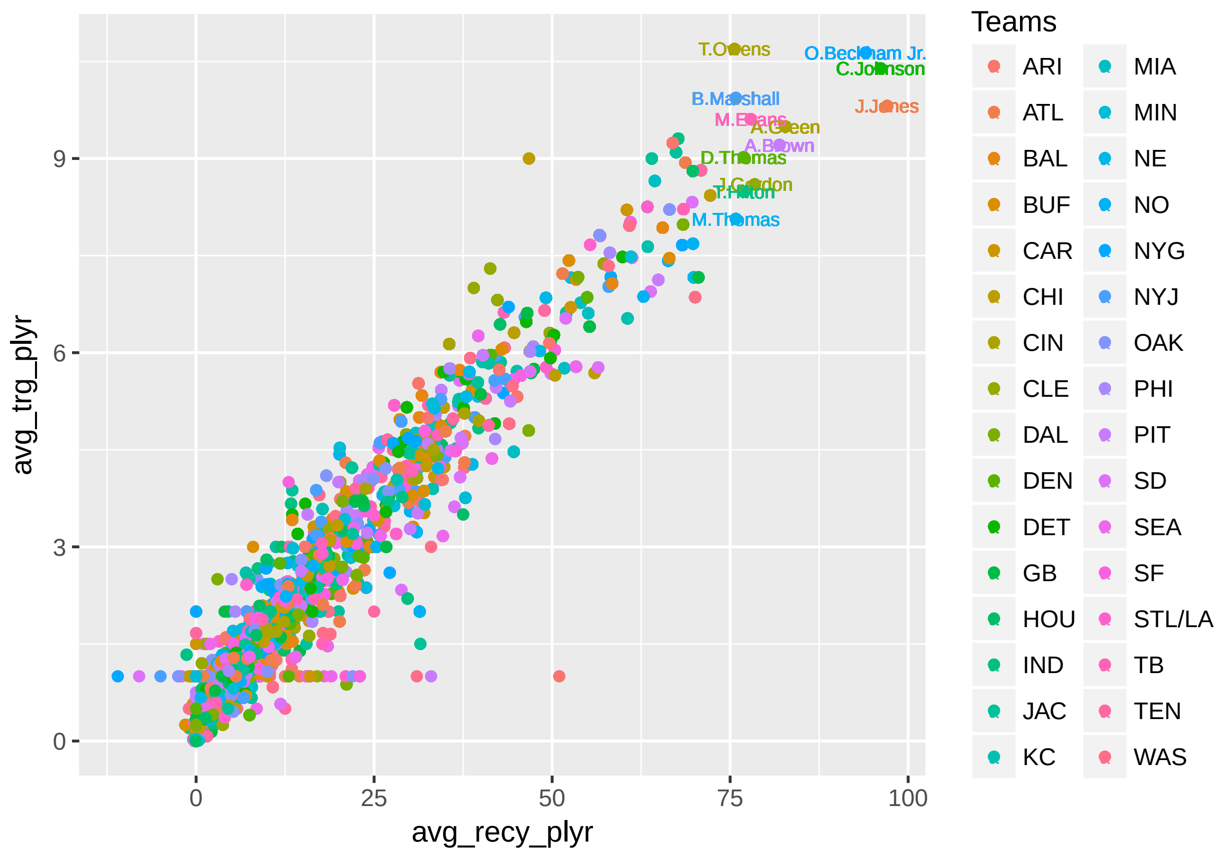
As mentioned above, baseball will be better for predicting because of the nature of the game: One batter vs one pitcher. I would like to build an analysis for baseball based on a similar method as I have used for this.

Appendix

Charts

WR targets by avg yards per player

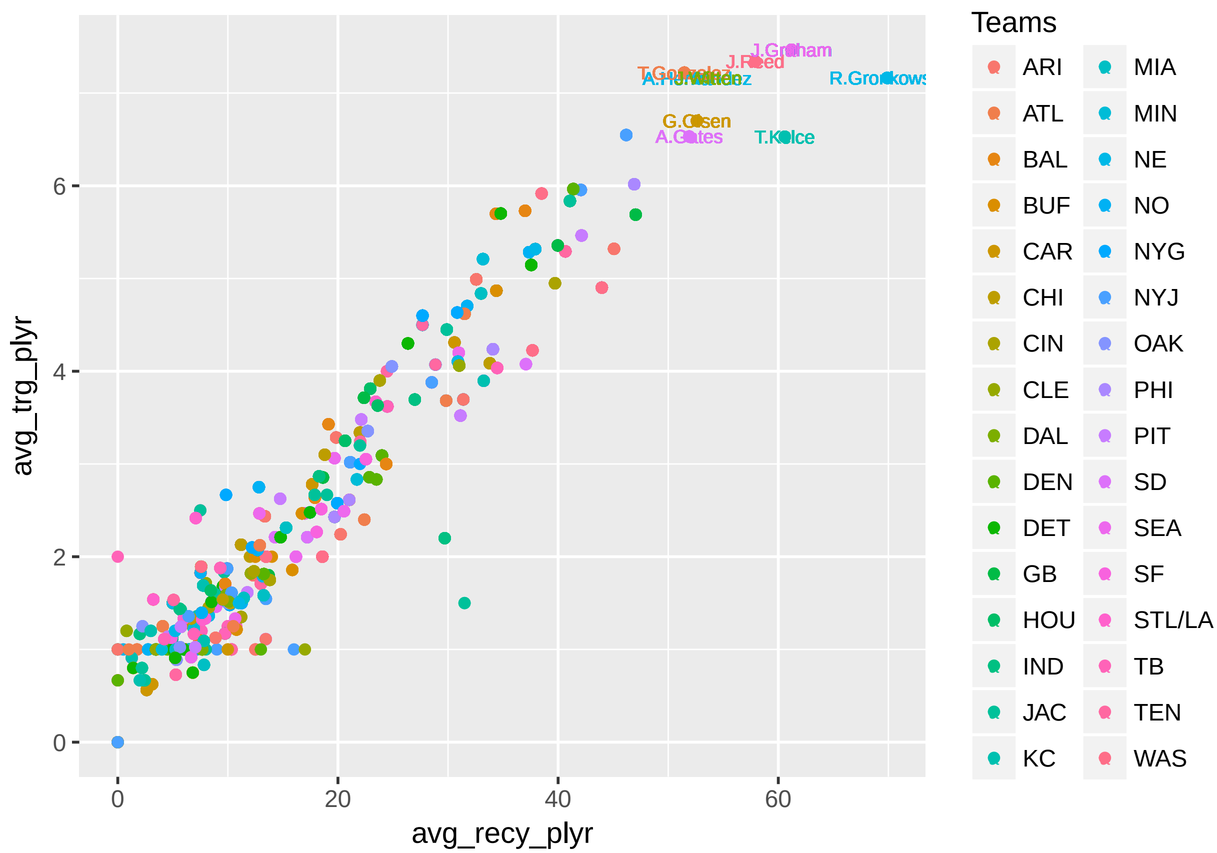
```r
ggplot(data = nfl_data, aes(x = avg_recy_plyr, y = avg_trg_plyr, col = Teams ))+
  geom_point()+
  geom_text(data = subset(nfl_data, avg_recy_plyr > 75), aes(label = pname), size = 2.5)
```



There are few anomalies in this graph, not surprising, the amount of targets correlates with the amount of yards a player gets. The top right corner is "ALL PRO" corner.

**Tight ends should not be compared to WR

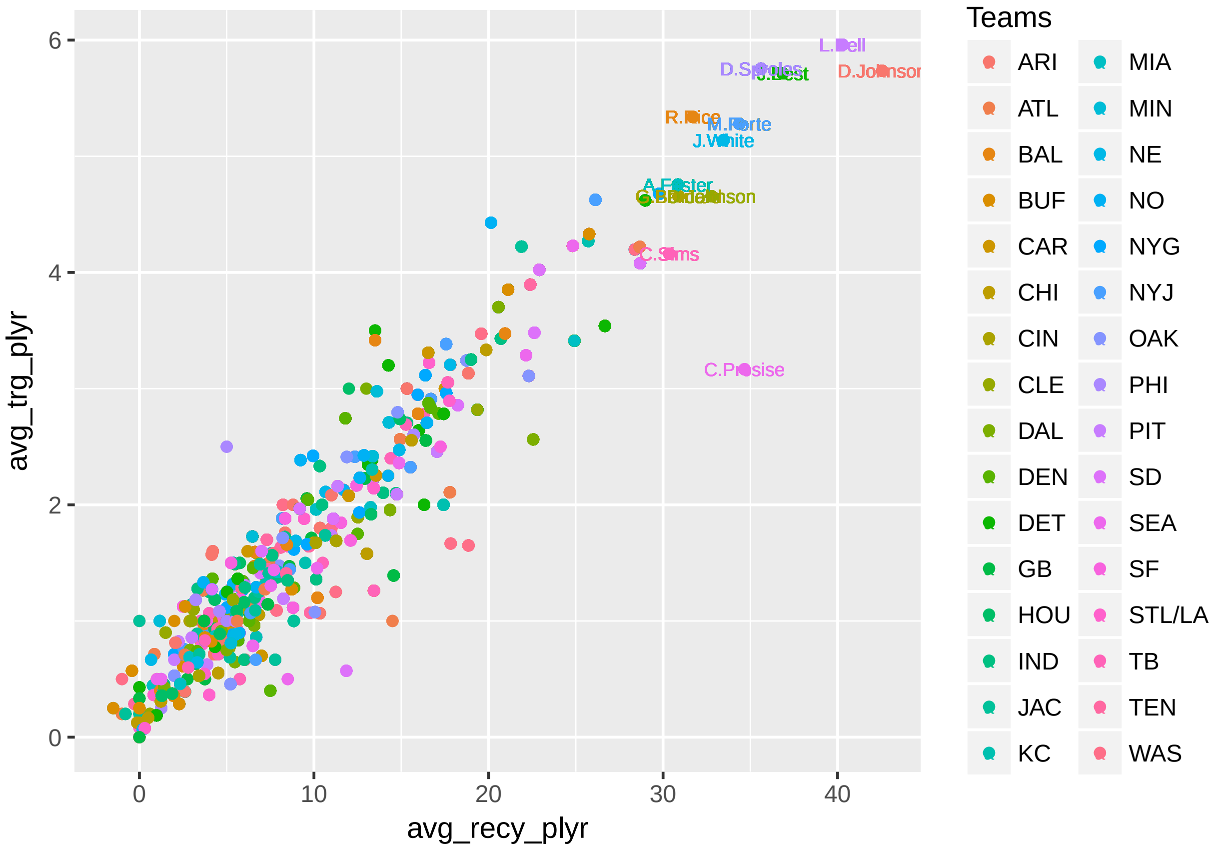
```r
ggplot(data = nfl_data, aes(x = avg_recy_plyr, y = avg_trg_plyr, col = Teams ))+
  geom_point(data = subset(nfl_data, pos1 == "TE"))+
  geom_text(data = subset(nfl_data, avg_recy_plyr > 50 & pos1 == "TE"), aes(label = pname), size = 2.5)
```



I separated out the TE from the WR. TE are not "homerun" hitters, but are frequent targets of QB's. Rob Gronkowski is the biggest anomaly here, he is widely considered the best position player to ever play.

** RB's separated out

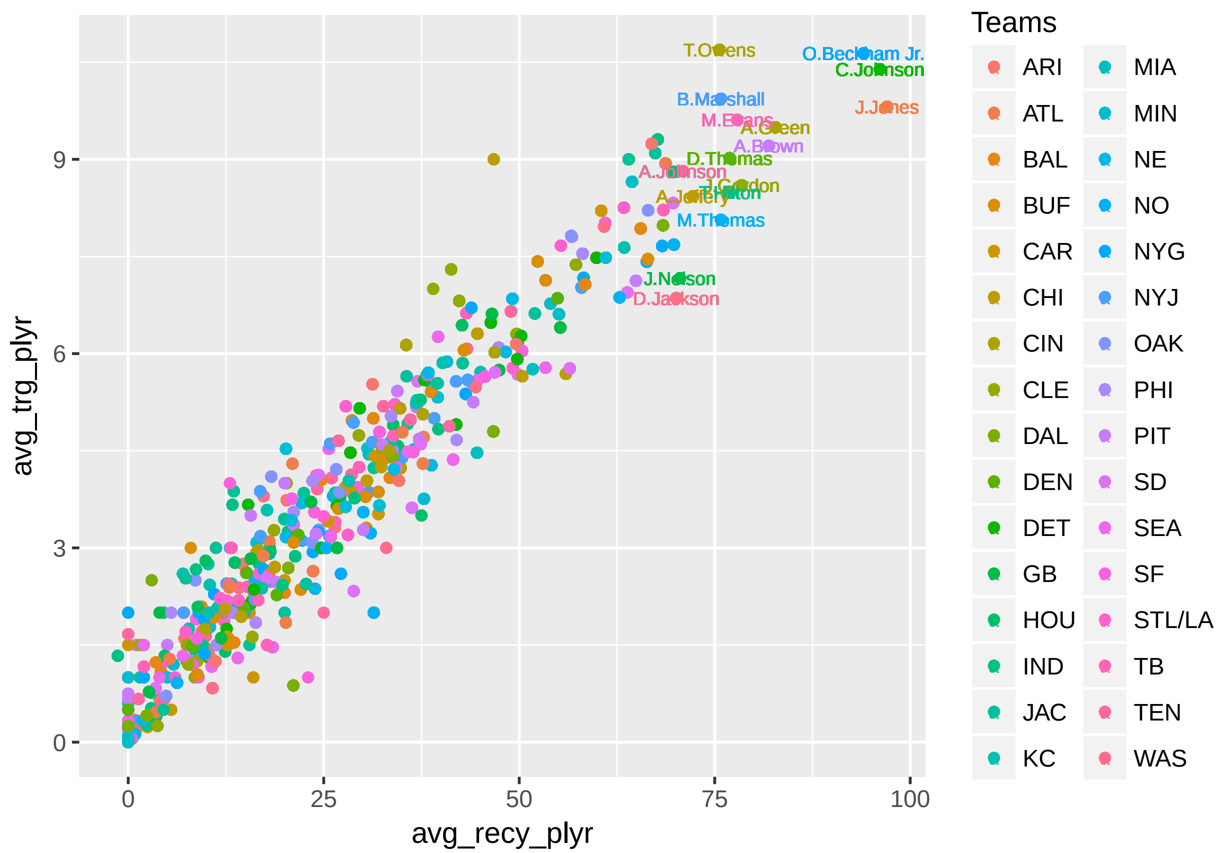
```
ggplot(data = nfl_data, aes(x = avg_recy_plyr, y = avg_trg_plyr, col = Teams ))+
  geom_point(data = subset(nfl_data, pos1 == "RB"))+
  geom_text(data = subset(nfl_data, avg_recy_plyr > 30 & pos1 == "RB"), aes(label = pname), size = 2.5)
```



CJ prosise was a rookie who had a couple of explosive games. He is a RB who played WR in college. He switched to RB his senior year of college and became an elite RB. This trend will regress somewhat, however, he is a very legit dual threat.

**WR only

```
ggplot(data = nfl_data, aes(x = avg_recy_plyr, y = avg_trg_plyr, col = Teams ))+
  geom_point(data = subset(nfl_data, pos1 == "WR"))+
  geom_text(data = subset(nfl_data, avg_recy_plyr > 70 & pos1 == "WR"), aes(label = pname), size = 2.5)
```



T.Owens
O.Beckham Jr.
C.Johnson
J.Jones
B.Marshall
M.Evans
A.Green
A.Brown
D.Thomas
A.Johnson
J.Gordon
A.Jeffery
T.Hilton
M.Thomas
J.Nelson
D.Jackson
avg_trg_plyr
0
3
6
9
0
25
50
75
100
avg_recy_plyr
Teams
ARI
ATL
BAL
BUF
CAR
CHI
CIN
CLE
DAL
DEN
DET
GB
HOU
IND
JAC
KC
MIA
MIN
NE
NO
NYG
NYJ
OAK
PHI
PIT
SD
SEA
SF
STL/LA
TB
TEN
WAS