

The NFL Combine and Fantasy Football

Ethan Allavarpu

Kyle Boal

4/10/2020

Contents

1	Data Gathering and Cleaning	2
1.1	Download the Data	2
1.2	Write Helper Functions	3
1.3	Create the Necessary Data Frames	4
1.3.1	Years and Positions that Abide by the Function	4
1.3.2	Years and Positions that are Outliers	4
1.4	Subset the Proper Variables	5
1.5	Remove Combine Non-Participants	5
1.6	Eight Game Minimum	6
2	Data Analysis	6
2.1	Helper Function	6
2.2	The Quarterback	7
2.2.1	A Model	9
2.2.2	A Transformation	10
2.2.3	Model Selection	13
2.3	The Running Back	16
2.3.1	A Model	17
2.3.2	A Transformation	20
2.3.3	Model Selection	22
2.4	The Wide Receiver	25
2.4.1	A Model	27
2.4.2	A Transformation	30
2.4.3	Model Selection	33
2.5	The Tight End	36
2.5.1	A Model	38
2.5.2	A Transformation	41
2.5.3	Model Selection	44

3	Final Models	47
3.1	Quarterback	47
3.2	Running Back	47
3.3	Wide Receiver	48
3.4	Tight End	48
4	How Good Are These Rookies?	49
4.1	Quarterbacks	49
4.1.1	Change Over Time	49
4.2	Running Backs	51
4.2.1	Change Over Time	51
4.3	Wide Receivers	53
4.3.1	Change Over Time	53
4.4	Tight Ends	55
4.4.1	Change Over Time	55
5	Predictions	57
5.1	Quarterbacks	57
5.2	Running Backs	58
5.3	Wide Receivers	59
5.4	Tight Ends	60
6	Wonderlic Scores	60

1 Data Gathering and Cleaning

1.1 Download the Data

```

qb_combine_1 <- read.csv("2000QBcombine.csv", stringsAsFactors = FALSE)
qb_combine_2 <- read.csv("2010QBcombine.csv", stringsAsFactors = FALSE)
qb_combine <- rbind(qb_combine_1, qb_combine_2)

rb_combine_1 <- read.csv("2000RBcombine.csv", stringsAsFactors = FALSE)
rb_combine_2 <- read.csv("2008RBcombine.csv", stringsAsFactors = FALSE)
rb_combine_3 <- read.csv("2014RBcombine.csv", stringsAsFactors = FALSE)
rb_combine <- rbind(rb_combine_1, rb_combine_2, rb_combine_3)

wr_combine_1 <- read.csv("2000WRcombine.csv", stringsAsFactors = FALSE)
wr_combine_2 <- read.csv("2005WRCombine.csv", stringsAsFactors = FALSE)
wr_combine_3 <- read.csv("2010WRCombine.csv", stringsAsFactors = FALSE)
wr_combine_4 <- read.csv("2014WRcombine.csv", stringsAsFactors = FALSE)
wr_combine <- rbind(wr_combine_1, wr_combine_2, wr_combine_3, wr_combine_4)

```

```
te_combine_1 <- read.csv("2000TEcombine.csv", stringsAsFactors = FALSE)
te_combine_2 <- read.csv("2011TEcombine.csv", stringsAsFactors = FALSE)
te_combine <- rbind(te_combine_1, te_combine_2)
```

1.2 Write Helper Functions

```
clean_players <- function(fantasy) {
  players <- as.character(fantasy$Player)
  players <- strsplit(players, "[*+]")
  for (i in seq_along(players)) {
    fantasy$Player[i] <- players[[i]][1]
  }
  fantasy$Player
}

combine_and_football <- function(fantasy_year, position) {
  position <- casefold(position)
  if (position == "qb") {
    combine <- qb_combine
  } else if (position == "rb") {
    combine <- rb_combine
  } else if (position == "wr") {
    combine <- wr_combine
  } else if (position == "te") {
    combine <- te_combine
  } else {
    stop("Enter a valid position abbreviation!")
  }
  pos <- data.frame()
  for (i in fantasy_year) {
    fantasy <- read.csv(paste(i, "fantasy.csv", sep = ""),
                        stringsAsFactors = FALSE)
    combine_specific <- combine %>% filter(Year == i)
    fantasy$Player <- clean_players(fantasy)
    rookie <- fantasy[fantasy$Player %in% combine_specific$Player, ] %>%
      filter(FantPos == toupper(position))
    rookie <- rookie[order(rookie$Player), ]
    relevant <- combine_specific[as.character(combine_specific$Player) %in%
                                rookie$Player, ]
    relevant <- relevant[order(relevant$Player), ]
    specific <- cbind(relevant, "Games" = rookie$G, "Points" = rookie$FantPt.)
    pos <- rbind(pos, specific)
  }
  pos
}
```

1.3 Create the Necessary Data Frames

1.3.1 Years and Positions that Abide by the Function

```
qbs <- combine_and_football(2000:2019, "qb")

rbs_1 <- combine_and_football(2000:2001, "rb")
rbs_2 <- combine_and_football(2003:2006, "rb")
rbs_3 <- combine_and_football(2008:2019, "rb")

wrs_1 <- combine_and_football(2000:2006, "wr")
wrs_2 <- combine_and_football(2008:2009, "wr")
wrs_3 <- combine_and_football(2011:2019, "wr")

tes <- combine_and_football(2000:2019, "te")
```

1.3.2 Years and Positions that are Outliers

```
position <- "RB"
i <- 2002
combine <- rb_combine
fantasy <- read.csv(paste(i, "fantasy.csv", sep = ""), stringsAsFactors = FALSE)
combine_specific <- combine %>% filter(Year == i)
fantasy$Player <- clean_players(fantasy)
rookie <- fantasy[fantasy$Player %in% combine_specific$Player, ] %>%
  filter(FantPos == toupper(position))
rookie <- rookie[order(rookie$Player), ]
relevant <- combine_specific[as.character(combine_specific$Player) %in%
  rookie$Player, ]
relevant <- relevant[order(relevant$Player), ]
rookie <- rookie[-11, ]
rbs_2002 <- cbind(relevant, "Games" = rookie$G, "Points" = rookie$FantPt.)
```

```
position <- "RB"
i <- 2007
combine <- rb_combine
fantasy <- read.csv(paste(i, "fantasy.csv", sep = ""), stringsAsFactors = FALSE)
combine_specific <- combine %>% filter(Year == i)
fantasy$Player <- clean_players(fantasy)
rookie <- fantasy[fantasy$Player %in% combine_specific$Player, ] %>%
  filter(FantPos == toupper(position))
rookie <- rookie[order(rookie$Player), ]
relevant <- combine_specific[as.character(combine_specific$Player) %in%
  rookie$Player, ]
relevant <- relevant[order(relevant$Player), ]
rookie <- rookie[-2, ]
rbs_2007 <- cbind(relevant, "Games" = rookie$G, "Points" = rookie$FantPt.)
```

```
position <- "WR"
i <- 2007
```

```

combine <- wr_combine
fantasy <- read.csv(paste(i, "fantasy.csv", sep = ""), stringsAsFactors = FALSE)
combine_specific <- combine %>% filter(Year == i)
fantasy$Player <- clean_players(fantasy)
rookie <- fantasy[fantasy$Player %in% combine_specific$Player, ] %>%
  filter(FantPos == toupper(position))
rookie <- rookie[order(rookie$Player), ]
relevant <- combine_specific[as.character(combine_specific$Player) %in%
  rookie$Player, ]
relevant <- relevant[order(relevant$Player), ]
rookie <- rookie[-c(5, 17), ]
wrs_2007 <- cbind(relevant, "Games" = rookie$G, "Points" = rookie$FantPt.)

```

```

position <- "WR"
i <- 2010
combine <- wr_combine
fantasy <- read.csv(paste(i, "fantasy.csv", sep = ""), stringsAsFactors = FALSE)
combine_specific <- combine %>% filter(Year == i)
fantasy$Player <- clean_players(fantasy)
rookie <- fantasy[fantasy$Player %in% combine_specific$Player, ] %>%
  filter(FantPos == toupper(position))
rookie <- rookie[order(rookie$Player), ]
relevant <- combine_specific[as.character(combine_specific$Player) %in% rookie$Player, ]
relevant <- relevant[order(relevant$Player), ]
rookie <- rookie[-25, ]
wrs_2010 <- cbind(relevant, "Games" = rookie$G, "Points" = rookie$FantPt.)

```

```

rbs <- rbind(rbs_1, rbs_2002, rbs_2, rbs_2007, rbs_3)
wrs <- rbind(wrs_1, wrs_2007, wrs_2, wrs_2010, wrs_3)

```

1.4 Subset the Proper Variables

```

qbs <- qbs %>% dplyr::select(Year:Pos, X40YD:Shuttle, Games:Points)
rbs <- rbs %>% dplyr::select(Year:Pos, X40YD:Shuttle, Games:Points)
wrs <- wrs %>% dplyr::select(Year:Pos, X40YD:Shuttle, Games:Points)
tes <- tes %>% dplyr::select(Year:Pos, X40YD:Shuttle, Games:Points)

```

1.5 Remove Combine Non-Participants

```

no_combine <- function(position) {
  combine_stats <- position %>% dplyr::select(X40YD:Shuttle)
  any_combine <- apply(combine_stats, 1, is.na)
  none <- apply(any_combine, 2, sum)
  events <- ncol(combine_stats)
  position$Player[none == events]
}
no_combine(qbs)

```

```
[1] "Zach Mettenberger" "Lamar Jackson"      "Kyler Murray"
```

```
qbs <- qbs[qbs$Player != no_combine(qbs)[1] &
          qbs$Player != no_combine(qbs)[2] &
          qbs$Player != no_combine(qbs)[3], ]
qbs <- qbs[!is.na(qbs$Points), ]
no_combine(rbs)
```

```
[1] "Josh Jacobs"
```

```
rbs <- rbs[rbs$Player != no_combine(rbs), ]
rbs <- rbs[!is.na(rbs$Points), ]
no_combine(wrs)
```

```
[1] "Corey Davis"      "Dede Westbrook"   "Dante Pettis"     "Deontay Burnett"
[5] "Marquise Brown"
```

```
wrs <- wrs[wrs$Player != no_combine(wrs)[1] &
          wrs$Player != no_combine(wrs)[2] &
          wrs$Player != no_combine(wrs)[3] &
          wrs$Player != no_combine(wrs)[4] &
          wrs$Player != no_combine(wrs)[5], ]
wrs <- wrs[!is.na(wrs$Points), ]
no_combine(tes)
```

```
[1] "Jordan Akins"
```

```
tes <- tes[tes$Player != no_combine(tes), ]
tes <- tes[!is.na(tes$Points), ]
```

1.6 Eight Game Minimum

```
qbs <- qbs %>% filter(Games >= 8) %>% filter(Points >= 10)
rbs <- rbs %>% filter(Games >= 8) %>% filter(Points >= 10)
wrs <- wrs %>% filter(Games >= 8) %>% filter(Points >= 10)
tes <- tes %>% filter(Games >= 8) %>% filter(Points >= 10)
```

2 Data Analysis

*Note: When determining analysis on the combine events, before simply fitting a model to **everything**, we first only consider the events in which **at least 60% of the participants at that position** partook in the event during the combine.*

2.1 Helper Function

```
total_nas <- function(position) {
  na_chart <- lapply(position, is.na)
  vapply(na_chart, sum, numeric(1))
}
```

2.2 The Quarterback

```
nrow(qbs)
```

```
[1] 54
```

```
total_nas(qbs)
```

Year	Player	Pos	X40YD	Vertical	BenchReps	Broad.Jump
0	0	0	1	14	53	15
X3Cone	Shuttle	Games	Points			
16	15	0	0			

```
total_nas(qbs) / nrow(qbs)
```

Year	Player	Pos	X40YD	Vertical	BenchReps	Broad.Jump
0.00000000	0.00000000	0.00000000	0.01851852	0.25925926	0.98148148	0.27777778
X3Cone	Shuttle	Games	Points			
0.29629630	0.27777778	0.00000000	0.00000000			

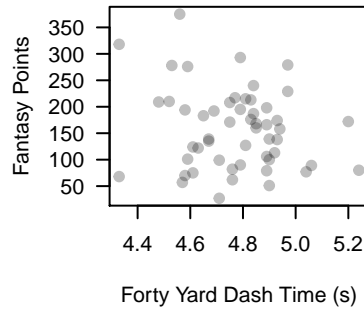
```
names(qbs)[total_nas(qbs) / nrow(qbs) >= 0.4]
```

```
[1] "BenchReps"
```

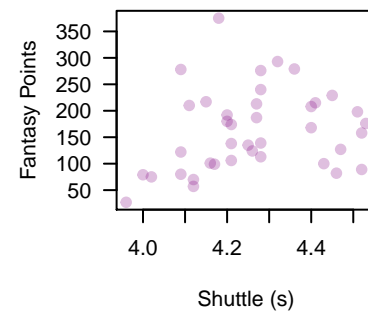
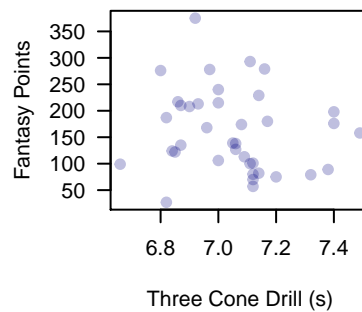
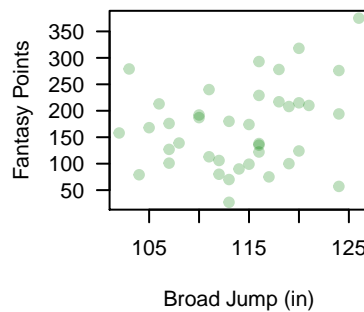
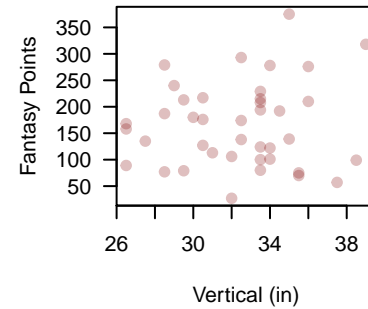
```
qb_rel_vars <- qbs %>% dplyr::select(Player,
                                     X40YD:Vertical,
                                     Broad.Jump:Shuttle,
                                     Points)

par(mfrow = c(2, 3))
plot(Points ~ X40YD, data = qb_rel_vars,
     xlab = "Forty Yard Dash Time (s)",
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[1],
     las = 1)
plot(0:10, 0:10, type = "n",
     xaxt = "n", yaxt = "n",
     xlab = "", ylab = "")
text(5, 5, "Fantasy Points\n vs.\nCombine Events",
     cex = 1.5)
plot(Points ~ Vertical, data = qb_rel_vars,
     xlab = "Vertical (in)",
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[2],
     las = 1)
plot(Points ~ Broad.Jump, data = qb_rel_vars,
     xlab = "Broad Jump (in)",
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[3],
     las = 1)
```

```
plot(Points ~ X3Cone, data = qb_rel_vars,
     xlab = "Three Cone Drill (s)",
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[4],
     las = 1)
plot(Points ~ Shuttle, data = qb_rel_vars,
     xlab = "Shuttle (s)",
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[5],
     las = 1)
```



Fantasy Points
vs.
Combine Events



```
with(qb_rel_vars, cor(X40YD, Points, use = "complete.obs"))
```

```
[1] -0.1893663
```

```
with(qb_rel_vars, cor(Vertical, Points, use = "complete.obs"))
```

```
[1] 0.09480982
```

```
with(qb_rel_vars, cor(Broad.Jump, Points, use = "complete.obs"))
```

```
[1] 0.2581988
```

```
with(qb_rel_vars, cor(X3Cone, Points, use = "complete.obs"))
```

```
[1] -0.1544544
```

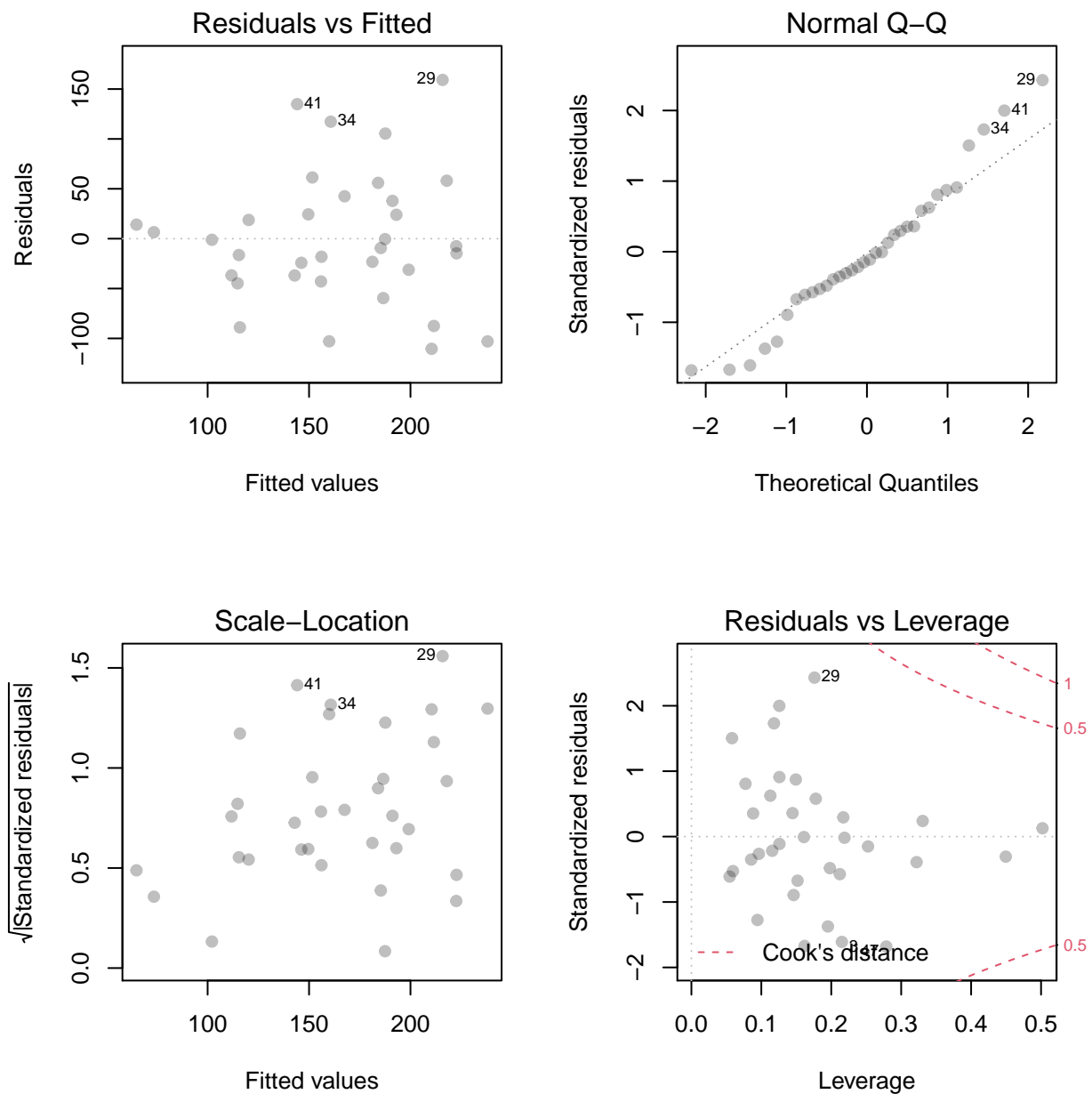


```
with(qb_rel_vars, cor(Shuttle, Points, use = "complete.obs"))
```

```
[1] 0.2290112
```

2.2.1 A Model

```
qb_model <- lm(Points ~ . - Player, data = qb_rel_vars)
par(mfrow = c(2, 2))
plot(qb_model,
     pch = 19,
     col = rgb(0, 0, 0, alpha = 0.25),
     add.smooth = FALSE)
```



```
summary(qb_model)
```

Call:

```
lm(formula = Points ~ . - Player, data = qb_rel_vars)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-110.440	-36.879	-8.477	34.496	159.175

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-470.438	871.893	-0.540	0.5938
X40YD	-73.101	95.488	-0.766	0.4504
Vertical	-9.522	6.113	-1.558	0.1306
Broad.Jump	5.768	2.999	1.923	0.0647 .
X3Cone	-36.248	85.826	-0.422	0.6760
Shuttle	209.796	96.383	2.177	0.0381 *

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

Residual standard error: 72.16 on 28 degrees of freedom

(20 observations deleted due to missingness)

Multiple R-squared: 0.3024, Adjusted R-squared: 0.1779

F-statistic: 2.428 on 5 and 28 DF, p-value: 0.05997

2.2.2 A Transformation

```
summary(powerTransform(Points ~ . - Player, data = qb_rel_vars))
```

Warning in estimateTransform.default(X, Y, weights, family, ...): Convergence failure: return code = 52

bcPower Transformation to Normality

	Est	Power	Rounded	Pwr	Wald	Lwr	Bnd	Wald	Upr	Bnd
Y1	0.3553			0		-0.1459			0.8565	

Likelihood ratio test that transformation parameter is equal to 0
(log transformation)

	LRT	df	pval
LR test, lambda = (0)	2.138913	1	0.1436

Likelihood ratio test that no transformation is needed

	LRT	df	pval
LR test, lambda = (1)	5.305285	1	0.021261

```
summary(powerTransform(cbind(X40YD, Vertical, Broad.Jump,  
                             X3Cone, Shuttle) ~ 1,  
                  data = qb_rel_vars))
```

bcPower Transformations to Multinormality

	Est Power	Rounded Pwr	Wald Lwr Bnd	Wald Up Bnd
X40YD	-7.7176	1	-16.6223	1.1870
Vertical	1.4037	1	-1.5119	4.3192
Broad.Jump	0.4020	1	-5.4326	6.2367
X3Cone	-2.8106	1	-13.2733	7.6522
Shuttle	1.3605	1	-7.9540	10.6751

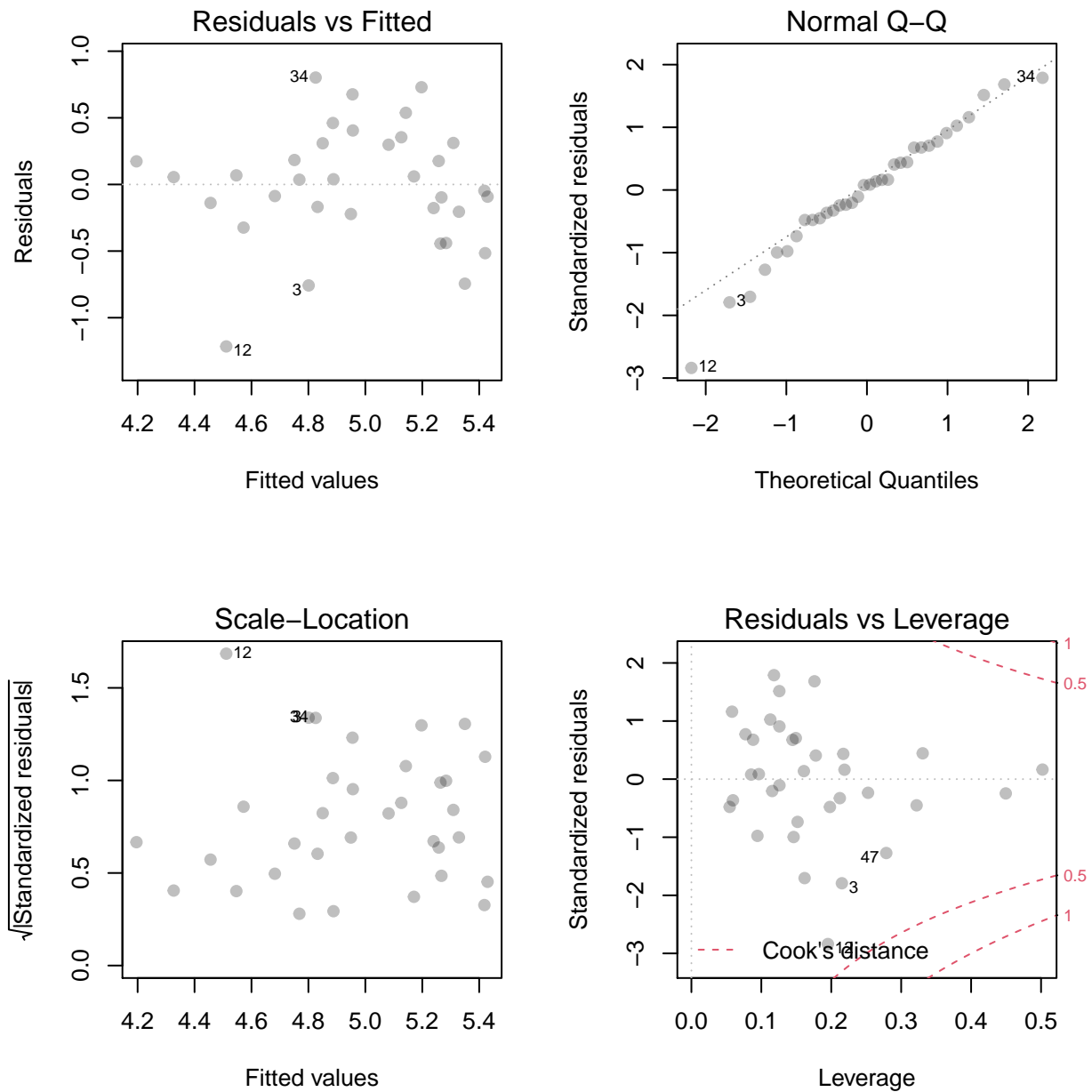
Likelihood ratio test that transformation parameters are equal to 0
(all log transformations)

LRT df pval
LR test, lambda = (0 0 0 0 0) 4.365722 5 0.49805

Likelihood ratio test that no transformations are needed

LRT df pval
LR test, lambda = (1 1 1 1 1) 4.846294 5 0.43493

```
qb_relevant_transformed <- qb_rel_vars %>% mutate("New_Points" = log(Points))
qb_transformed_model <- lm(New_Points ~ . - Points - Player,
                           data = qb_relevant_transformed)
par(mfrow = c(2, 2))
plot(qb_transformed_model,
     pch = 19,
     col = rgb(0, 0, 0, alpha = 0.25),
     add.smooth = FALSE)
```



```
summary(qb_transformed_model)
```

Call:

```
lm(formula = New_Points ~ . - Points - Player, data = qb_relevant_transformed)
```

Residuals:

	Min	1Q	Median	3Q	Max
Residuals	-1.21560	-0.19795	0.03742	0.30621	0.80218

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-0.49999	5.76765	-0.087	0.9315
X40YD	-0.40512	0.63166	-0.641	0.5265

```

Vertical      -0.05570    0.04044  -1.377   0.1793
Broad.Jump    0.03056    0.01984   1.540   0.1347
X3Cone        -0.37061    0.56775  -0.653   0.5192
Shuttle       1.96380    0.63759   3.080   0.0046 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4773 on 28 degrees of freedom
(20 observations deleted due to missingness)
Multiple R-squared:  0.3661,    Adjusted R-squared:  0.2529
F-statistic: 3.234 on 5 and 28 DF,  p-value: 0.01984

```

2.2.3 Model Selection

```

qb_best_subsets <- regsubsets(New_Points ~ . - Points - Player,
                              data = qb_relevant_transformed,
                              nvmax = 5)
summary(qb_best_subsets)

```

```

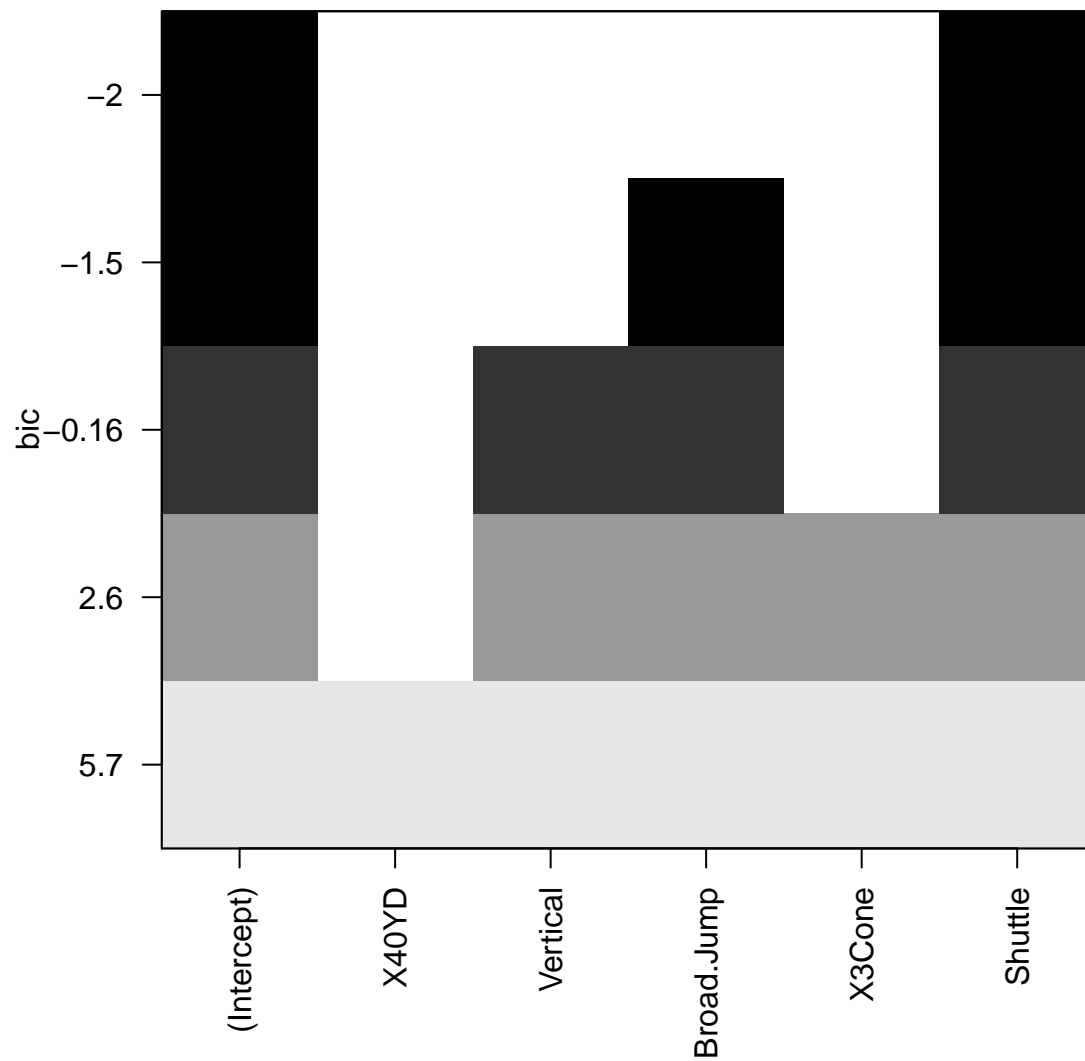
Subset selection object
Call: regsubsets.formula(New_Points ~ . - Points - Player, data = qb_relevant_transformed,
                          nvmax = 5)
5 Variables (and intercept)
      Forced in Forced out
X40YD      FALSE      FALSE
Vertical    FALSE      FALSE
Broad.Jump  FALSE      FALSE
X3Cone      FALSE      FALSE
Shuttle     FALSE      FALSE
1 subsets of each size up to 5
Selection Algorithm: exhaustive
      X40YD Vertical Broad.Jump X3Cone Shuttle
1 ( 1 ) " " " " " " "*"
2 ( 1 ) " " " " "*" " "*"
3 ( 1 ) " " "*" "*" " "*"
4 ( 1 ) " " "*" "*" "*" "*"
5 ( 1 ) "*" "*" "*" "*" "*"

```

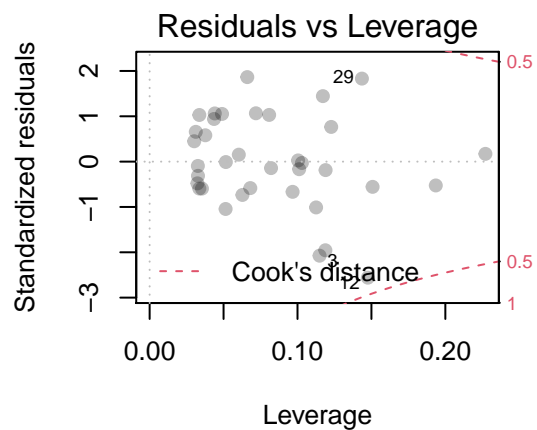
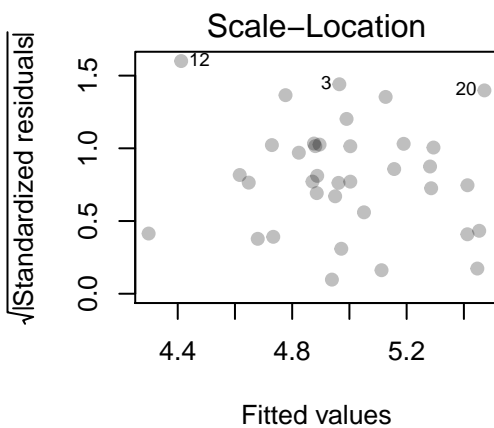
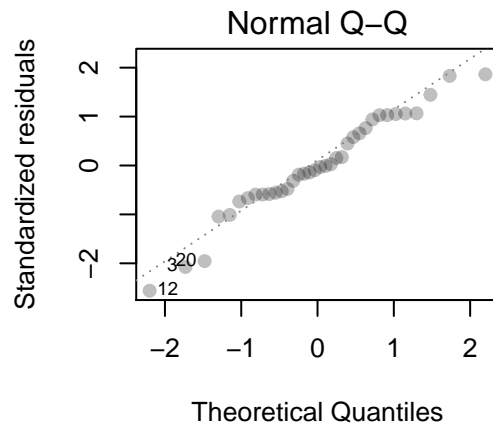
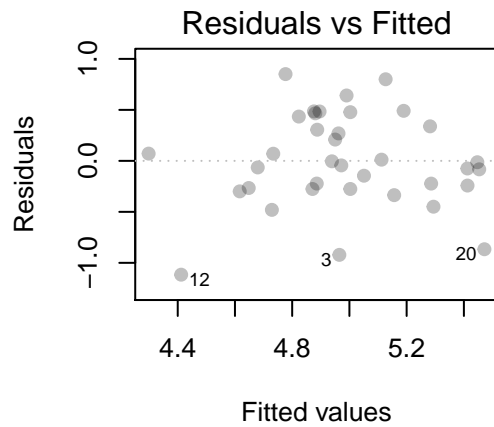
```

plot(qb_best_subsets)

```



```
final_qb_model <- lm(New_Points ~ Broad.Jump + Shuttle,
                     data = qb_relevant_transformed)
par(mfrow = c(2, 2))
plot(final_qb_model,
     pch = 19,
     col = rgb(0, 0, 0, alpha = 0.25),
     add.smooth = FALSE)
```



```
summary(final_qb_model)
```

Call:

```
lm(formula = New_Points ~ Broad.Jump + Shuttle, data = qb_relevant_transformed)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-1.11619	-0.26852	-0.02889	0.36250	0.85084

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-5.85609	3.15626	-1.855	0.07249 .
Broad.Jump	0.02144	0.01337	1.603	0.11847
Shuttle	1.98119	0.56961	3.478	0.00144 **

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

Residual standard error: 0.4722 on 33 degrees of freedom
 (18 observations deleted due to missingness)
 Multiple R-squared: 0.2778, Adjusted R-squared: 0.234
 F-statistic: 6.346 on 2 and 33 DF, p-value: 0.004655

2.3 The Running Back

```
nrow(rbs)
```

```
[1] 215
```

```
total_nas(rbs)
```

Year	Player	Pos	X40YD	Vertical	BenchReps	Broad.Jump
0	0	0	4	39	52	42
X3Cone	Shuttle	Games	Points			
98	93	0	0			

```
total_nas(rbs) / nrow(rbs)
```

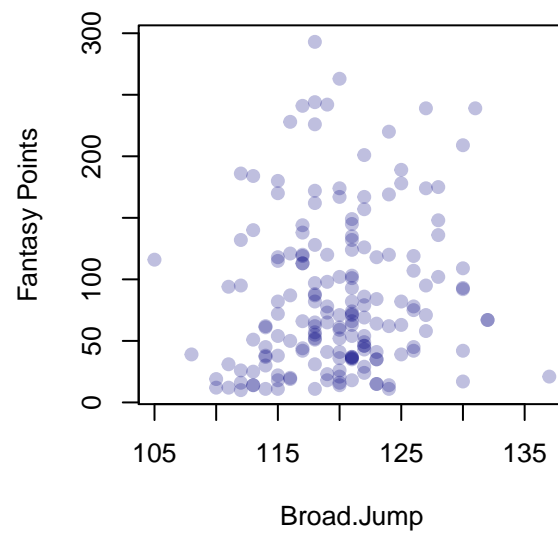
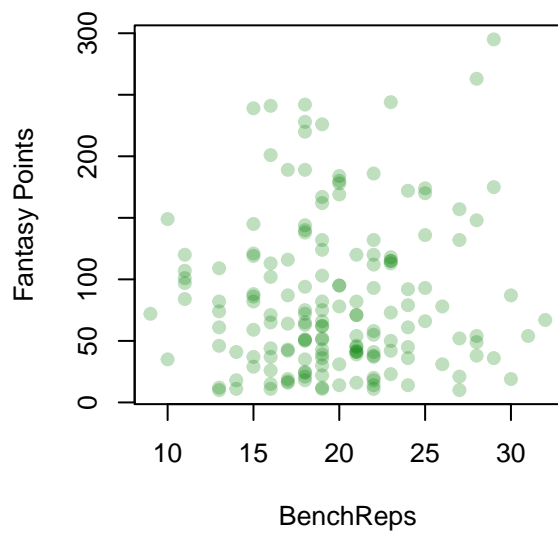
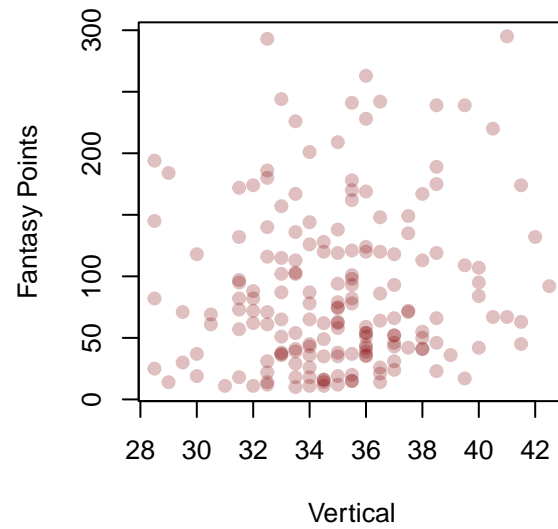
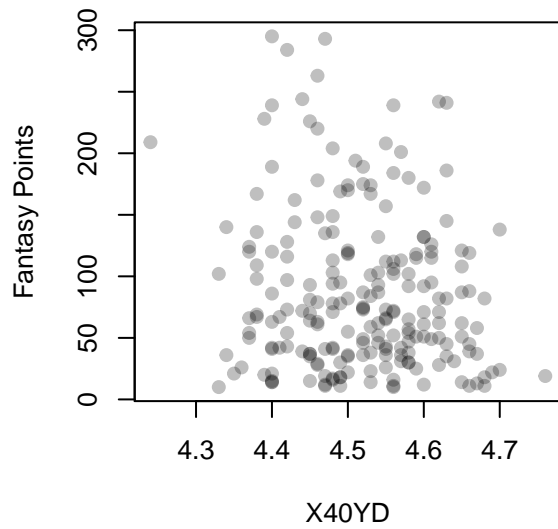
Year	Player	Pos	X40YD	Vertical	BenchReps	Broad.Jump
0.00000000	0.00000000	0.00000000	0.01860465	0.18139535	0.24186047	0.19534884
X3Cone	Shuttle	Games	Points			
0.45581395	0.43255814	0.00000000	0.00000000			

```
names(rbs)[total_nas(rbs) / nrow(rbs) >= 0.4]
```

```
[1] "X3Cone" "Shuttle"
```

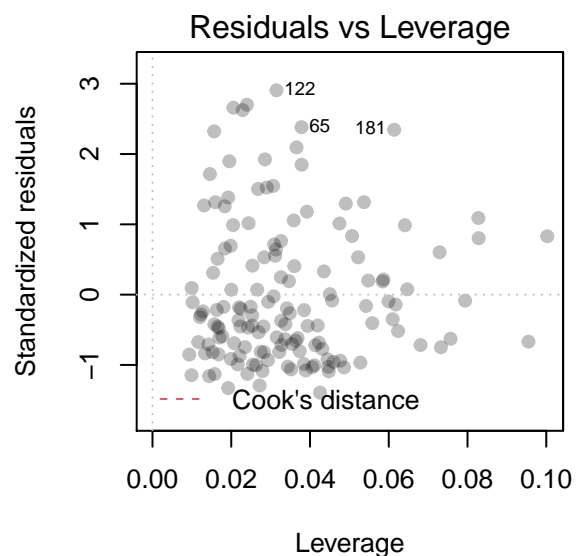
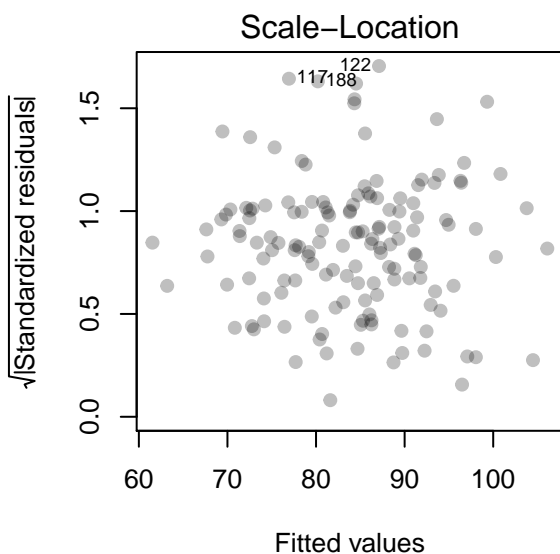
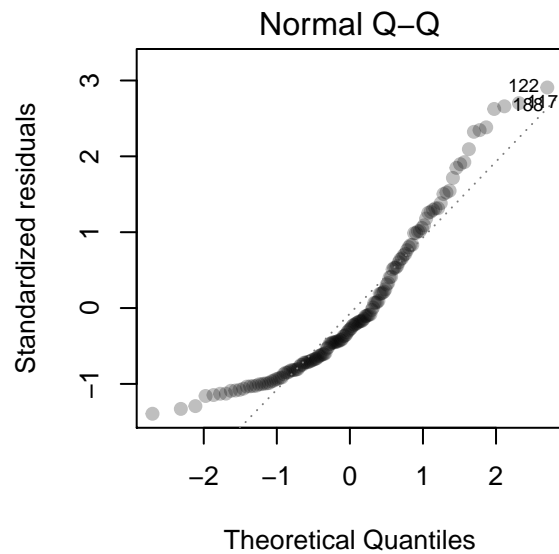
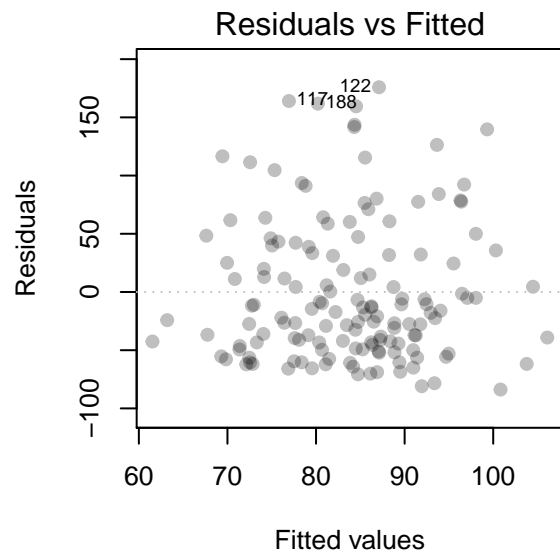
```
rb_rel_vars <- rbs %>% dplyr::select(Player,
                                     X40YD:Broad.Jump,
                                     Points)

par(mfrow = c(2, 2))
plot(Points ~ X40YD, data = rb_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[1])
plot(Points ~ Vertical, data = rb_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[2])
plot(Points ~ BenchReps, data = rb_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[3])
plot(Points ~ Broad.Jump, data = rb_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[4])
```

2.3.1 A Model

```
rb_model <- lm(Points ~ . - Player, data = rb_rel_vars)
par(mfrow = c(2, 2))
plot(rb_model,
     pch = 19,
     col = rgb(0, 0, 0, alpha = 0.25),
     add.smooth = FALSE)
```



```
summary(rb_model)
```

Call:

```
lm(formula = Points ~ . - Player, data = rb_rel_vars)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-83.83	-45.36	-17.20	35.70	175.89

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	78.29926	329.75352	0.237	0.813
X40YD	-36.14237	59.79563	-0.604	0.547

Vertical	0.22259	2.21984	0.100	0.920
BenchReps	-0.01209	1.10074	-0.011	0.991
Broad.Jump	1.35275	1.27340	1.062	0.290

Residual standard error: 61.47 on 140 degrees of freedom

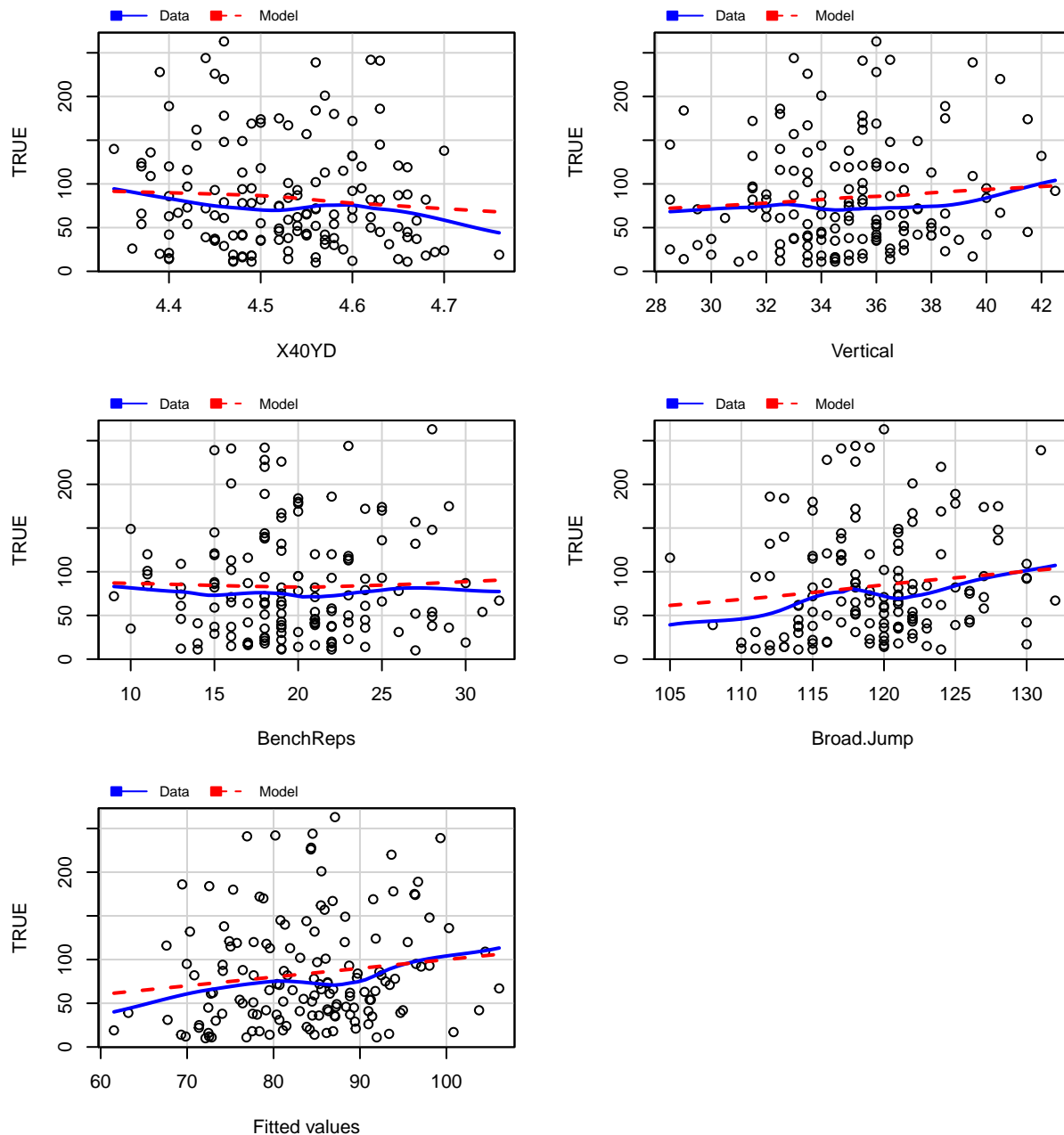
(70 observations deleted due to missingness)

Multiple R-squared: 0.02029, Adjusted R-squared: -0.007705

F-statistic: 0.7247 on 4 and 140 DF, p-value: 0.5764

`mmps(rb_model)`

Marginal Model Plots



2.3.2 A Transformation

```
summary(powerTransform(Points ~ . - Player, data = rb_rel_vars))
```

bcPower Transformation to Normality

	Est	Power	Rounded Pwr	Wald Lwr Bnd	Wald Up Bnd
Y1	0.1878		0	-0.0096	0.3852

Likelihood ratio test that transformation parameter is equal to 0
(log transformation)

	LRT	df	pval
LR test, lambda = (0)	3.528021	1	0.06034

Likelihood ratio test that no transformation is needed

	LRT	df	pval
LR test, lambda = (1)	58.55035	1	1.9762e-14

```
summary(powerTransform(cbind(X40YD, Vertical,  
                             BenchReps, Broad.Jump) ~ 1,  
                      data = rb_rel_vars))
```

bcPower Transformations to Multinormality

	Est	Power	Rounded Pwr	Wald Lwr Bnd	Wald Up Bnd
X40YD	-0.4595		1	-8.3813	7.4624
Vertical	1.0758		1	-0.4287	2.5803
BenchReps	0.6701		1	0.1152	1.2251
Broad.Jump	0.8828		1	-2.0134	3.7789

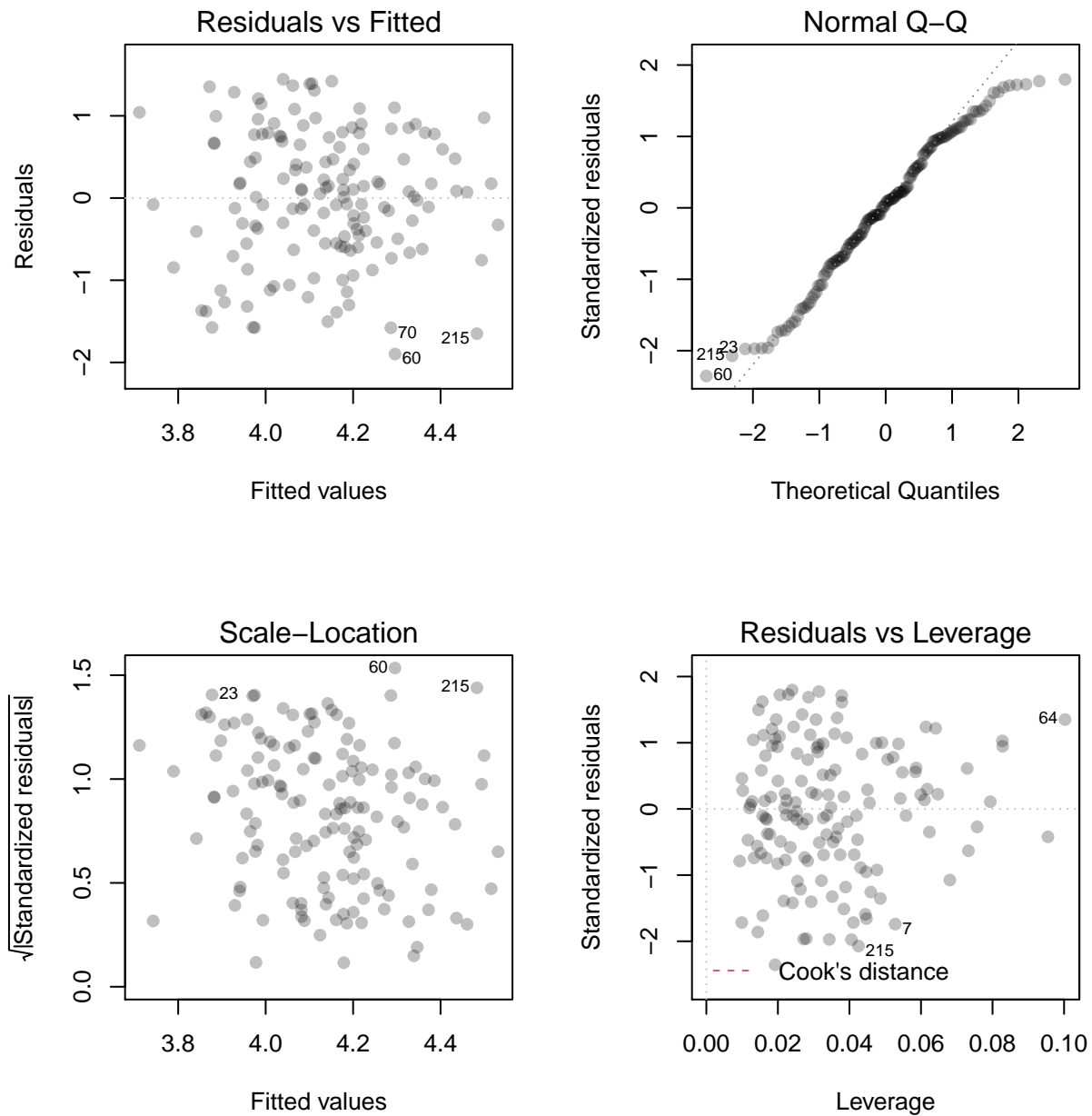
Likelihood ratio test that transformation parameters are equal to 0
(all log transformations)

	LRT	df	pval
LR test, lambda = (0 0 0 0)	8.007816	4	0.091292

Likelihood ratio test that no transformations are needed

	LRT	df	pval
LR test, lambda = (1 1 1 1)	1.476565	4	0.83078

```
rb_relevant_transformed <- rb_rel_vars %>% mutate("New_Points" = log(Points))  
rb_transformed_model <- lm(New_Points ~ . - Points - Player,  
                           data = rb_relevant_transformed)  
  
par(mfrow = c(2, 2))  
plot(rb_transformed_model,  
     pch = 19,  
     col = rgb(0, 0, 0, alpha = 0.25),  
     add.smooth = FALSE)
```



```
summary(rb_transformed_model)
```

Call:

```
lm(formula = New_Points ~ . - Points - Player, data = rb_relevant_transformed)
```

Residuals:

	Min	1Q	Median	3Q	Max
Residuals	-1.89799	-0.55298	0.04995	0.67209	1.44476

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	1.5240649	4.3651628	0.349	0.7275
X40YD	-0.2438744	0.7915538	-0.308	0.7585

```

Vertical      -0.0009534  0.0293855  -0.032   0.9742
BenchReps     -0.0020753  0.0145712  -0.142   0.8869
Broad.Jump    0.0317279  0.0168568   1.882   0.0619 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8138 on 140 degrees of freedom
(70 observations deleted due to missingness)
Multiple R-squared:  0.04106,    Adjusted R-squared:  0.01366
F-statistic: 1.499 on 4 and 140 DF,  p-value: 0.2059

```

2.3.3 Model Selection

```

rb_best_subsets <- regsubsets(New_Points ~ . - Points - Player,
                              data = rb_relevant_transformed,
                              nvmax = 4)
summary(rb_best_subsets)

```

```

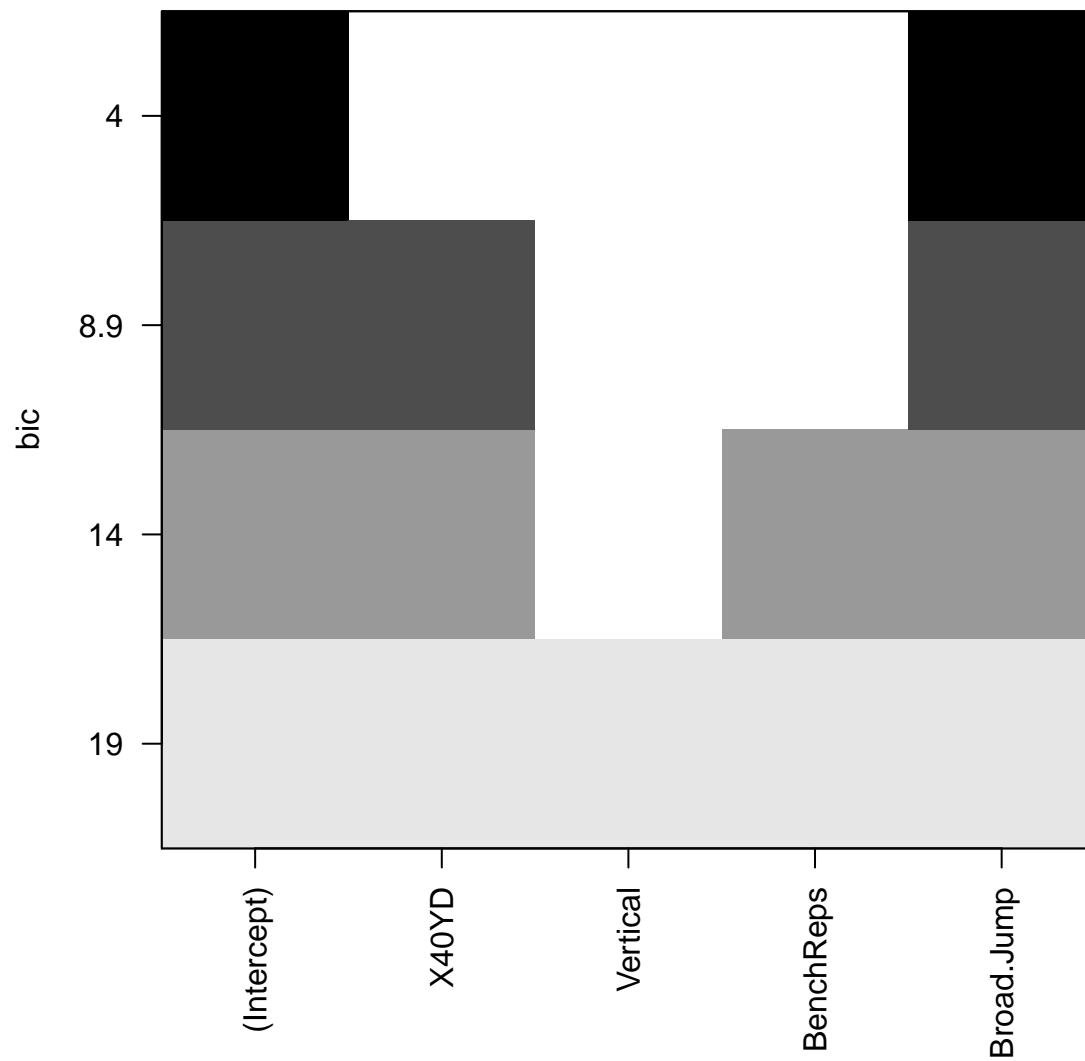
Subset selection object
Call: regsubsets.formula(New_Points ~ . - Points - Player, data = rb_relevant_transformed,
                          nvmax = 4)
4 Variables (and intercept)
      Forced in Forced out
X40YD      FALSE      FALSE
Vertical    FALSE      FALSE
BenchReps   FALSE      FALSE
Broad.Jump  FALSE      FALSE
1 subsets of each size up to 4
Selection Algorithm: exhaustive
      X40YD Vertical BenchReps Broad.Jump
1 ( 1 ) " "      " "      " "      "*"
2 ( 1 ) "*"     " "      " "      "*"
3 ( 1 ) "*"     " "      "*"     "*"
4 ( 1 ) "*"     "*"     "*"     "*"

```

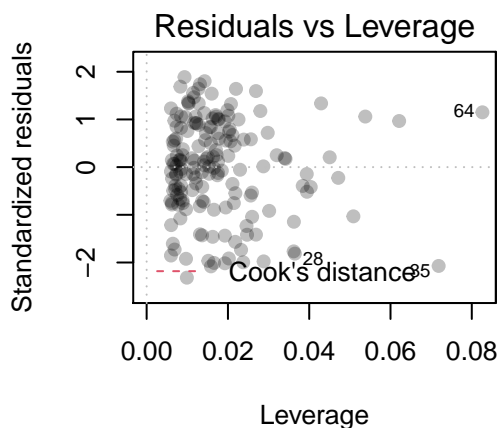
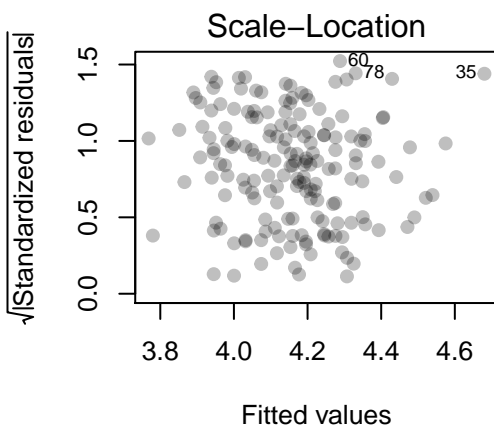
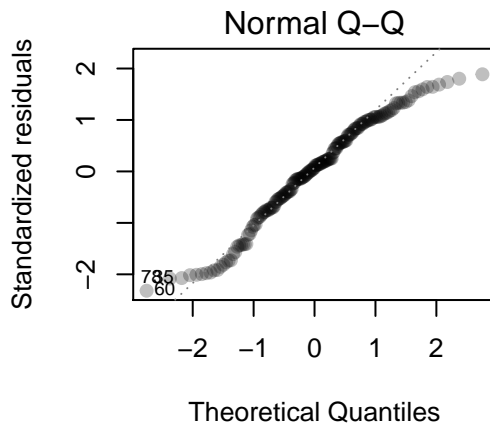
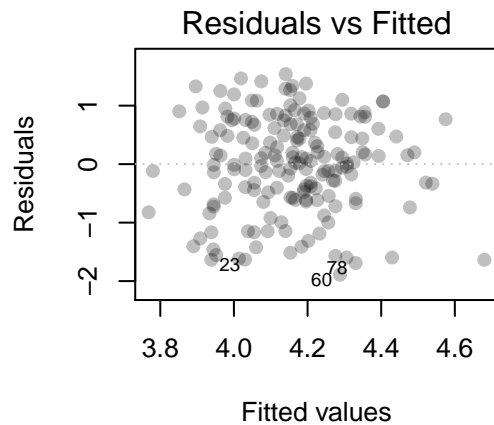
```

plot(rb_best_subsets)

```



```
final_rb_model <- lm(New_Points ~ Broad.Jump + X40YD, data = rb_relevant_transformed)
par(mfrow = c(2, 2))
plot(final_rb_model,
     pch = 19,
     col = rgb(0, 0, 0, alpha = 0.25),
     add.smooth = FALSE)
```



```
summary(final_rb_model)
```

Call:

```
lm(formula = New_Points ~ Broad.Jump + X40YD, data = rb_relevant_transformed)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.89030	-0.55716	0.05755	0.67639	1.53941

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	3.95120	4.06020	0.973	0.3319
Broad.Jump	0.02457	0.01294	1.899	0.0592
X40YD	-0.60625	0.71524	-0.848	0.3979

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

Residual standard error: 0.8193 on 168 degrees of freedom
 (44 observations deleted due to missingness)
 Multiple R-squared: 0.0363, Adjusted R-squared: 0.02483
 F-statistic: 3.164 on 2 and 168 DF, p-value: 0.04477

2.4 The Wide Receiver

```
nrow(wrs)
```

```
[1] 274
```

```
total_nas(wrs)
```

Year	Player	Pos	X40YD	Vertical	BenchReps	Broad.Jump
0	0	0	2	60	144	64
X3Cone	Shuttle	Games	Points			
108	100	0	0			

```
total_nas(wrs) / nrow(wrs)
```

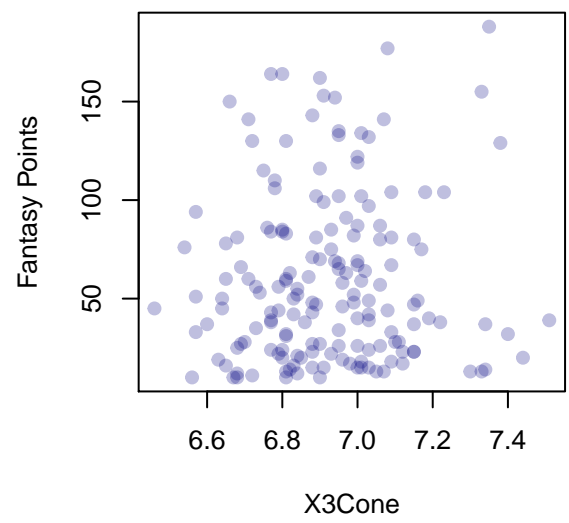
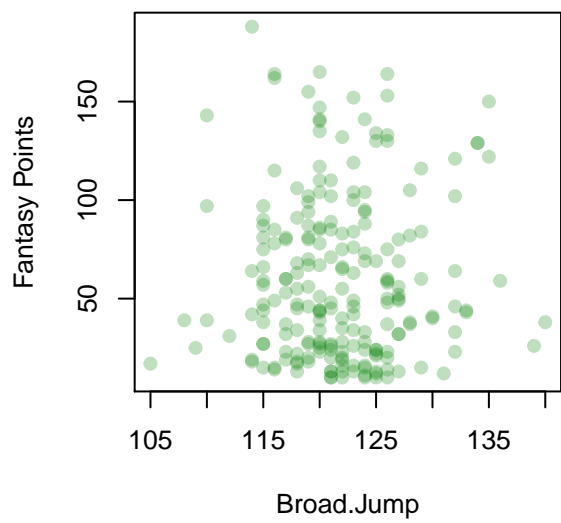
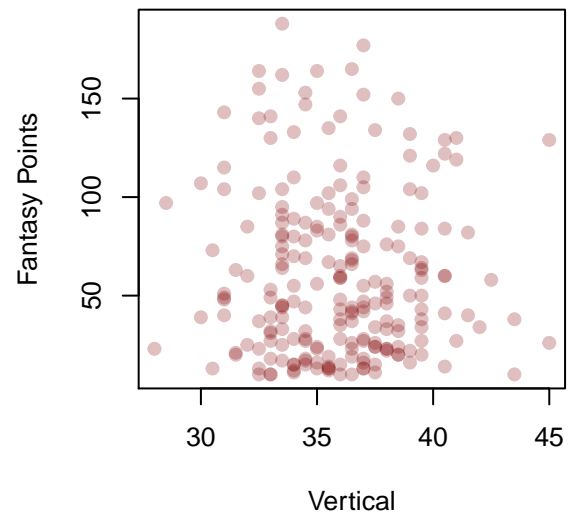
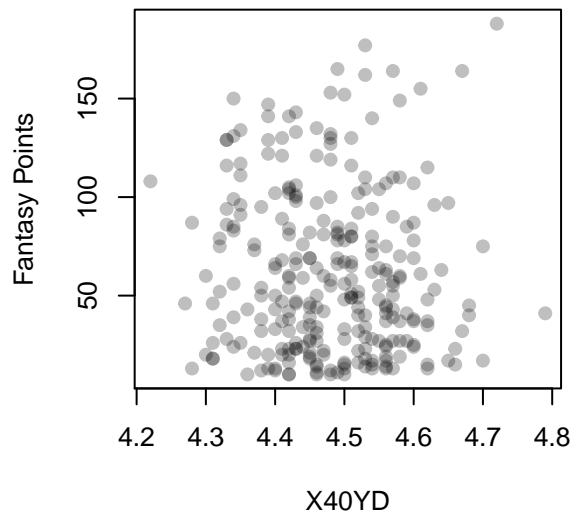
Year	Player	Pos	X40YD	Vertical	BenchReps	Broad.Jump
0.00000000	0.00000000	0.00000000	0.00729927	0.21897810	0.52554745	0.23357664
X3Cone	Shuttle	Games	Points			
0.39416058	0.36496350	0.00000000	0.00000000			

```
names(wrs)[total_nas(wrs) / nrow(wrs) >= 0.4]
```

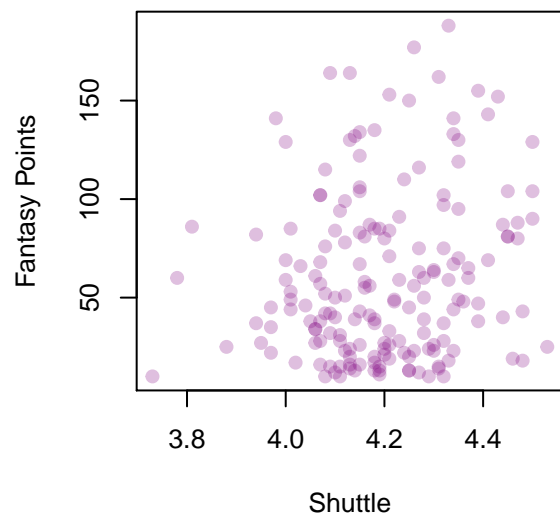
```
[1] "BenchReps"
```

```
wr_rel_vars <- wrs %>% dplyr::select(Player,
                                     X40YD:Vertical,
                                     Broad.Jump:Shuttle,
                                     Points)

par(mfrow = c(2, 2))
plot(Points ~ X40YD, data = wr_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[1])
plot(Points ~ Vertical, data = wr_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[2])
plot(Points ~ Broad.Jump, data = wr_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[3])
plot(Points ~ X3Cone, data = wr_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[4])
```

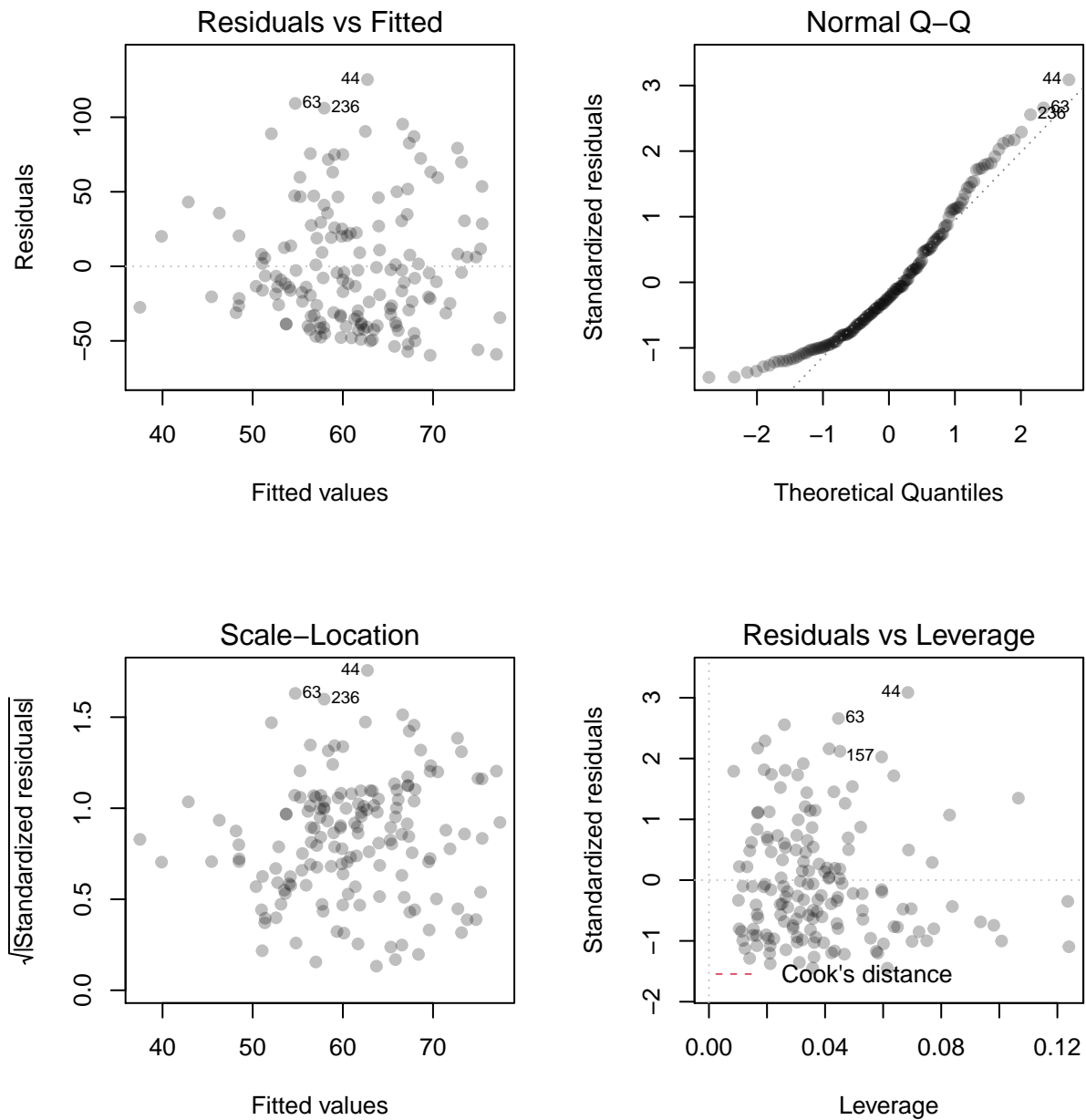


```
plot(Points ~ Shuttle, data = wr_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[5])
```



2.4.1 A Model

```
wr_model <- lm(Points ~ . - Player, data = wr_rel_vars)
par(mfrow = c(2, 2))
plot(wr_model,
     pch = 19,
     col = rgb(0, 0, 0, alpha = 0.25),
     add.smooth = FALSE)
```



```
summary(wr_model)
```

```
Call:
lm(formula = Points ~ . - Player, data = wr_rel_vars)
```

```
Residuals:
    Min       1Q   Median       3Q      Max
-59.68 -32.83  -9.14   25.09  125.27
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -73.25331   243.72721  -0.301   0.7642
X40YD       -10.08902    38.76219  -0.260   0.7950
```

```

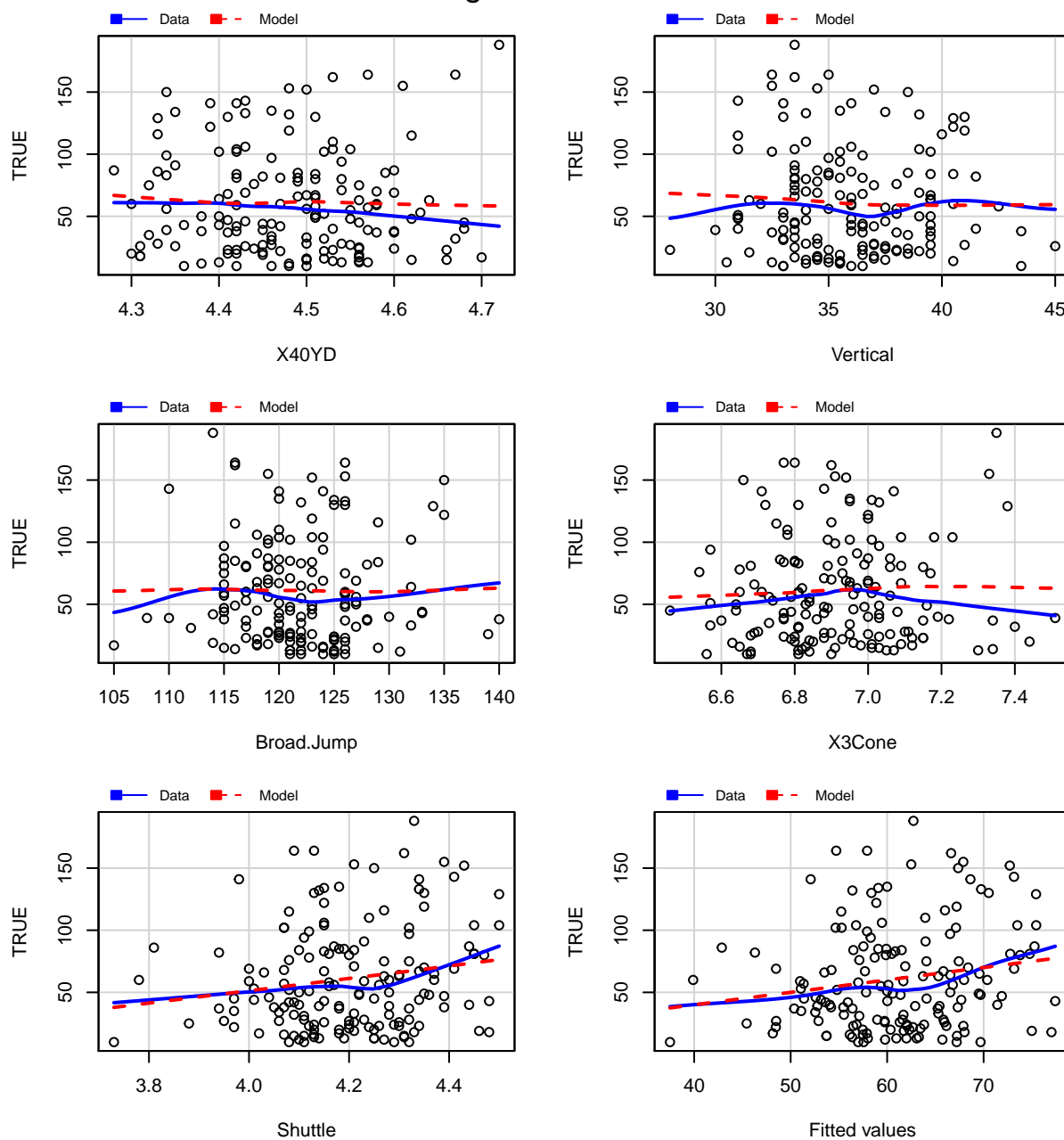
Vertical      -0.27956      1.39016   -0.201    0.8409
Broad.Jump     0.09779      0.72757    0.134    0.8933
X3Cone        -6.32532     20.30223   -0.312    0.7558
Shuttle       52.72718     27.44265    1.921    0.0566 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 42.05 on 151 degrees of freedom
(117 observations deleted due to missingness)
Multiple R-squared:  0.0317,    Adjusted R-squared:  -0.0003603
F-statistic: 0.9888 on 5 and 151 DF,  p-value: 0.4266

```

```
mmpr(wr_model)
```

Marginal Model Plots



2.4.2 A Transformation

```
summary(powerTransform(Points ~ . - Player, data = wr_rel_vars))
```

bcPower Transformation to Normality

	Est Power	Rounded Pwr	Wald	Lwr Bnd	Wald	Upwr Bnd
Y1	0.1811	0	-0.0321	0.3943		

Likelihood ratio test that transformation parameter is equal to 0

```

(log transformation)
              LRT df      pval
LR test, lambda = (0) 2.793331 1 0.094657

Likelihood ratio test that no transformation is needed
              LRT df      pval
LR test, lambda = (1) 52.88876 1 3.5294e-13

summary(powerTransform(cbind(X40YD, Vertical, Broad.Jump,
                             X3Cone, Shuttle) ~ 1,
                             data = wr_rel_vars))

bcPower Transformations to Multinormality
      Est Power Rounded Pwr Wald Lwr Bnd Wald Up Bnd
X40YD      0.0036         1   -6.5455      6.5528
Vertical    0.5047         1   -0.9666      1.9761
Broad.Jump  -0.2165         1   -2.5288      2.0958
X3Cone     -4.4547        -1   -8.7949     -0.1145
Shuttle      3.0331         1   -0.4006      6.4668

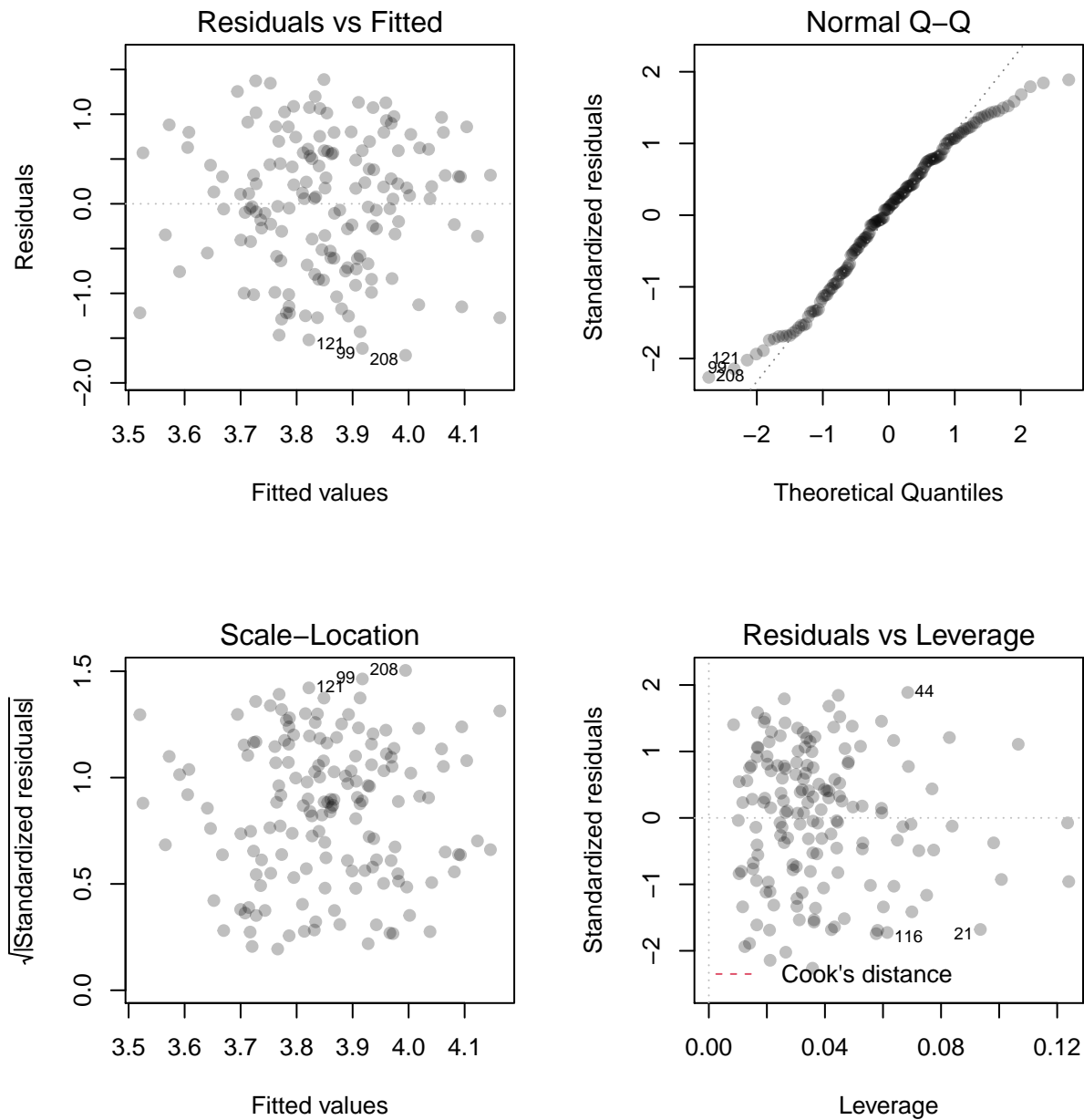
Likelihood ratio test that transformation parameters are equal to 0
(all log transformations)
              LRT df      pval
LR test, lambda = (0 0 0 0 0) 8.009242 5 0.15573

Likelihood ratio test that no transformations are needed
              LRT df      pval
LR test, lambda = (1 1 1 1 1) 9.218736 5 0.10065

wr_relevant_transformed <- wr_rel_vars %>% mutate("New_Points" = log(Points))
wr_transformed_model <- lm(New_Points ~ . - Points - Player,
                           data = wr_relevant_transformed)

par(mfrow = c(2, 2))
plot(wr_transformed_model,
     pch = 19,
     col = rgb(0, 0, 0, alpha = 0.25),
     add.smooth = FALSE)

```



```
summary(wr_transformed_model)
```

Call:

```
lm(formula = New_Points ~ . - Points - Player, data = wr_relevant_transformed)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.69191	-0.58633	0.07789	0.59310	1.38758

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	2.966194	4.414616	0.672	0.5027
X40YD	-0.310414	0.702097	-0.442	0.6590


```

Vertical      0.006149   0.025180   0.244   0.8074
Broad.Jump    -0.005236   0.013178  -0.397   0.6917
X3Cone        -0.179449   0.367733  -0.488   0.6263
Shuttle       0.937100   0.497067   1.885   0.0613 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7616 on 151 degrees of freedom
(117 observations deleted due to missingness)
Multiple R-squared:  0.0278,    Adjusted R-squared:  -0.004393
F-statistic: 0.8635 on 5 and 151 DF,  p-value: 0.5072

```

2.4.3 Model Selection

```

wr_best_subsets <- regsubsets(New_Points ~ . - Points - Player,
                              data = wr_relevant_transformed,
                              nvmax = 5)
summary(wr_best_subsets)

```

```

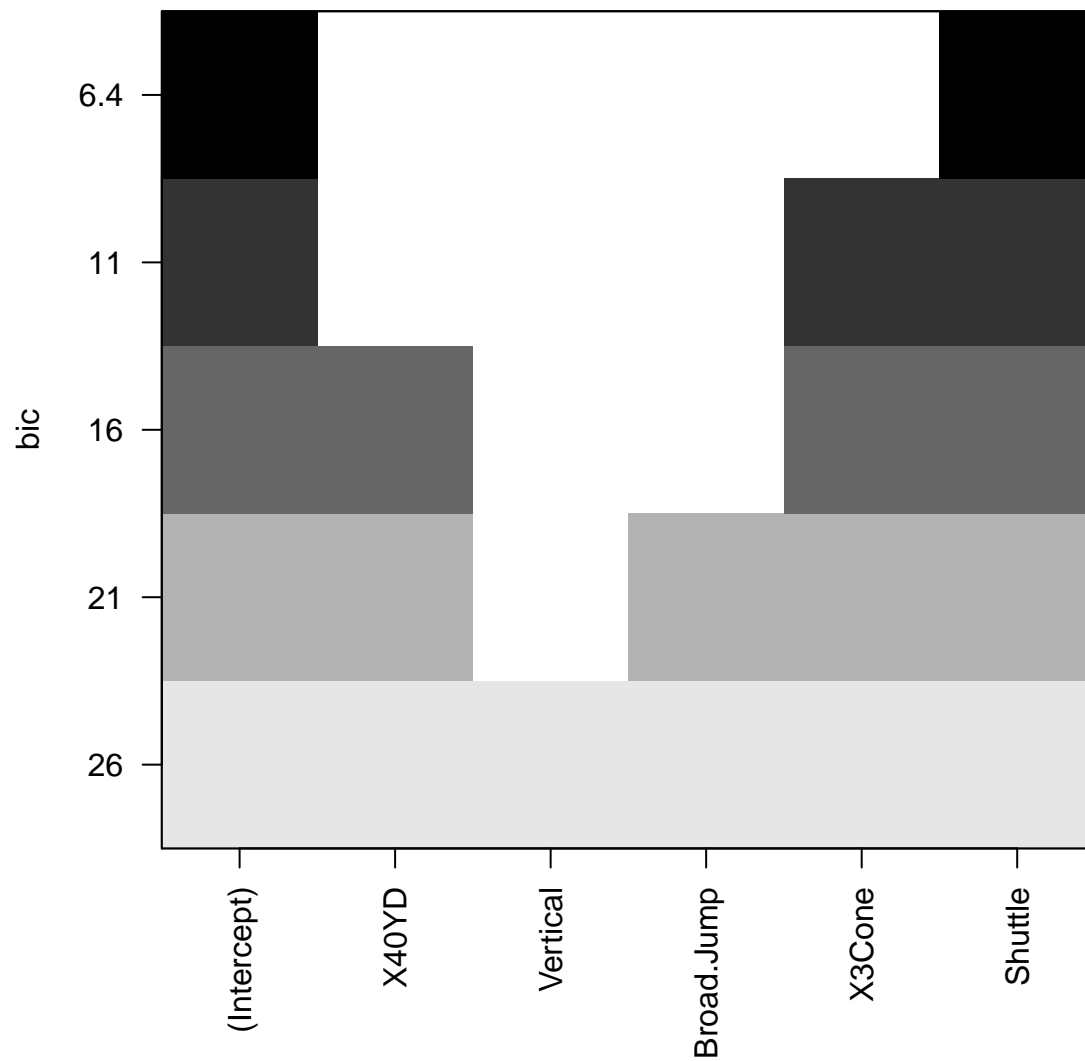
Subset selection object
Call: regsubsets.formula(New_Points ~ . - Points - Player, data = wr_relevant_transformed,
                          nvmax = 5)
5 Variables (and intercept)
      Forced in Forced out
X40YD      FALSE      FALSE
Vertical    FALSE      FALSE
Broad.Jump  FALSE      FALSE
X3Cone      FALSE      FALSE
Shuttle     FALSE      FALSE
1 subsets of each size up to 5
Selection Algorithm: exhaustive
      X40YD Vertical Broad.Jump X3Cone Shuttle
1 ( 1 ) " " " " " " " "*"
2 ( 1 ) " " " " " " "*" "*"
3 ( 1 ) "*" " " " " "*" "*"
4 ( 1 ) "*" " " "*" "*" "*"
5 ( 1 ) "*" "*" "*" "*" "*"

```

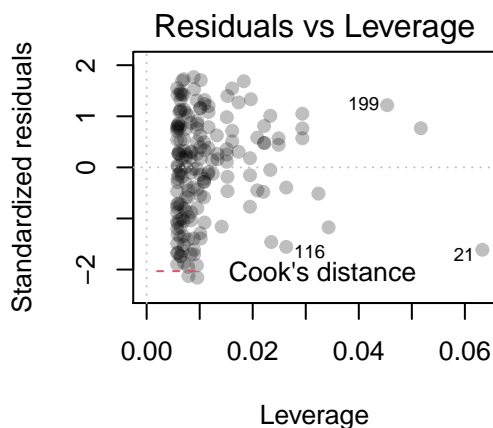
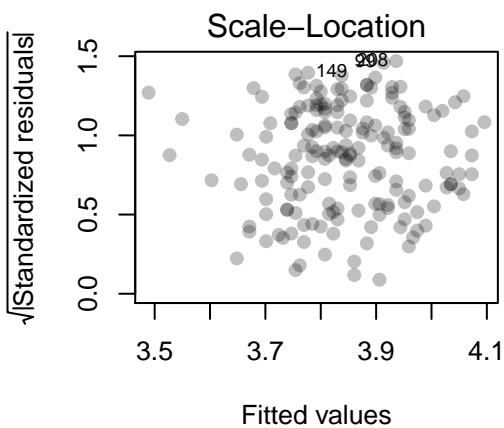
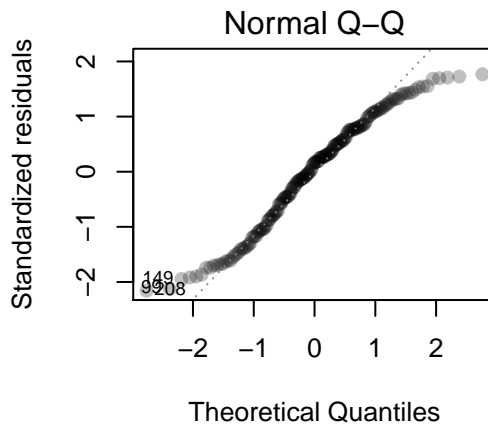
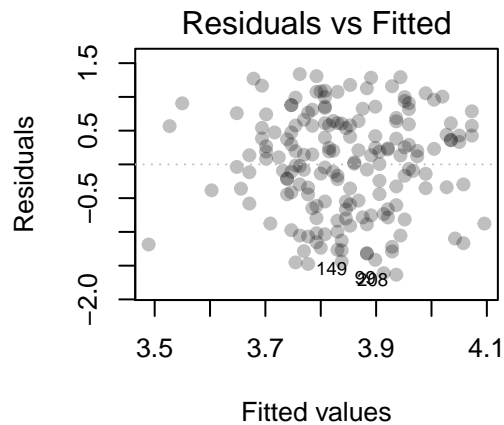
```

plot(wr_best_subsets)

```



```
final_wr_model <- lm(New_Points ~ Shuttle, data = wr_relevant_transformed)
par(mfrow = c(2, 2))
plot(final_wr_model,
     pch = 19,
     col = rgb(0, 0, 0, alpha = 0.25),
     add.smooth = FALSE)
```



```
summary(final_wr_model)
```

Call:

```
lm(formula = New_Points ~ Shuttle, data = wr_relevant_transformed)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.6338	-0.5912	0.1268	0.5909	1.3378

Coefficients:

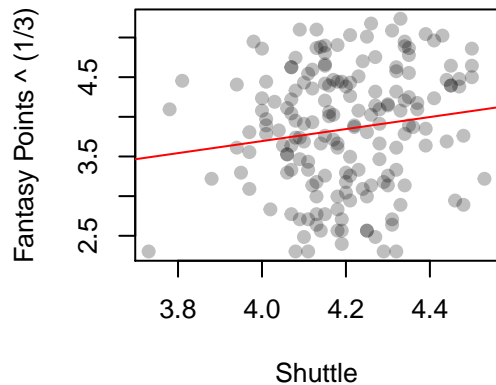
	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.6612	1.6333	0.405	0.6861
Shuttle	0.7581	0.3887	1.950	0.0527

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

Residual standard error: 0.7604 on 172 degrees of freedom

```
(100 observations deleted due to missingness)
Multiple R-squared:  0.02164,    Adjusted R-squared:  0.01595
F-statistic: 3.804 on 1 and 172 DF,  p-value: 0.05274
```

```
plot(New_Points ~ Shuttle, data = wr_relevant_transformed,
     ylab = "Fantasy Points ^ (1/3)",
     pch = 19,
     col = rgb(0, 0, 0, alpha = 0.25))
abline(final_wr_model, col = "red")
```



2.5 The Tight End

```
nrow(tes)
```

```
[1] 102
```

```
total_nas(tes)
```

Year	Player	Pos	X40YD	Vertical	BenchReps	Broad.Jump
0	0	0	3	22	23	25
X3Cone	Shuttle	Games	Points			
30	30	0	0			

```
total_nas(tes) / nrow(tes)
```

Year	Player	Pos	X40YD	Vertical	BenchReps	Broad.Jump
0.00000000	0.00000000	0.00000000	0.02941176	0.21568627	0.22549020	0.24509804
X3Cone	Shuttle	Games	Points			
0.29411765	0.29411765	0.00000000	0.00000000			

```
names(tes)[total_nas(tes) / nrow(tes) >= 0.4]
```

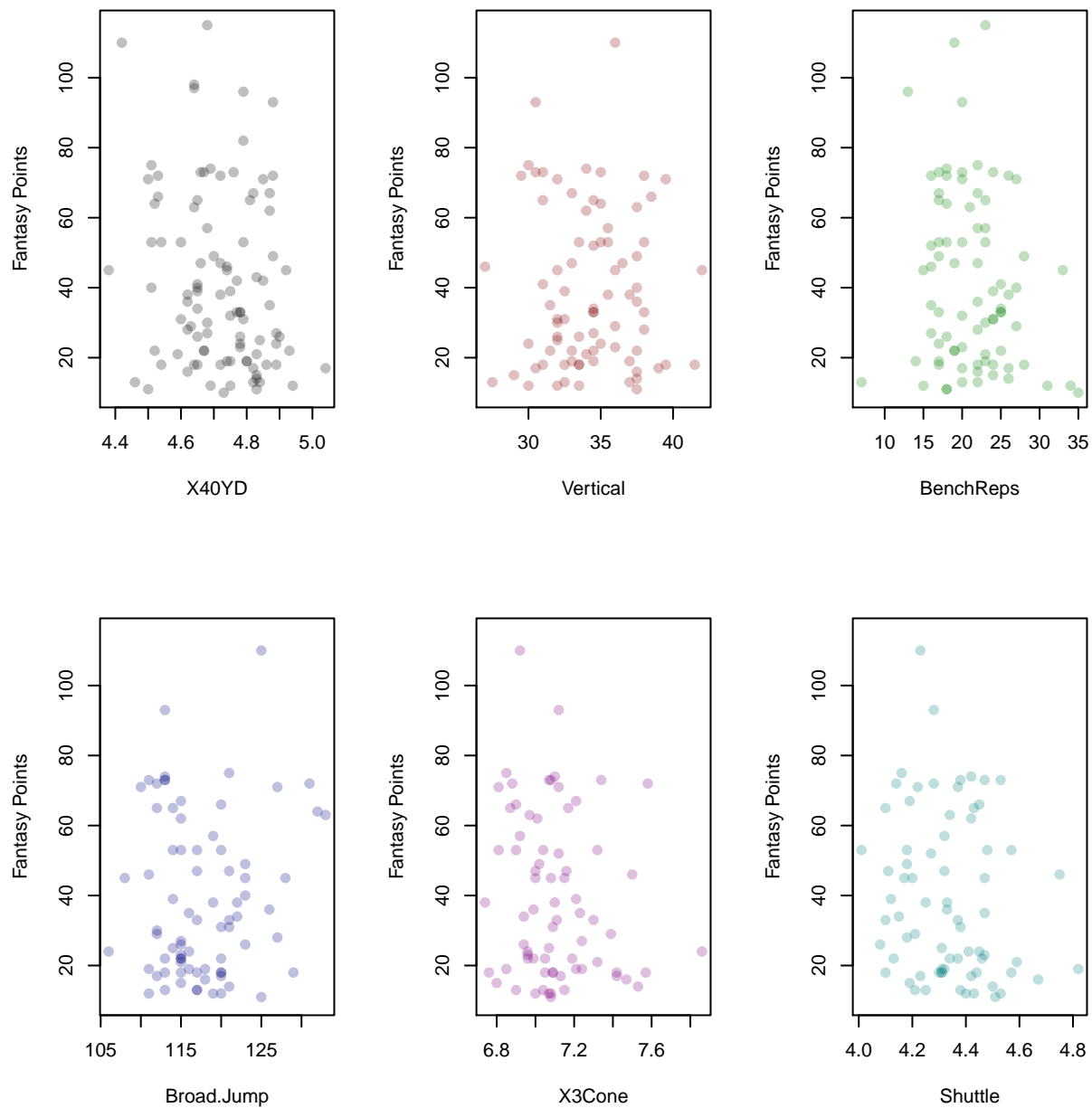
```
character(0)
```

```

te_rel_vars <- tes %>% dplyr::select(Player,
                                   X40YD:Shuttle,
                                   Points)

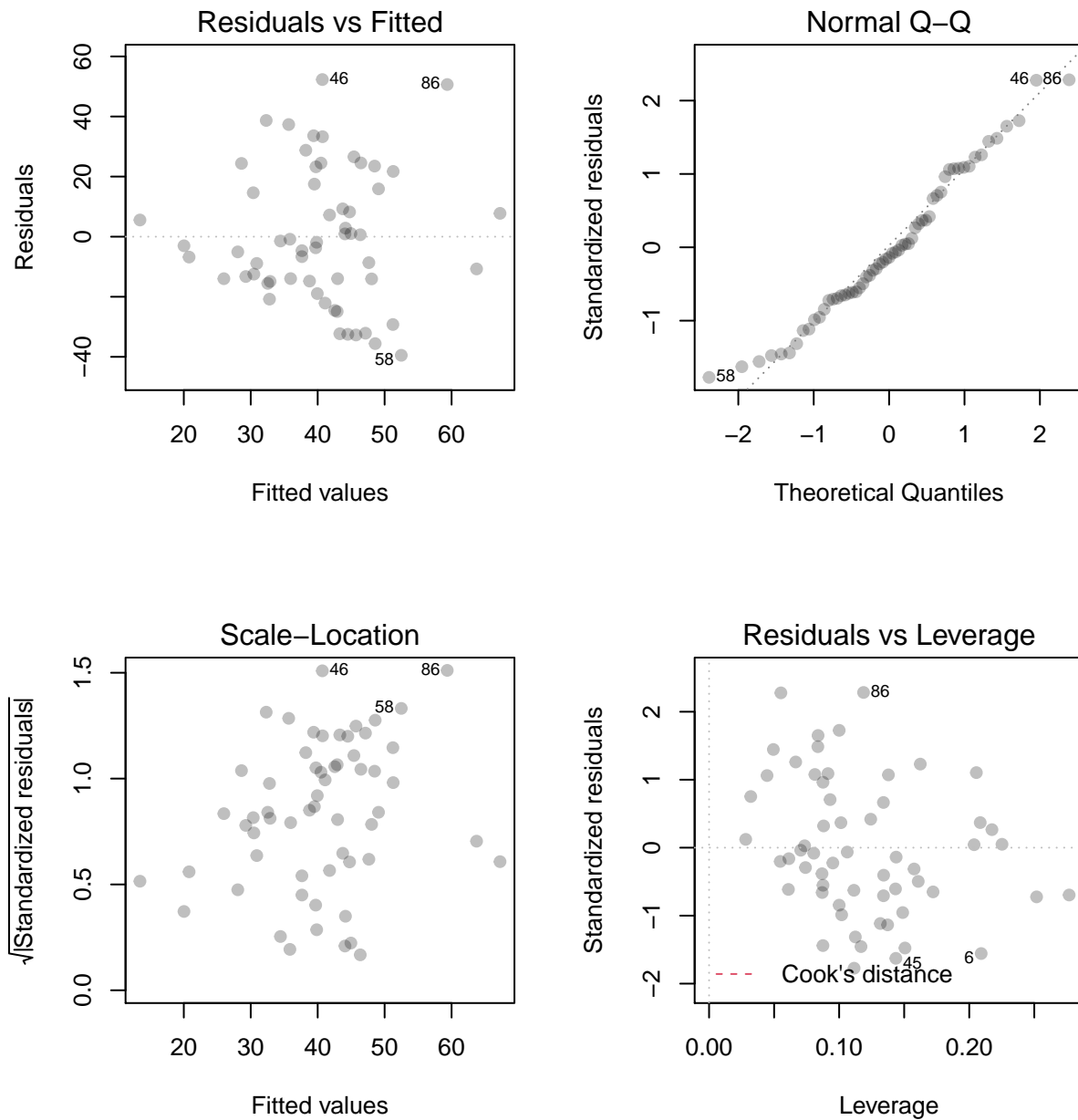
par(mfrow = c(2, 3))
plot(Points ~ X40YD, data = te_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[1])
plot(Points ~ Vertical, data = te_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[2])
plot(Points ~ BenchReps, data = te_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[3])
plot(Points ~ Broad.Jump, data = te_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[4])
plot(Points ~ X3Cone, data = te_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[5])
plot(Points ~ Shuttle, data = te_rel_vars,
     ylab = "Fantasy Points",
     pch = 19, col = color_scheme[6])

```



2.5.1 A Model

```
te_model <- lm(Points ~ . - Player, data = te_rel_vars)
par(mfrow = c(2, 2))
plot(te_model,
     pch = 19,
     col = rgb(0, 0, 0, alpha = 0.25),
     add.smooth = FALSE)
```



```
summary(te_model)
```

Call:

```
lm(formula = Points ~ . - Player, data = te_rel_vars)
```

Residuals:

	Min	1Q	Median	3Q	Max
Residuals	-39.49	-14.42	-3.03	16.70	52.30

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	505.1175	214.3477	2.357	0.0222 *
X40YD	-60.1252	28.3699	-2.119	0.0389 *

Vertical	-2.2412	1.3437	-1.668	0.1014
BenchReps	-0.4947	0.6984	-0.708	0.4819
Broad.Jump	0.1628	0.8120	0.200	0.8419
X3Cone	3.5874	17.0011	0.211	0.8337
Shuttle	-31.9374	20.6852	-1.544	0.1287

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

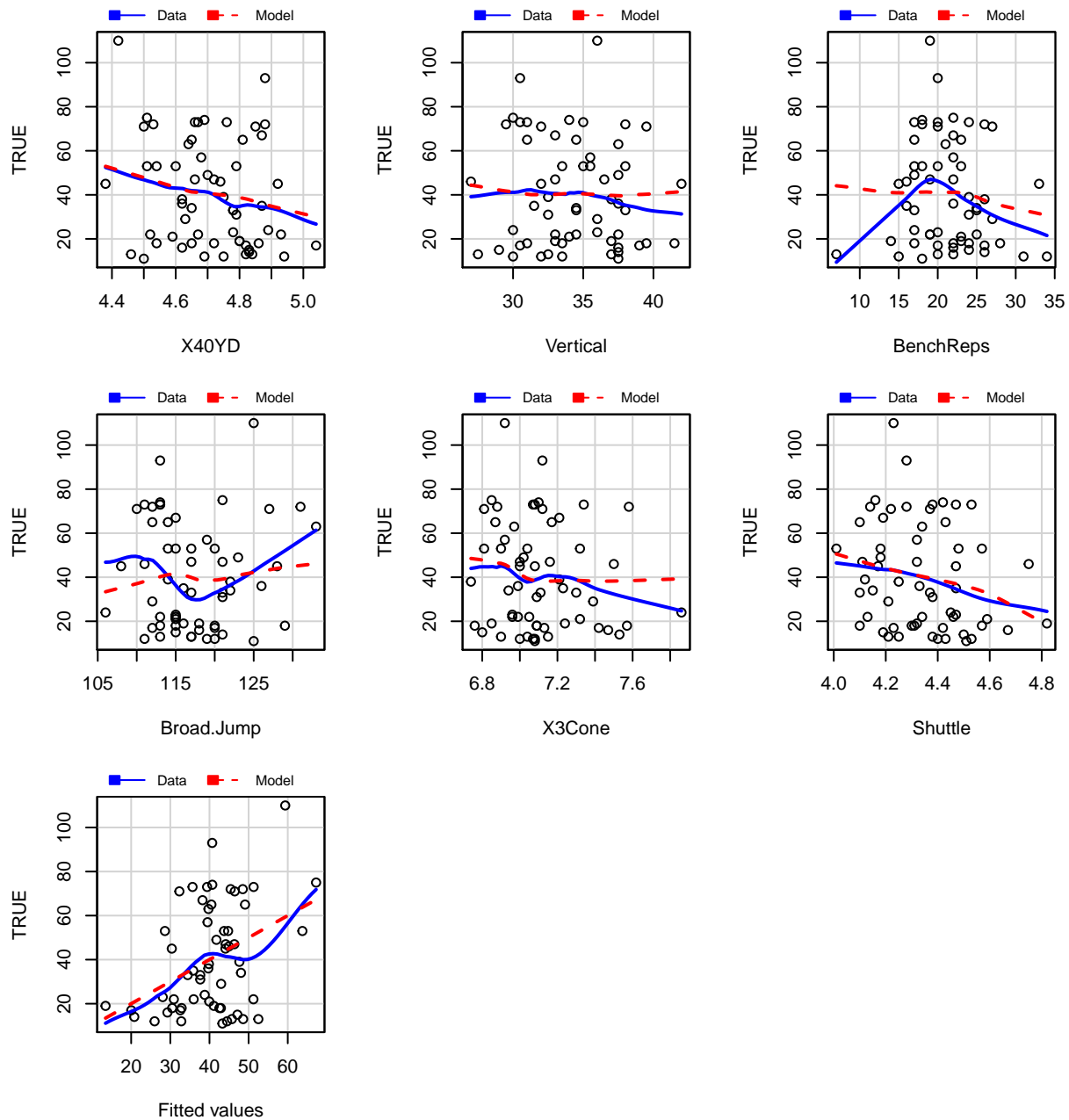
Residual standard error: 23.63 on 52 degrees of freedom
 (43 observations deleted due to missingness)

Multiple R-squared: 0.1574, Adjusted R-squared: 0.06021

F-statistic: 1.619 on 6 and 52 DF, p-value: 0.1605

```
mmpr(te_model)
```


Marginal Model Plots



2.5.2 A Transformation

```
summary(powerTransform(Points ~ . - Player, data = te_rel_vars))
```

```
bcPower Transformation to Normality
  Est Power Rounded Pwr Wald Lwr Bnd Wald Up Bnd
Y1    0.1494          0   -0.3088      0.6077
```

Likelihood ratio test that transformation parameter is equal to 0

```
(log transformation)
              LRT df    pval
LR test, lambda = (0) 0.4073206  1 0.52333

Likelihood ratio test that no transformation is needed
              LRT df    pval
LR test, lambda = (1) 13.24616  1 0.00027314
```

```
summary(powerTransform(cbind(X40YD, Vertical, BenchReps,
                             Broad.Jump, X3Cone, Shuttle) ~ 1,
                             data = te_rel_vars))
```

Warning in estimateTransform.default(X, Y, weights, family, ...): Convergence failure: return code = 52

```
bcPower Transformations to Multinormality
```

	Est Power	Rounded Pwr	Wald Lwr Bnd	Wald Up Bnd
X40YD	2.8043	1.00	-4.2388	9.8474
Vertical	1.8463	1.00	-0.2157	3.9083
BenchReps	0.9178	1.00	0.2382	1.5974
Broad.Jump	-3.6491	-1.00	-6.6295	-0.6686
X3Cone	-11.7318	-11.73	-12.1793	-11.2842
Shuttle	-3.8940	1.00	-9.5811	1.7932

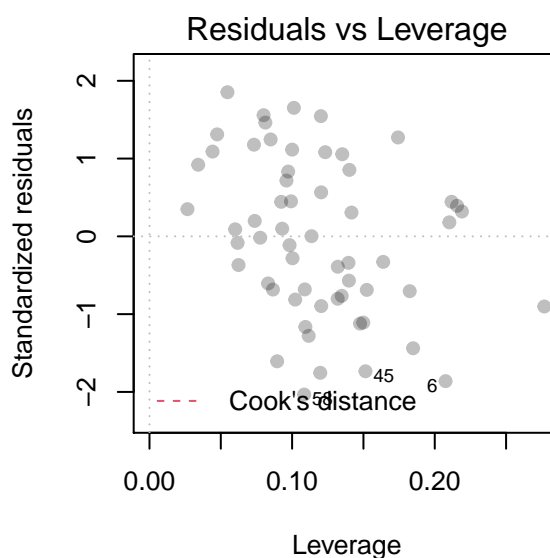
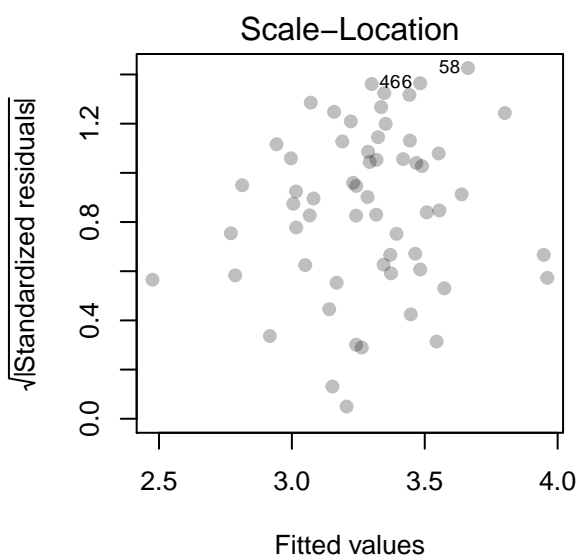
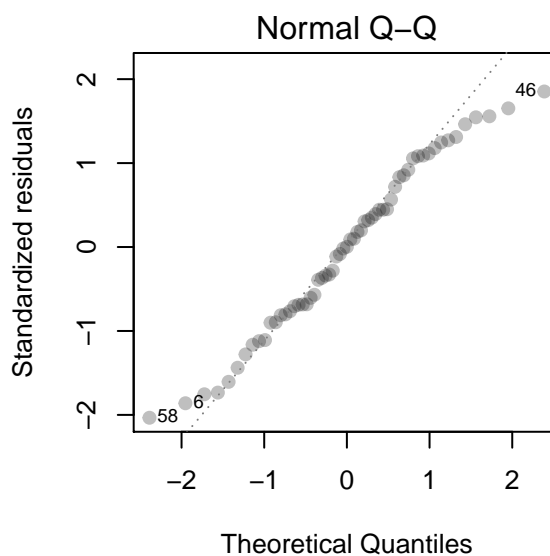
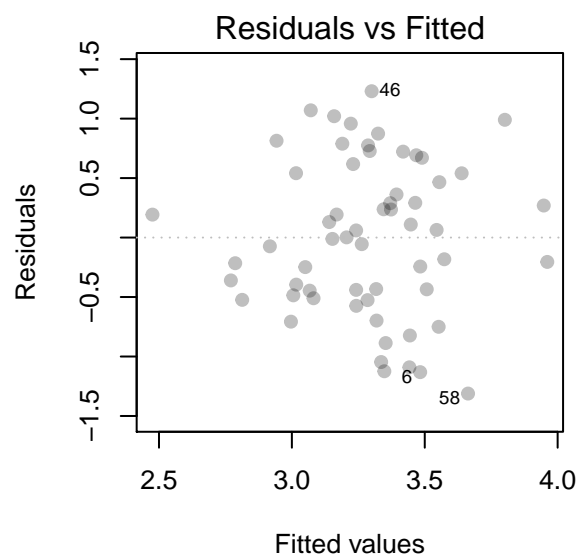
Likelihood ratio test that transformation parameters are equal to 0
(all log transformations)

```
              LRT df    pval
LR test, lambda = (0 0 0 0 0 0) 29.99437  6 3.9406e-05
```

Likelihood ratio test that no transformations are needed

```
              LRT df    pval
LR test, lambda = (1 1 1 1 1 1) 24.35547  6 0.00044918
```

```
te_relevant_transformed <- te_rel_vars %>% mutate("New_Points" = Points^(1/3),
                                                  "New_3_Cone" = X3Cone^(-11.81),
                                                  "New_Broad" = Broad.Jump^(-1))
te_transformed_model <- lm(New_Points ~ . - Points - X3Cone - Broad.Jump - Player,
                           data = te_relevant_transformed)
par(mfrow = c(2, 2))
plot(te_transformed_model,
     pch = 19,
     col = rgb(0, 0, 0, alpha = 0.25),
     add.smooth = FALSE)
```



```
summary(te_transformed_model)
```

Call:

```
lm(formula = New_Points ~ . - Points - X3Cone - Broad.Jump -  
    Player, data = te_relevant_transformed)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.31160	-0.46603	0.00159	0.54078	1.23036

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	1.856e+01	6.279e+00	2.955	0.00469 **

```

X40YD      -1.760e+00  8.324e-01  -2.114  0.03931 *
Vertical    -5.006e-02  3.898e-02  -1.284  0.20470
BenchReps   -1.590e-02  2.024e-02  -0.786  0.43571
Shuttle     -1.143e+00  6.169e-01  -1.853  0.06962 .
New_3_Cone  -2.349e+09  3.810e+09  -0.617  0.54021
New_Broad    3.032e+01  3.329e+02   0.091  0.92779
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6829 on 52 degrees of freedom
(43 observations deleted due to missingness)
Multiple R-squared:  0.1517,    Adjusted R-squared:  0.05379
F-statistic:  1.55 on 6 and 52 DF,  p-value: 0.1808

```

2.5.3 Model Selection

```

te_best_subsets <- regsubsets(New_Points ~ . - Points - X3Cone - Broad.Jump - Player,
                              data = te_relevant_transformed,
                              nvmax = 6)

```

Warning in leaps.exhaustive(a, really.big): XHAUST returned error code -999

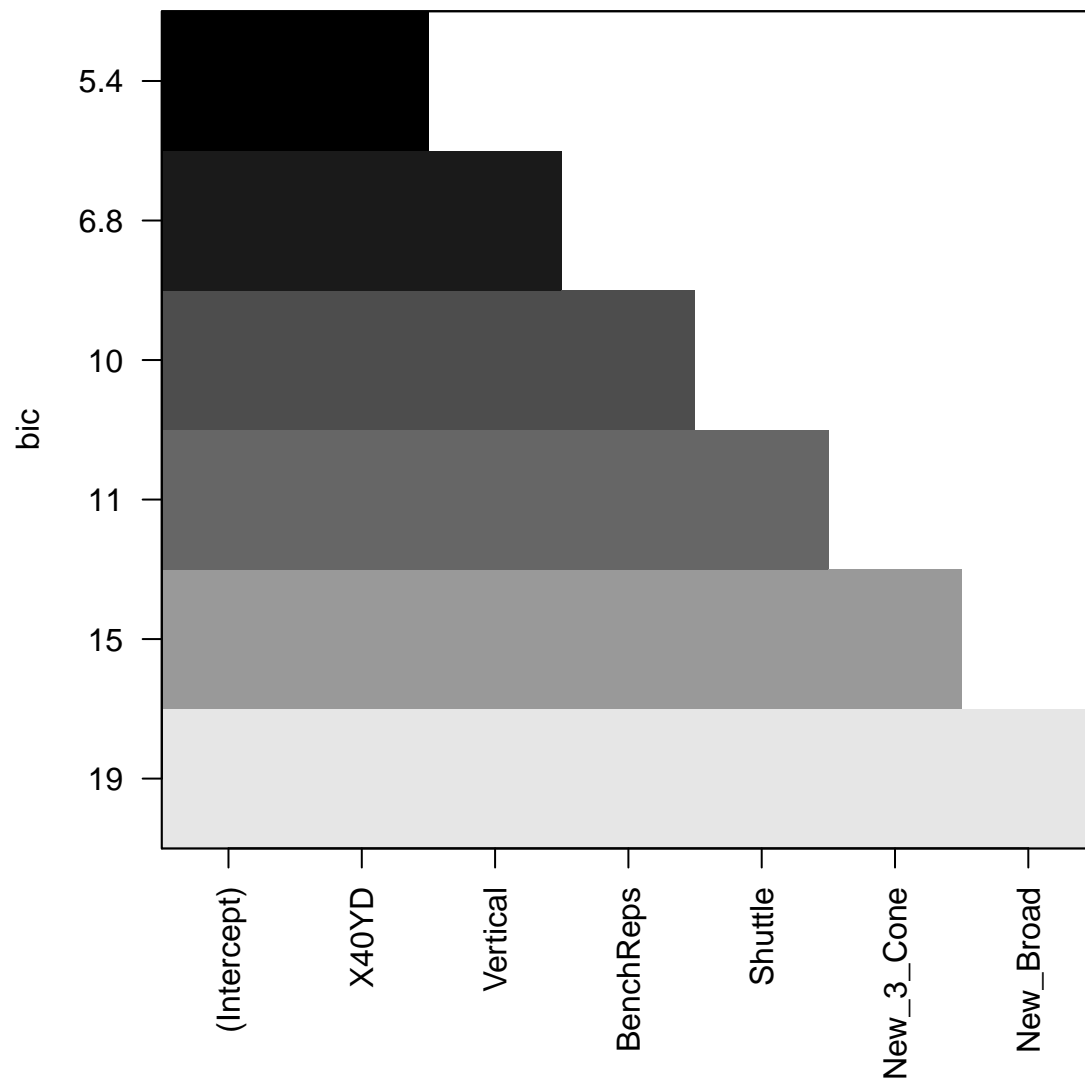
```
summary(te_best_subsets)
```

```

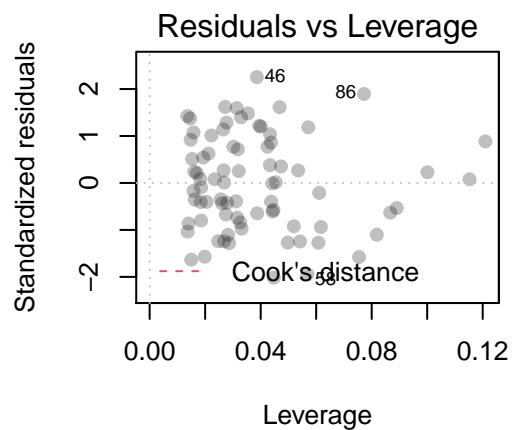
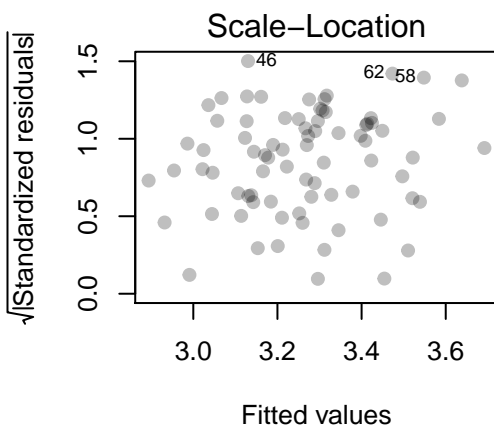
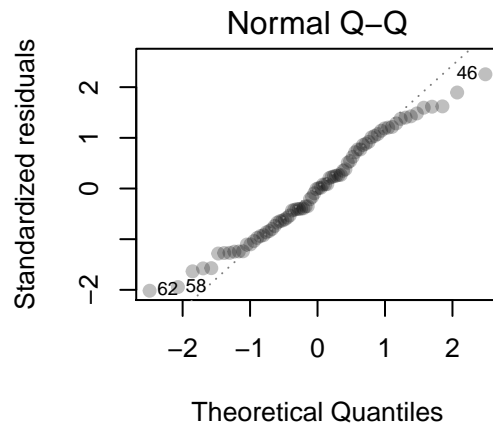
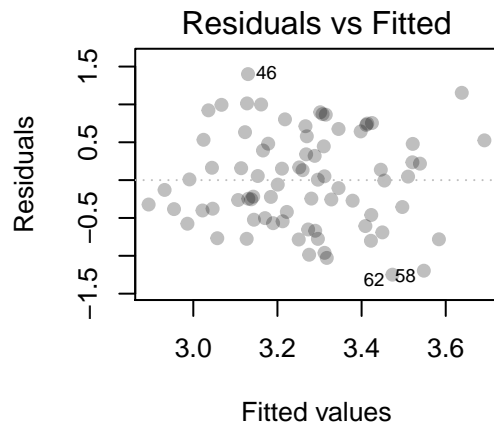
Subset selection object
Call: regsubsets.formula(New_Points ~ . - Points - X3Cone - Broad.Jump -
      Player, data = te_relevant_transformed, nvmax = 6)
6 Variables (and intercept)
      Forced in Forced out
X40YD      FALSE      FALSE
Vertical    FALSE      FALSE
BenchReps   FALSE      FALSE
Shuttle     FALSE      FALSE
New_3_Cone  FALSE      FALSE
New_Broad   FALSE      FALSE
1 subsets of each size up to 6
Selection Algorithm: exhaustive
      X40YD Vertical BenchReps Shuttle New_3_Cone New_Broad
1  ( 1 ) "*"      " "      " "      " "      " "      " "
2  ( 1 ) "*"      "*"      " "      " "      " "      " "
3  ( 1 ) "*"      "*"      "*"      " "      " "      " "
4  ( 1 ) "*"      "*"      "*"      "*"      " "      " "
5  ( 1 ) "*"      "*"      "*"      "*"      "*"      " "
6  ( 1 ) "*"      "*"      "*"      "*"      "*"      "*"

```

```
plot(te_best_subsets)
```



```
final_te_model <- lm(New_Points ~ X40YD + Vertical,
                     data = te_relevant_transformed)
par(mfrow = c(2, 2))
plot(final_te_model,
     pch = 19,
     col = rgb(0, 0, 0, alpha = 0.25),
     add.smooth = FALSE)
```



```
summary(final_te_model)
```

Call:

```
lm(formula = New_Points ~ X40YD + Vertical, data = te_relevant_transformed)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-1.24937	-0.49164	-0.00008	0.51420	1.40056

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	11.28489	3.48527	3.238	0.00179 **
X40YD	-1.47657	0.61935	-2.384	0.01965 *
Vertical	-0.03112	0.02692	-1.156	0.25136

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

Residual standard error: 0.6337 on 75 degrees of freedom
 (24 observations deleted due to missingness)
 Multiple R-squared: 0.07053, Adjusted R-squared: 0.04575
 F-statistic: 2.846 on 2 and 75 DF, p-value: 0.06439

3 Final Models

3.1 Quarterback

```
summary(final_qb_model)
```

Call:
 lm(formula = New_Points ~ Broad.Jump + Shuttle, data = qb_relevant_transformed)

Residuals:

	Min	1Q	Median	3Q	Max
	-1.11619	-0.26852	-0.02889	0.36250	0.85084

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-5.85609	3.15626	-1.855	0.07249 .
Broad.Jump	0.02144	0.01337	1.603	0.11847
Shuttle	1.98119	0.56961	3.478	0.00144 **

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

Residual standard error: 0.4722 on 33 degrees of freedom
 (18 observations deleted due to missingness)
 Multiple R-squared: 0.2778, Adjusted R-squared: 0.234
 F-statistic: 6.346 on 2 and 33 DF, p-value: 0.004655

$$\log(\hat{FantasyPoints}) = -5.856 + 0.02(BroadJump) + 1.98(Shuttle)$$

3.2 Running Back

```
summary(final_rb_model)
```

Call:
 lm(formula = New_Points ~ Broad.Jump + X40YD, data = rb_relevant_transformed)

Residuals:

	Min	1Q	Median	3Q	Max
	-1.89030	-0.55716	0.05755	0.67639	1.53941

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	3.95120	4.06020	0.973	0.3319
Broad.Jump	0.02457	0.01294	1.899	0.0592 .
X40YD	-0.60625	0.71524	-0.848	0.3979

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

Residual standard error: 0.8193 on 168 degrees of freedom
(44 observations deleted due to missingness)
Multiple R-squared: 0.0363, Adjusted R-squared: 0.02483
F-statistic: 3.164 on 2 and 168 DF, p-value: 0.04477

$$\log(\widehat{FantasyPoints}) = 3.95 + 0.0246(BroadJump) - 0.606(FortyYardTime)$$

3.3 Wide Receiver

```
summary(final_wr_model)
```

Call:
lm(formula = New_Points ~ Shuttle, data = wr_relevant_transformed)

Residuals:

Min	1Q	Median	3Q	Max
-1.6338	-0.5912	0.1268	0.5909	1.3378

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.6612	1.6333	0.405	0.6861
Shuttle	0.7581	0.3887	1.950	0.0527 .

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

Residual standard error: 0.7604 on 172 degrees of freedom
(100 observations deleted due to missingness)
Multiple R-squared: 0.02164, Adjusted R-squared: 0.01595
F-statistic: 3.804 on 1 and 172 DF, p-value: 0.05274

$$\log(\widehat{FantasyPoints}) = 0.66 + 0.76(ShuttleTime)$$

3.4 Tight End

```
summary(final_te_model)
```

Call:


```
lm(formula = New_Points ~ X40YD + Vertical, data = te_relevant_transformed)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-1.24937	-0.49164	-0.00008	0.51420	1.40056

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	11.28489	3.48527	3.238	0.00179 **
X40YD	-1.47657	0.61935	-2.384	0.01965 *
Vertical	-0.03112	0.02692	-1.156	0.25136

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

Residual standard error: 0.6337 on 75 degrees of freedom

(24 observations deleted due to missingness)

Multiple R-squared: 0.07053, Adjusted R-squared: 0.04575

F-statistic: 2.846 on 2 and 75 DF, p-value: 0.06439

$$(\hat{FantasyPoints})^{\frac{1}{3}} = 11.28 - 1.48(FortyYardTime) - 0.03(Vertical)$$

4 How Good Are These Rookies?

4.1 Quarterbacks

```
rookie_qb <- qbs %>% summarise(meanFPts = mean(Points)) %>% as.numeric
rookie_qb
```

```
[1] 158.963
```

```
rookie_qb / 16
```

```
[1] 9.935185
```

```
with(qbs, tapply(Points, Year, mean))
```

	2001	2002	2003	2004	2005	2006	2007	2008
	91.7500	108.3333	117.5000	111.0000	58.0000	149.6667	82.0000	192.5000
	2009	2010	2011	2012	2013	2014	2015	2016
	121.6667	111.5000	208.5000	244.2000	176.0000	182.0000	244.5000	196.3333
	2017	2018	2019					
	155.5000	182.2500	173.6667					

4.1.1 Change Over Time

```

ppy <- with(qbs, tapply(Points, Year, mean))
year <- 2001:2019
plot(ppy ~ year,
     xlab = "Year",
     ylab = "Average Points",
     main = "Average Rookie QB Fantasy Points \n by Year",
     pch = 19, col = rgb(0.5, 0, 0, alpha = 0.5))
qb_yr_lm <- lm(ppy ~ year)
summary(qb_yr_lm)

```

Call:

```
lm(formula = ppy ~ year)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-63.057	-30.875	3.459	15.719	78.433

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-12685.089	3426.860	-3.702	0.00177 **
year	6.387	1.705	3.746	0.00161 **

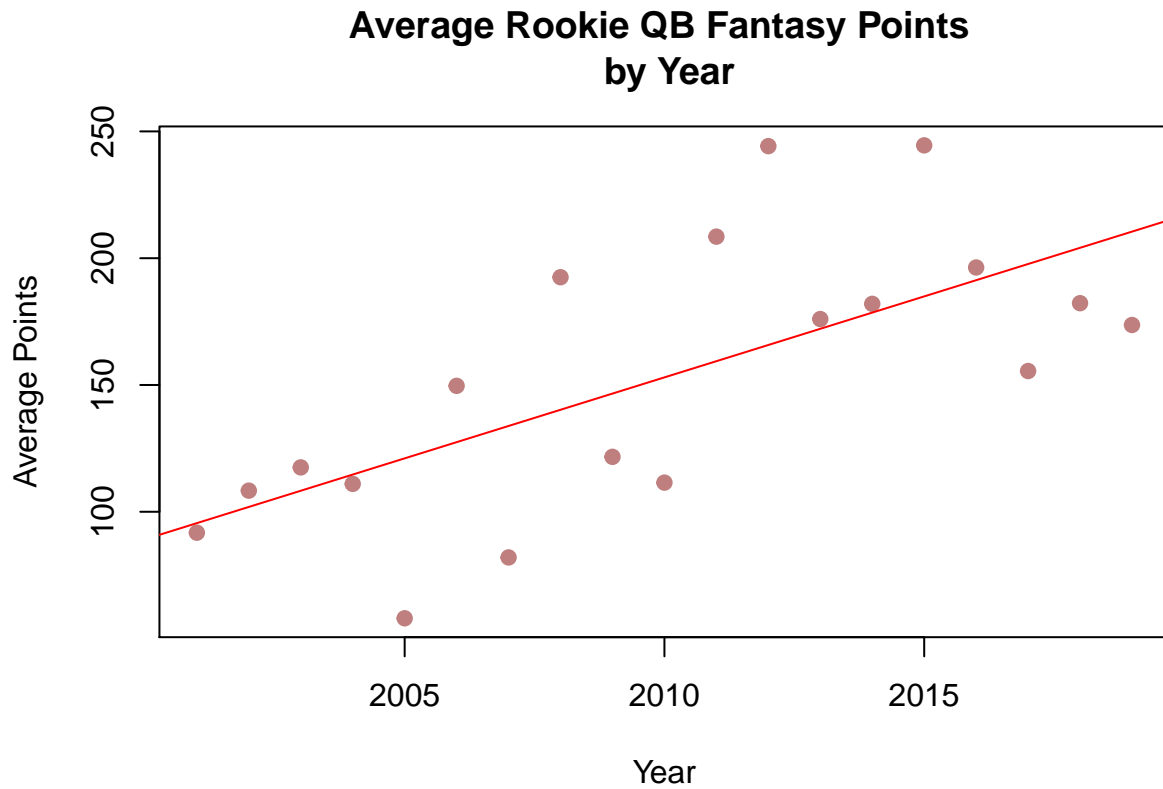
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 40.7 on 17 degrees of freedom

Multiple R-squared: 0.4522, Adjusted R-squared: 0.42

F-statistic: 14.03 on 1 and 17 DF, p-value: 0.001608

```
abline(qb_yr_lm, col = "red")
```



4.2 Running Backs

```
rookie_rb <- rbs %>% summarise(meanFPts = mean(Points)) %>% as.numeric
rookie_rb
```

```
[1] 86.71163
```

```
rookie_rb / 16
```

```
[1] 5.419477
```

```
with(rbs, tapply(Points, Year, mean))
```

2000	2001	2002	2003	2004	2005	2006	2007
58.33333	111.22222	62.92857	63.57143	84.14286	72.63636	111.50000	76.22222
2008	2009	2010	2011	2012	2013	2014	2015
127.09091	83.37500	78.44444	56.66667	104.90909	91.08333	71.13333	106.76923
2016	2017	2018	2019				
82.66667	111.14286	99.33333	70.90000				

4.2.1 Change Over Time

```

ppy <- with(rbs, tapply(Points, Year, mean))
year <- 2000:2019
plot(ppy ~ year,
     xlab = "Year",
     ylab = "Average Points",
     main = "Average Rookie RB Fantasy Points \n by Year",
     pch = 19, col = rgb(0, 0.5, 0, alpha = 0.5))
rb_yr_lm <- lm(ppy ~ year)
summary(rb_yr_lm)

```

Call:

```
lm(formula = ppy ~ year)
```

Residuals:

Min	1Q	Median	3Q	Max
-30.696	-17.512	-5.246	16.430	42.046

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-1466.6793	1608.0049	-0.912	0.374
year	0.7728	0.8002	0.966	0.347

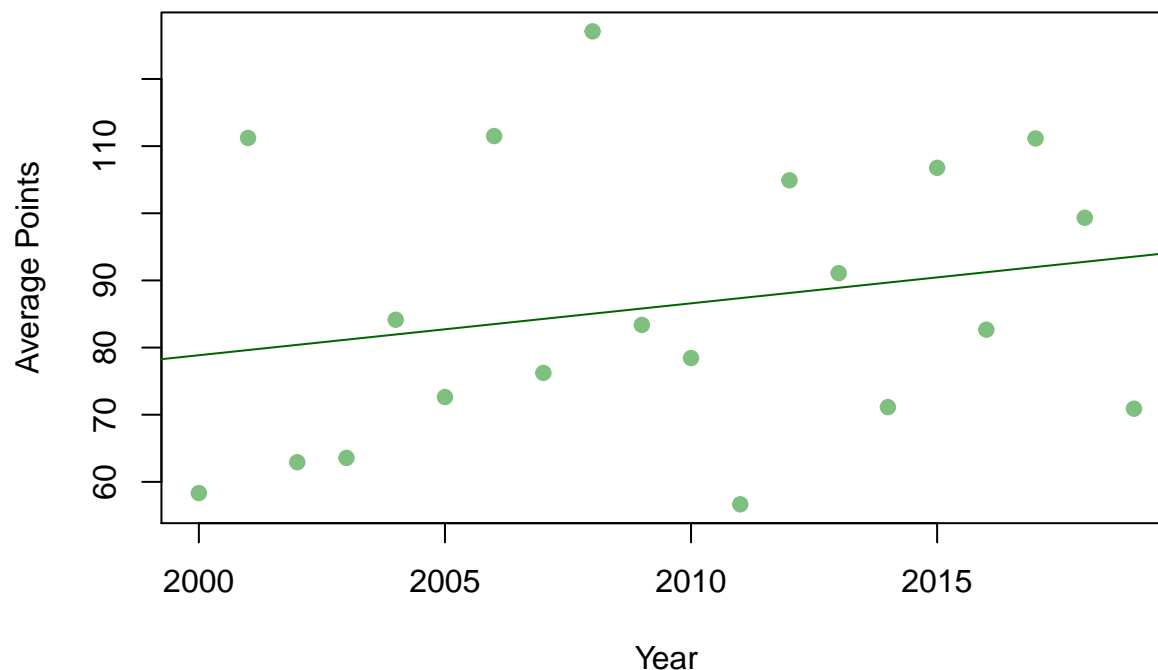
Residual standard error: 20.64 on 18 degrees of freedom

Multiple R-squared: 0.04926, Adjusted R-squared: -0.003559

F-statistic: 0.9326 on 1 and 18 DF, p-value: 0.347

```
abline(rb_yr_lm, col = "dark green")
```

Average Rookie RB Fantasy Points by Year



4.3 Wide Receivers

```
rookie_wr <- wrs %>% summarise(meanFPts = mean(Points)) %>% as.numeric
rookie_wr
```

```
[1] 61.94891
```

```
rookie_wr / 16
```

```
[1] 3.871807
```

```
with(wrs, tapply(Points, Year, mean))
```

```

 2000    2001    2002    2003    2004    2005    2006    2007
40.68750 50.46154 60.53846 58.75000 68.21429 48.50000 57.00000 60.60000
 2008    2009    2010    2011    2012    2013    2014    2015
63.70000 69.66667 48.22727 84.18182 53.11111 73.93750 78.52632 55.40000
 2016    2017    2018    2019
57.16667 46.76923 84.66667 96.90909

```

4.3.1 Change Over Time

```

ppy <- with(wrs, tapply(Points, Year, mean))
year <- 2000:2019
plot(ppy ~ year,
     xlab = "Year",
     ylab = "Average Points",
     main = "Average Rookie WR Fantasy Points \n by Year",
     pch = 19, col = rgb(0, 0, 0.5, alpha = 0.5))
wr_yr_lm <- lm(ppy ~ year)
summary(wr_yr_lm)

```

Call:

```
lm(formula = ppy ~ year)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-25.516	-10.881	1.815	8.088	22.108

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-2464.8577	1014.4530	-2.430	0.0258 *
year	1.2579	0.5048	2.492	0.0227 *

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

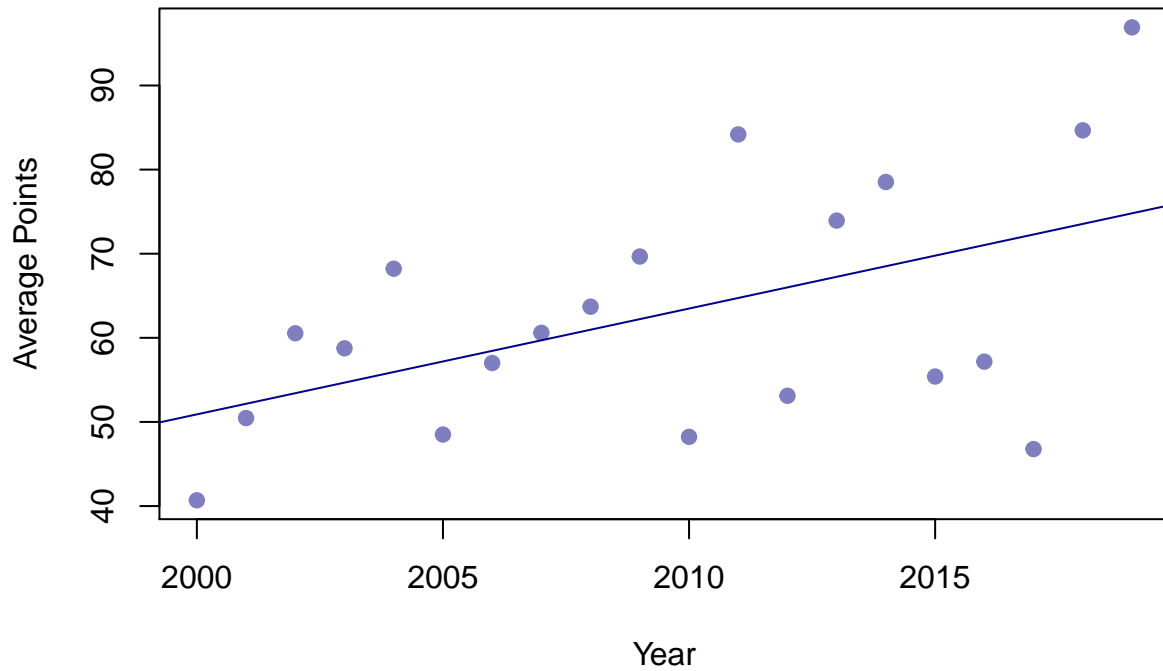
Residual standard error: 13.02 on 18 degrees of freedom

Multiple R-squared: 0.2565, Adjusted R-squared: 0.2152

F-statistic: 6.209 on 1 and 18 DF, p-value: 0.0227

```
abline(wr_yr_lm, col = "dark blue")
```

Average Rookie WR Fantasy Points by Year



4.4 Tight Ends

```
rookie_te <- tes %>% summarise(meanFPts = mean(Points)) %>% as.numeric
rookie_te
```

```
[1] 41.51961
```

```
rookie_te / 16
```

```
[1] 2.594975
```

```
with(tes, tapply(Points, Year, mean))
```

2000	2001	2002	2003	2004	2005	2006	2007
26.25000	26.00000	54.00000	28.42857	38.00000	50.75000	35.57143	30.57143
2008	2009	2010	2011	2012	2013	2014	2015
48.25000	22.00000	63.62500	30.00000	40.33333	44.44444	29.00000	25.83333
2016	2017	2018	2019				
53.66667	58.14286	39.00000	52.80000				

4.4.1 Change Over Time

```

ppy <- with(tes, tapply(Points, Year, mean))
year <- 2000:2019
plot(ppy ~ year,
     xlab = "Year",
     ylab = "Average Points",
     main = "Average Rookie TE Fantasy Points \n by Year",
     pch = 19, col = rgb(0.5, 0, 0.5, alpha = 0.5))
te_yr_lm <- lm(ppy ~ year)
summary(te_yr_lm)

```

Call:

```
lm(formula = ppy ~ year)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-17.655	-7.746	-1.549	9.438	23.459

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-1295.6538	958.1454	-1.352	0.193
year	0.6646	0.4768	1.394	0.180

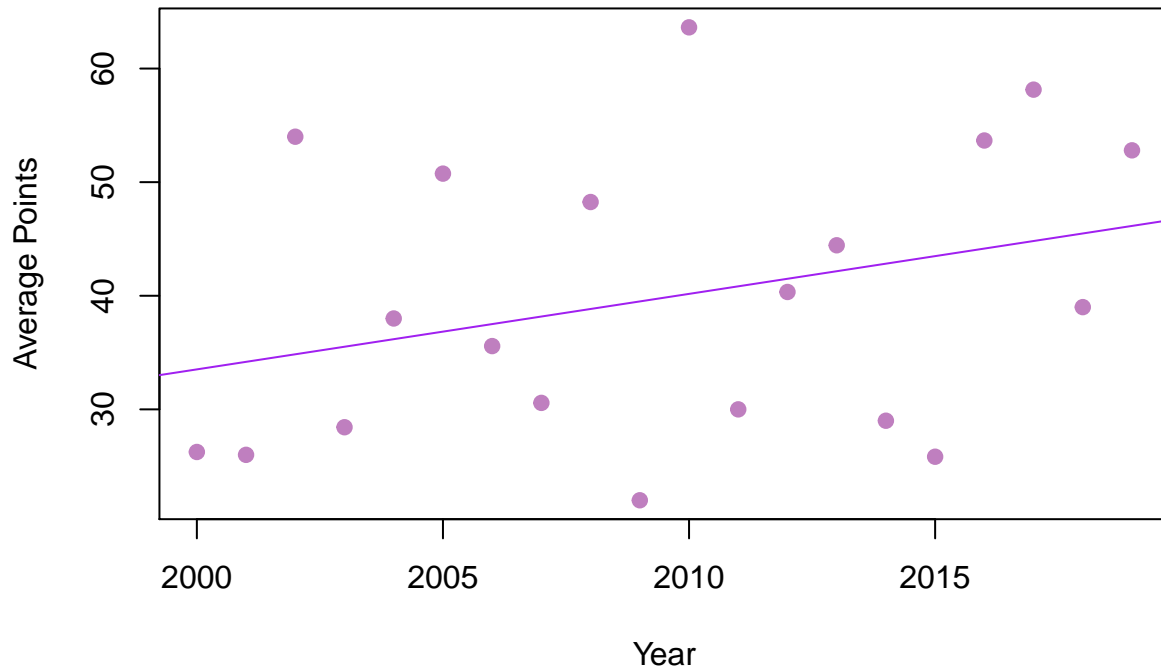
Residual standard error: 12.3 on 18 degrees of freedom

Multiple R-squared: 0.09742, Adjusted R-squared: 0.04727

F-statistic: 1.943 on 1 and 18 DF, p-value: 0.1803

```
abline(te_yr_lm, col = "purple")
```


Average Rookie TE Fantasy Points by Year



5 Predictions

```
combine2020 <- read.csv("2020combine.csv")
```

5.1 Quarterbacks

```
qbs_2020 <- filter(combine2020, Pos == "QB")
final_qb_model
```

Call:

```
lm(formula = New_Points ~ Broad.Jump + Shuttle, data = qb_relevant_transformed)
```

Coefficients:

(Intercept)	Broad.Jump	Shuttle
-5.85609	0.02144	1.98119

```
qb_predictions <- predict(final_qb_model,
                           newdata = data.frame(Broad.Jump = qbs_2020$Broad.Jump,
                                                  Shuttle = qbs_2020$Shuttle))
qb_predictions <- exp(qb_predictions)
names(qb_predictions) <- qbs_2020$Player
sort(qb_predictions, decreasing = TRUE, na.last = TRUE)
```

Jacob Eason	Kelly Bryant	James Morgan	Cole McDonald	Jordan Love
369.7571	317.0147	310.3789	296.7837	278.2963
Justin Herbert	Shea Patterson	Jake Fromm	Steven Montez	Nate Stanley
275.0659	256.2579	234.8156	227.9076	207.4827
Brian Lewerke	Kevin Davidson	Tua Tagovailoa	Jake Luton	Jalen Hurts
197.1073	170.4695	NA	NA	NA
Anthony Gordon	Joe Burrow			
NA	NA			

5.2 Running Backs

```
rbs_2020 <- filter(combine2020, Pos == "RB")
final_rb_model
```

Call:

```
lm(formula = New_Points ~ Broad.Jump + X40YD, data = rb_relevant_transformed)
```

Coefficients:

(Intercept)	Broad.Jump	X40YD
3.95120	0.02457	-0.60625

```
rb_predictions <- predict(final_rb_model,
                           newdata = data.frame(Broad.Jump = rbs_2020$Broad.Jump,
                                                  X40YD = rbs_2020$X40YD))
rb_predictions <- exp(rb_predictions)
names(rb_predictions) <- rbs_2020$Player
sort(rb_predictions, decreasing = TRUE, na.last = TRUE)
```

AJ Dillon	Darrynton Evans	Rico Dowdle
83.41506	77.41232	75.14935
Jonathan Taylor	Jet Anderson	LeVante Bellamy
74.59896	73.81868	73.30165
Brian Herrien	Cam Akers	Raymond Calais
69.85382	69.34223	68.04861
JaMyca! Hasty	James Robinson	D'Andre Swift
67.70269	67.33684	67.25015
Javon Leake	Patrick Taylor	Joshua Kelley
66.92984	66.88675	66.84368
Clyde Edwards-Helaire	Eno Benjamin	Sewo Olonilua
65.68123	65.26322	63.33499
Anthony McFarland	Salvon Ahmed	DeeJay Dallas
60.93503	60.27826	60.25885
Ke'Shawn Vaughn	J.J. Taylor	La'Mical Perine
59.85606	57.73650	57.38753
Tony Jones	Scottie Phillips	Benny LeMay
56.71416	53.94221	45.76779
Zack Moss	J.K. Dobbins	Mike Warren
NA	NA	NA

5.3 Wide Receivers

```
wrs_2020 <- filter(combine2020, Pos == "WR")
final_wr_model
```

Call:

```
lm(formula = New_Points ~ Shuttle, data = wr_relevant_transformed)
```

Coefficients:

```
(Intercept)      Shuttle
    0.6612         0.7581
```

```
wr_predictions <- predict(final_wr_model,
                           newdata = data.frame(Shuttle = wrs_2020$Shuttle))
wr_predictions <- exp(wr_predictions)
names(wr_predictions) <- wrs_2020$Player
sort(wr_predictions, decreasing = TRUE, na.last = TRUE)
```

Isaiah Coulter	Gabriel Davis	Antonio Gandy-Golden
64.31954	62.87315	60.99510
Jerry Jeudy	Cody White	Omar Bayless
60.07722	59.62347	58.72623
Kendrick Rogers	Jalen Reagor	Stephen Guidry
57.84249	56.97205	56.97205
Denzel Mims	Austin Mack	James Proche
55.69089	55.27027	54.43854
Dezmon Patmon	Juwan Johnson	Quez Watkins
53.61932	53.21435	52.81243
K.J. Osborn	Quintez Cephus	Malcolm Perry
52.41355	51.62481	50.84794
Tony Brown	Freddie Swain	Aaron Parker
49.32909	48.95652	47.85560
John Hightower	Devin Duvernay	Michael Pittman
47.13545	46.77945	44.69919
Isaiah Hodgins	Jeff Thomas	Darrell Stewart
44.02654	NA	NA
Laviska Shenault Jr.	Henry Ruggs III	Joe Reed
NA	NA	NA
Donovan Peoples-Jones	Darnell Mooney	Kaliya Lipscomb
NA	NA	NA
CeeDee Lamb	Tyler Johnson	Collin Johnson
NA	NA	NA
Jauan Jennings	Van Jefferson	Justin Jefferson
NA	NA	NA
Trishton Jackson	K.J. Hill	Tee Higgins
NA	NA	NA
KJ Hamler	Antonio Gibson	Aaron Fuller
NA	NA	NA
Chris Finke	Bryan Edwards	Quartney Davis
NA	NA	NA
Tyrie Cleveland	Chase Claypool	Marquez Callaway

NA	NA	NA
Lawrence Cager	Lynn Bowden	Brandon Aiyuk
NA	NA	NA
Ben Victor		
NA		

5.4 Tight Ends

```
tes_2020 <- filter(combine2020, Pos == "TE")
final_te_model
```

Call:

```
lm(formula = New_Points ~ X40YD + Vertical, data = te_relevant_transformed)
```

Coefficients:

(Intercept)	X40YD	Vertical
11.28489	-1.47657	-0.03112

```
te_predictions <- predict(final_te_model,
                           newdata = data.frame(Vertical = tes_2020$Vertical,
                                                  X40YD = tes_2020$X40YD))
te_predictions <- (te_predictions)^3
names(te_predictions) <- tes_2020$Player
sort(te_predictions, decreasing = TRUE, na.last = TRUE)
```

Brycen Hopkins	Devin Asiasi	Harrison Bryant	Hunter Bryant
37.98652	37.64853	35.58980	35.11267
Dalton Keene	Stephen Sullivan	Colby Parkinson	Josiah Deguara
35.03607	34.90865	33.70690	33.09917
Cole Kmet	Charlie Woerner	Charlie Taumoepeau	C.J. O'Grady
32.57165	31.35332	30.82114	30.50390
Adam Trautman	Mitchell Wilcox	Dom Wood-Anderson	Jared Pinkney
30.48065	30.21196	25.17929	NA
Albert Okwuegbunam	Thaddeus Moss	Sean McKeon	Jacob Breeland
NA	NA	NA	NA

6 Wonderlic Scores

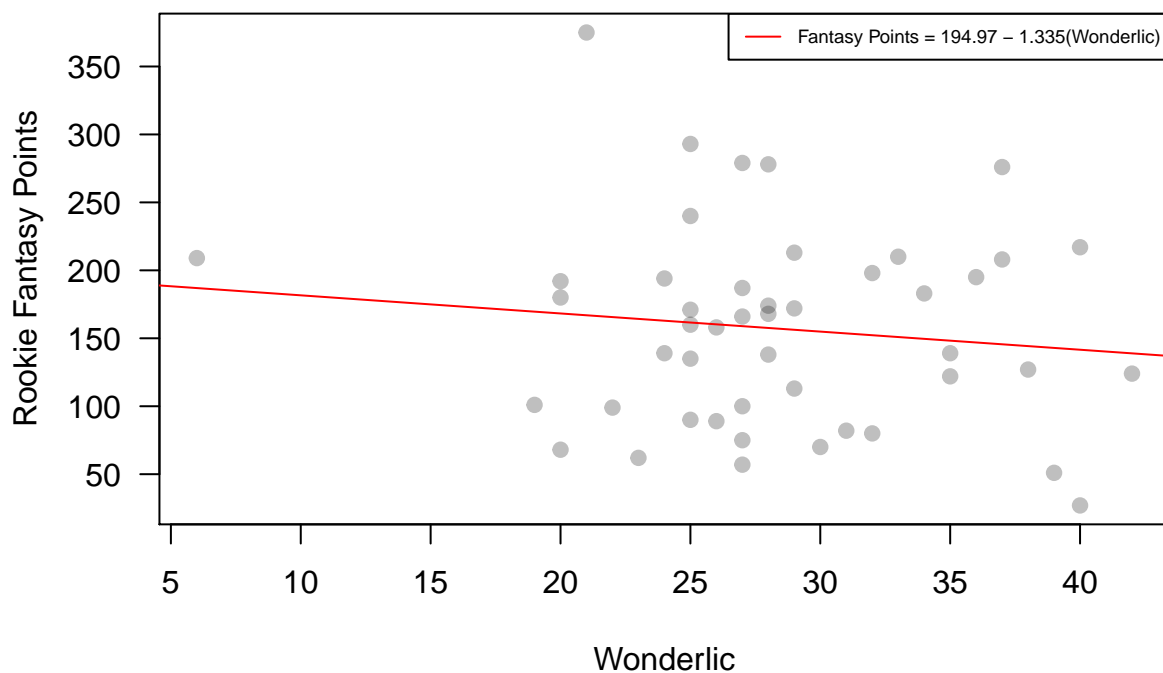
```
wonderlic <- read.csv("Wonderlic Scores - Sheet1.csv", stringsAsFactors = FALSE)
qb_wonder <- wonderlic %>% filter(Position == "QB")
qbs <- qbs[qbs$Player %in% qb_wonder$Player, ]
qb_wonder <- qb_wonder[qb_wonder$Player %in% qbs$Player, ]
qb_wonder <- qb_wonder[-47, ]
qb_wonder_score <- qb_wonder[[2]]
names(qb_wonder_score) <- qb_wonder[[1]]
qb_wonder_score <- qb_wonder_score[order(names(qb_wonder_score))]
qbs <- qbs[order(qbs$Player), ]
```

```

qb_and_wonder_score <- qbs %>% mutate(Wonderlic = qb_wonder_score)
fantasy_wonderlic <- lm(Points ~ Wonderlic, data = qb_and_wonder_score)
plot(Points ~ Wonderlic, data = qb_and_wonder_score,
     col = rgb(0, 0, 0, alpha = 0.25),
     pch = 19,
     las = 1,
     main = "Fantasy Points vs. Wonderlic Scores for QBs",
     ylab = "Rookie Fantasy Points")
abline(fantasy_wonderlic, col = "red")
legend("topright",
      legend = paste("Fantasy Points = ",
                     round(fantasy_wonderlic$coefficients[1], 2),
                     " - ",
                     round(abs(fantasy_wonderlic$coefficients[2]), 3),
                     "(Wonderlic)", sep = ""),
      col = "red", lty = 1,
      cex = 0.6)

```

Fantasy Points vs. Wonderlic Scores for QBs



```
summary(fantasy_wonderlic)
```

Call:

```
lm(formula = Points ~ Wonderlic, data = qb_and_wonder_score)
```

Residuals:

Min	1Q	Median	3Q	Max
-114.570	-67.601	-1.594	39.585	208.066

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	194.967	46.727	4.172	0.000136	***
Wonderlic	-1.335	1.604	-0.832	0.409555	

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

Residual standard error: 73.47 on 45 degrees of freedom

Multiple R-squared: 0.01517, Adjusted R-squared: -0.00672

F-statistic: 0.693 on 1 and 45 DF, p-value: 0.4096