PSTAT 126

Lab 5

Roupen Khanjian

Spring 2021

```r
library(faraway) # Functions and Datasets for Books by Julian Faraway
library(alr4) # Data to Accompany Applied Linear Regression 4th Edition
library(tidyverse) # Easily Install and Load the 'Tidyverse'
library(GGally) # Extension to 'ggplot2'
library(palmerpenguins) # Palmer Archipelago (Antarctica) Penguin Data
library(Lahman) # Sean 'Lahman' Baseball Database
library(patchwork) # The Composer of Plots
```

Contents

Linear Regression Model Assumptions

1) The relationship between each Y_n and each x_n, respectively, is linear. **L**inearity
2) Errors have **E**qual variance. $\text{Var}(Y_n) = \sigma^2$ for every n (homoscedasticity)
3) Errors are **N**ormally distributed
4) Errors are **I**ndependent

- Can use the acronym **L.I.N.E.** to help you remember.

How to test these assumptions?

- Linearity and Constant Variance = Residuals vs. Fitted plot
- Normality = QQ plot

Examples Baseball example

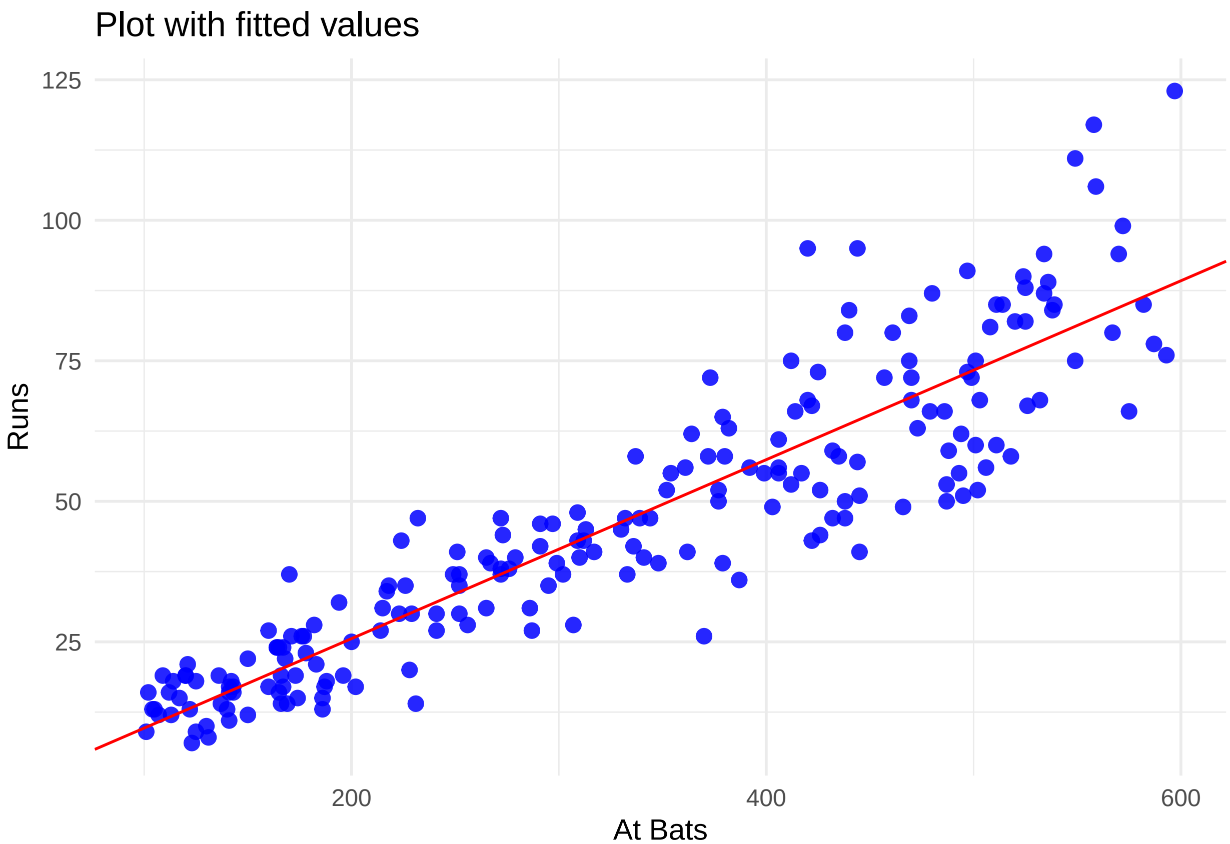
```r
df1 <- Batting %>%
  filter(yearID == "2017" & # stats from 2017
           lgID == "NL" & # only from the NL
           AB > 100 &
           AB < 600 ) # Only At Bats between 100-600.

dim(df1)
```

```
## [1] 207  22
```

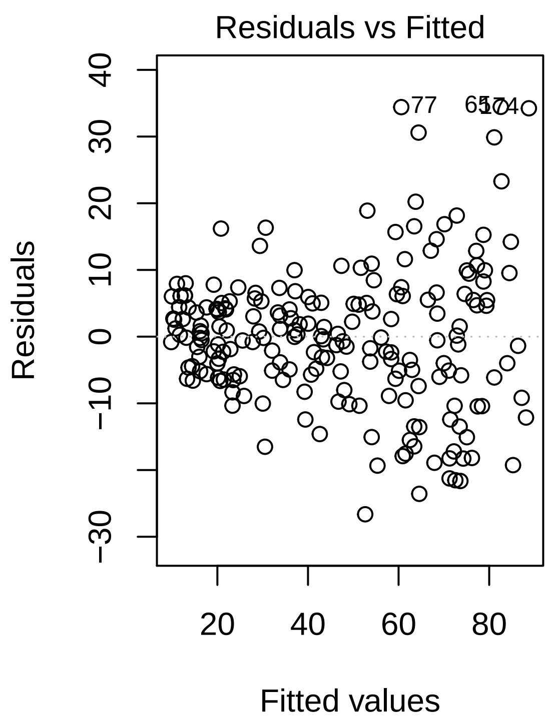
```r
Runs <- df1$R
At_Bats <- df1$AB

model_Runs <- lm(Runs ~ At_Bats)
summary(model_Runs)
```

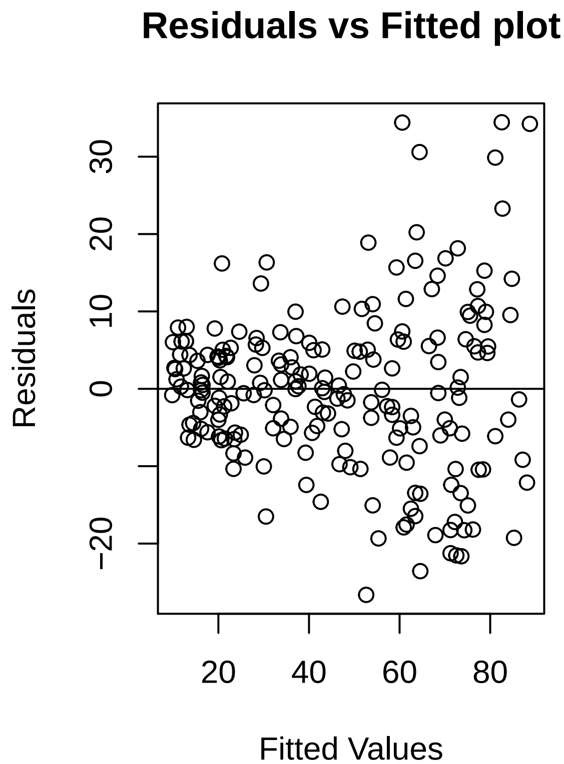
```
##
## Call:
## lm(formula = Runs ~ At_Bats)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -26.630  -5.980  -0.032   5.484  34.448
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.260152   1.812294  -3.454  0.00067 ***
## At_Bats      0.159162   0.005037  31.601  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.53 on 205 degrees of freedom
## Multiple R-squared:  0.8297, Adjusted R-squared:  0.8289
## F-statistic: 998.6 on 1 and 205 DF,  p-value: < 2.2e-16
```

```r
ggplot() +
  geom_point(aes(x = At_Bats, y = Runs), color = "blue",
             alpha = 0.85, size = 2.25) +
  geom_abline(aes(intercept = coef(model_Runs)[1],
                  slope = coef(model_Runs)[2]),
                  color = "red") +
  labs(x = "At Bats",
       y = "Runs",
       title = "Plot with fitted values") +
  theme_minimal()
```

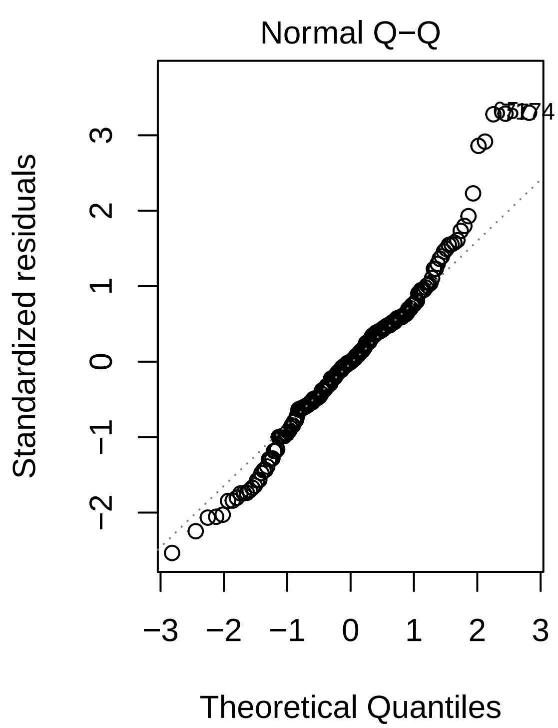


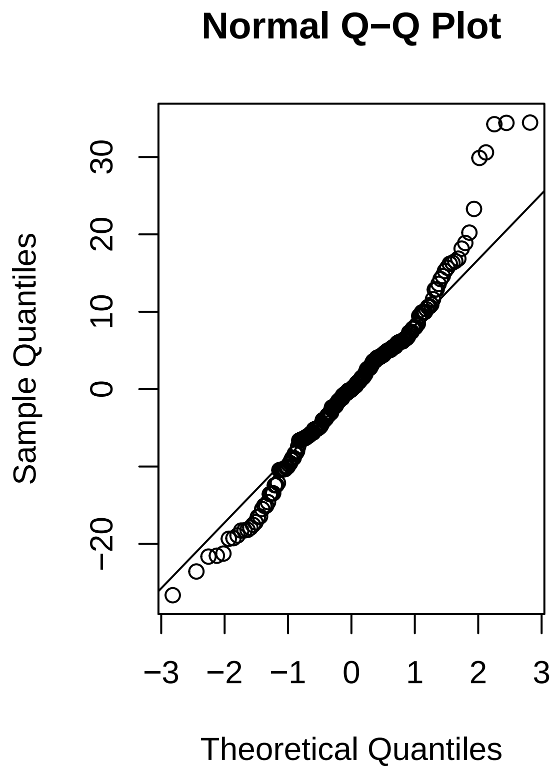

```r
par(mfrow = c(1, 2))

e1 <- resid(model_Runs) # Residuals
y_hat1 <- fitted(model_Runs) # Fitted Values

plot(model_Runs, which = 1, add.smooth = F) # Resid vs. Fit

plot(y_hat1, e1,
     xlab = "Fitted Values",
     ylab = "Residuals",
     main = "Residuals vs Fitted plot")  # Resid vs. Fit
abline(0,0)
```






```r
plot(model_Runs, which = 2) # QQ

qqnorm(e1) # QQ
qqline(e1)
```





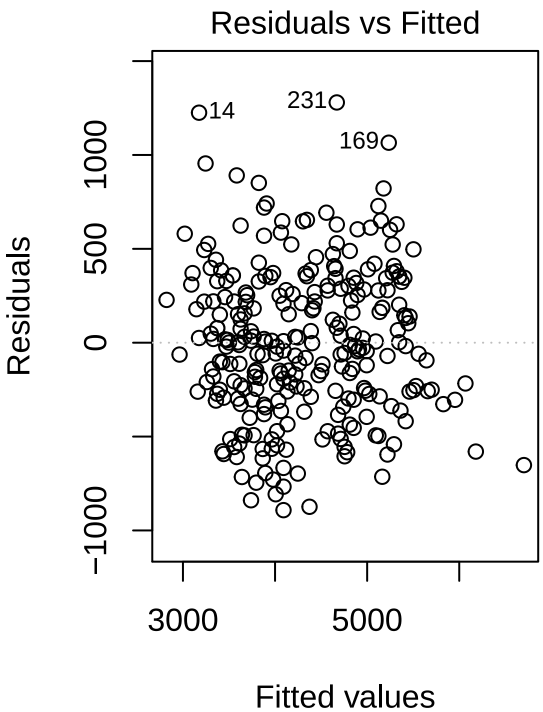
From the QQ-plot we can see that the residuals have a heavy-tailed distribution.

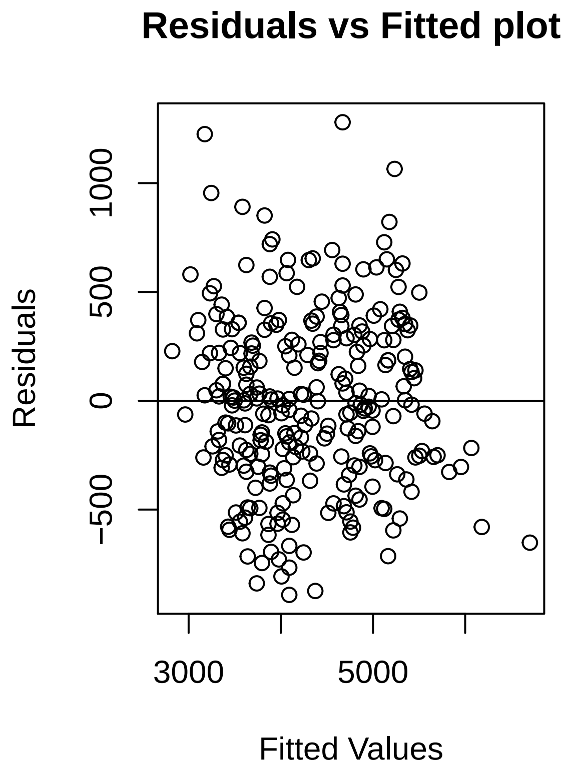
Penguins Example

```r
penguins_noChinstrap <- penguins %>%
  filter(species != "Chinstrap") %>%
  drop_na(bill_length_mm, body_mass_g)

model <- lm(body_mass_g ~ bill_length_mm , data = penguins_noChinstrap)
summary(model)
```

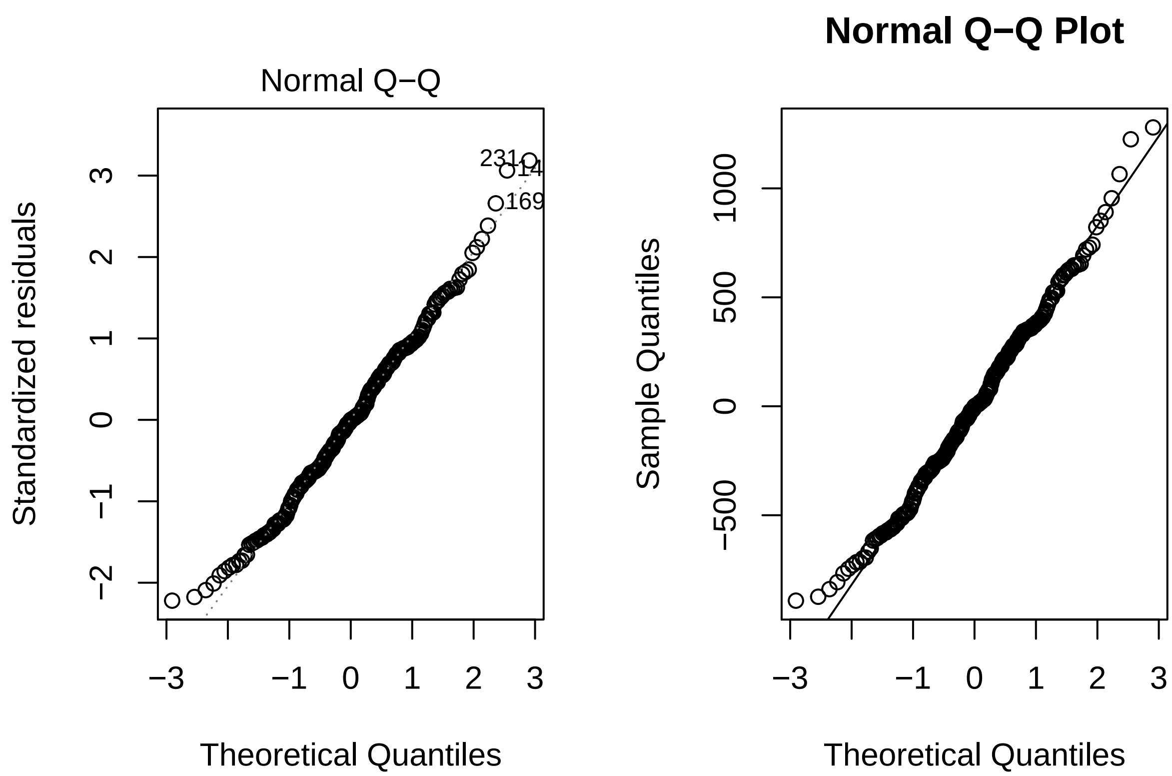
```
##
## Call:
## lm(formula = body_mass_g ~ bill_length_mm, data = penguins_noChinstrap)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -891.91 -272.91   -0.82  282.47 1279.63
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -1706.821    201.712  -8.462 1.65e-15 ***
## bill_length_mm   141.088      4.689  30.088  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 402.5 on 272 degrees of freedom
## Multiple R-squared:  0.769,  Adjusted R-squared:  0.7681
## F-statistic: 905.3 on 1 and 272 DF,  p-value: < 2.2e-16
```

```r
par(mfrow = c(1, 2))

e1 <- resid(model) # Residuals
y_hat1 <- fitted(model) # Fitted Values

plot(model, which = 1,add.smooth = F) # Resid vs. Fit

plot(y_hat1, e1,
     xlab = "Fitted Values",
     ylab = "Residuals",
     main = "Residuals vs Fitted plot")  # Resid vs. Fit
abline(0,0)
```

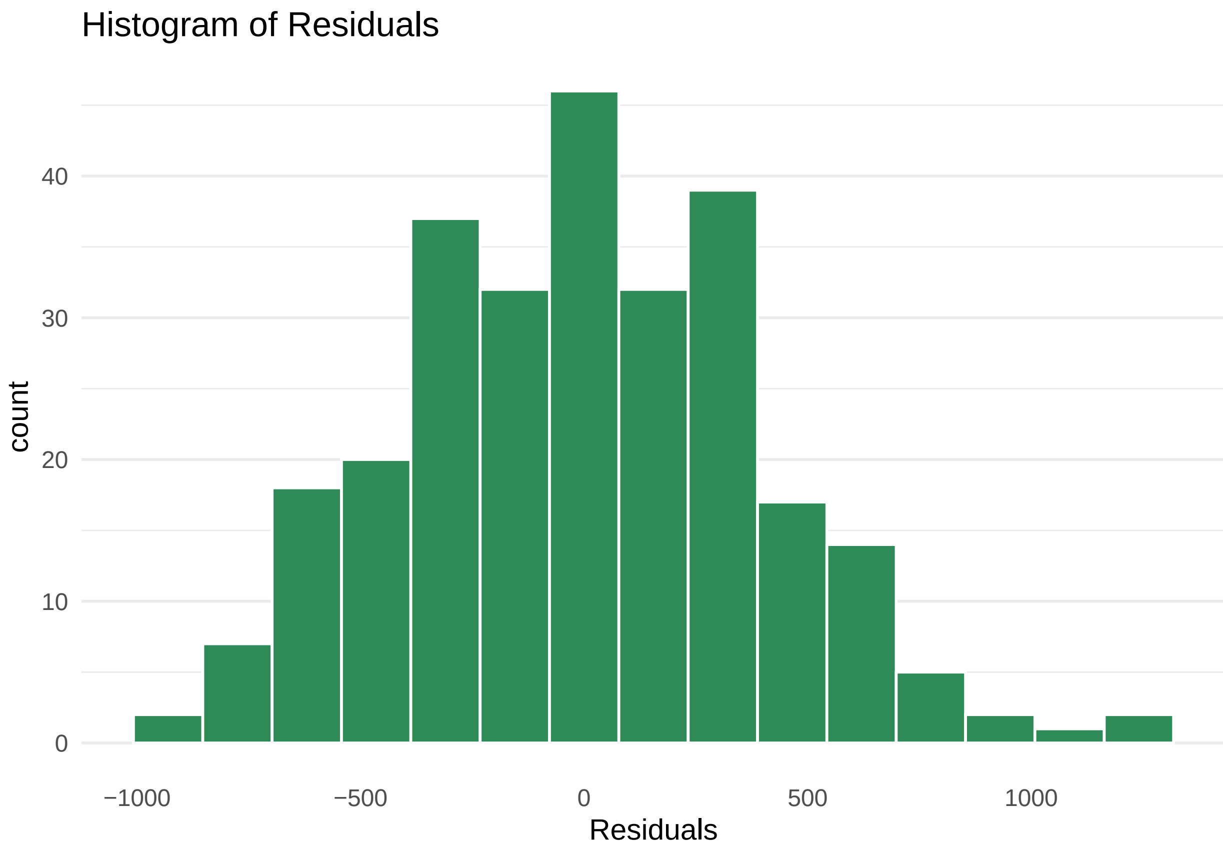





```r
plot(model, which = 2) # QQ

e <- residuals(model) # Residuals

qqnorm(e) # QQ
qqline(e)
```




```r
par(mfrow = c(1, 1))
resid_model <- tibble(residuals = residuals(model))

ggplot(data = resid_model,
       aes(x = residuals)) +
  geom_histogram(color = "white",
                 fill = "seagreen",
                 bins = 15) +
  labs(x = "Residuals",
       title = "Histogram of Residuals") +
  theme_minimal() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank())
```

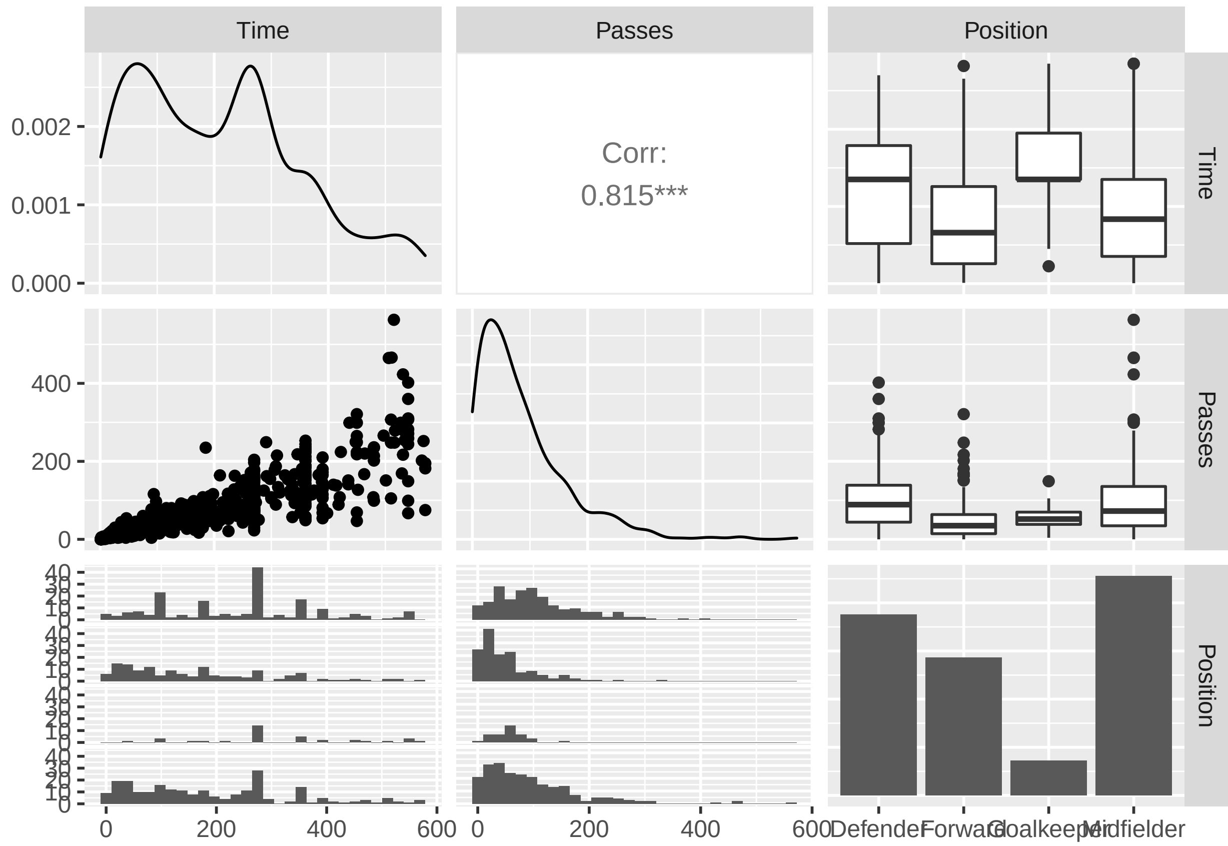


Soccer Example

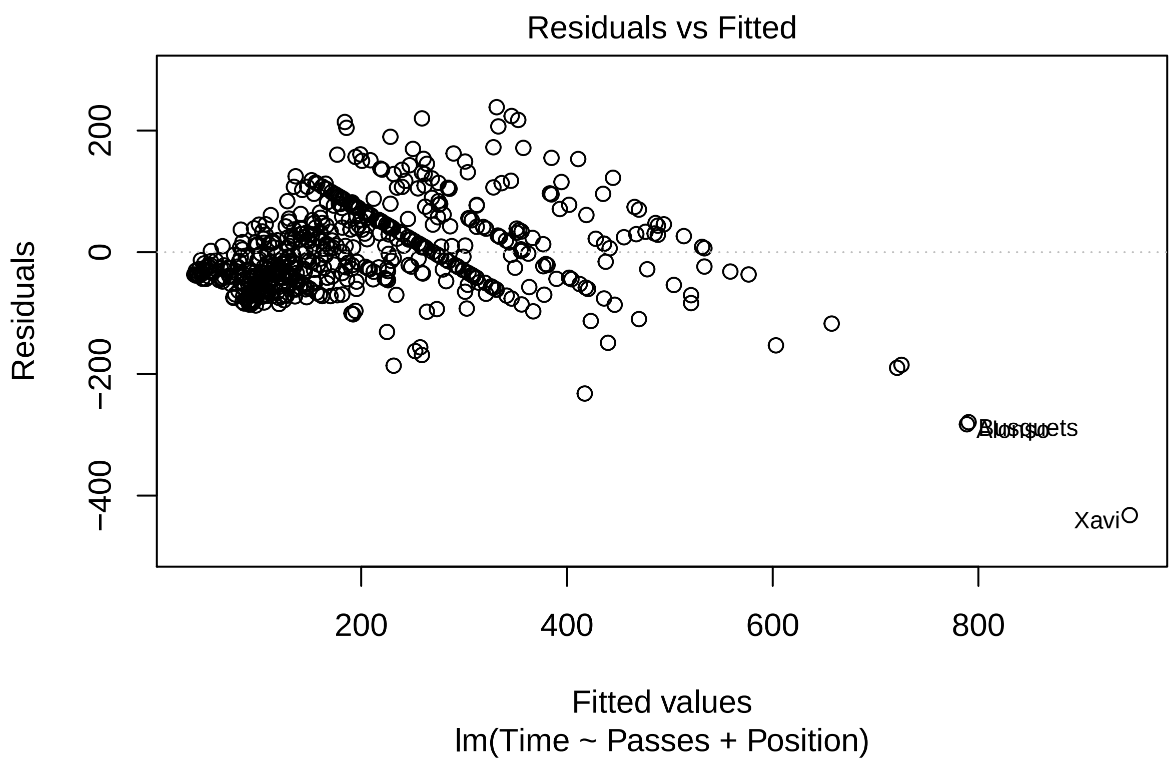
```
head(worldcup)
```

```
##                 Team   Position Time Shots Passes Tackles Saves
## Abdoun       Algeria Midfielder   16     0      6       0     0
## Abe            Japan Midfielder  351     0    101      14     0
## Abidal        France   Defender  180     0     91       6     0
## Abou Diaby    France Midfielder  270     1    111       5     0
## Aboubakar   Cameroon    Forward   46     2     16       0     0
## Abreu        Uruguay    Forward   72     0     15       0     0
```

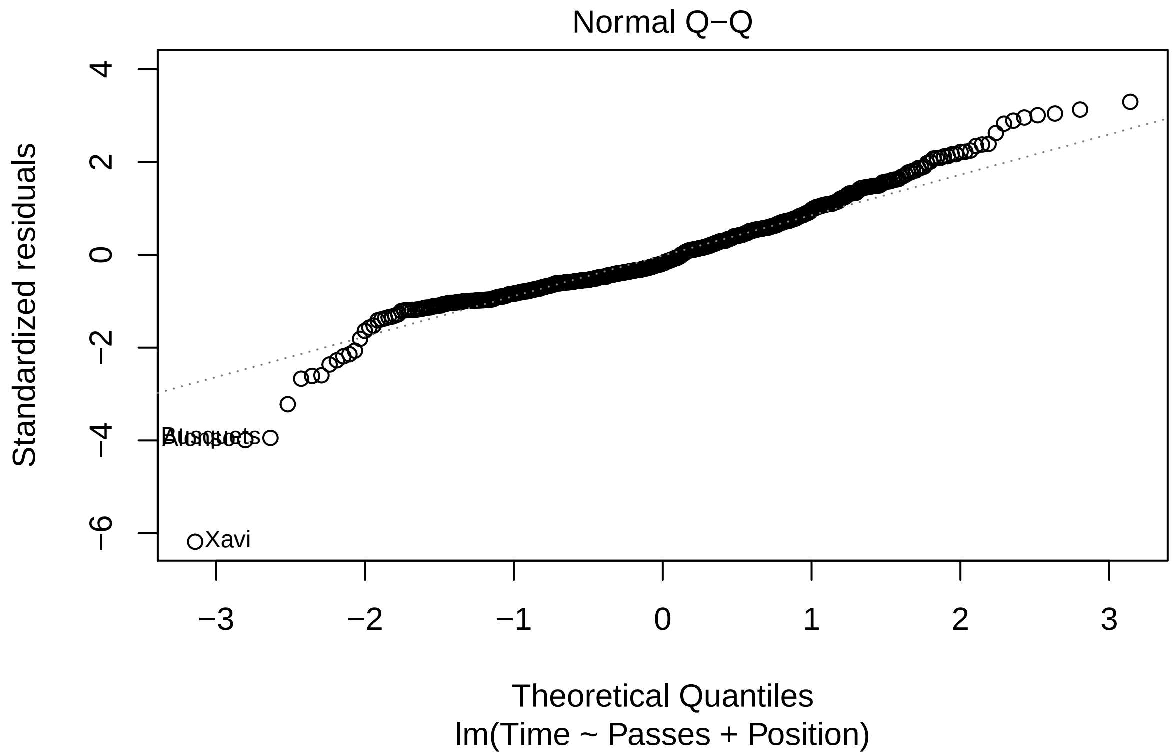
```
ggpairs(worldcup[,c("Time", "Passes", "Position")], lower.panel = NULL)
```




```
model_3 <- lm(Time ~ Passes + Position, worldcup)

plot(model_3, which = 1, add.smooth = F)
```



```r
plot(model_3, which = 2, add.smooth = F)
```

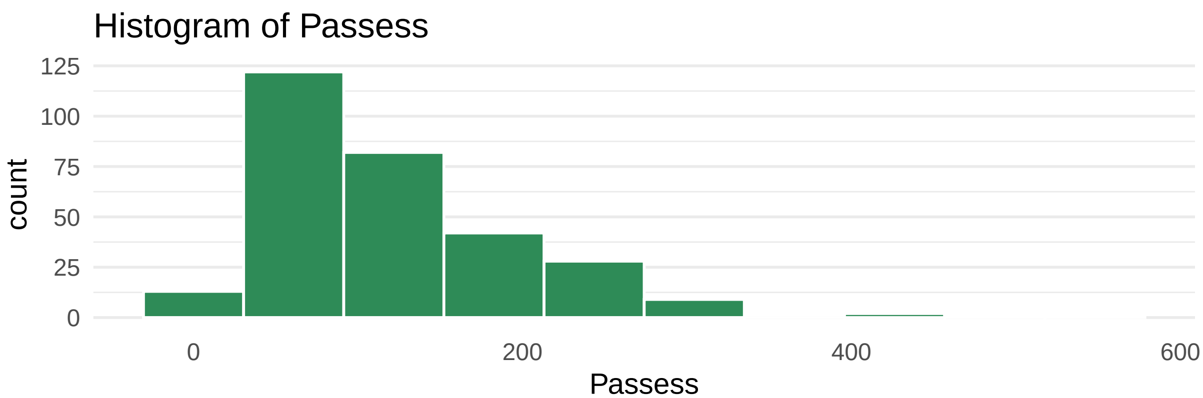


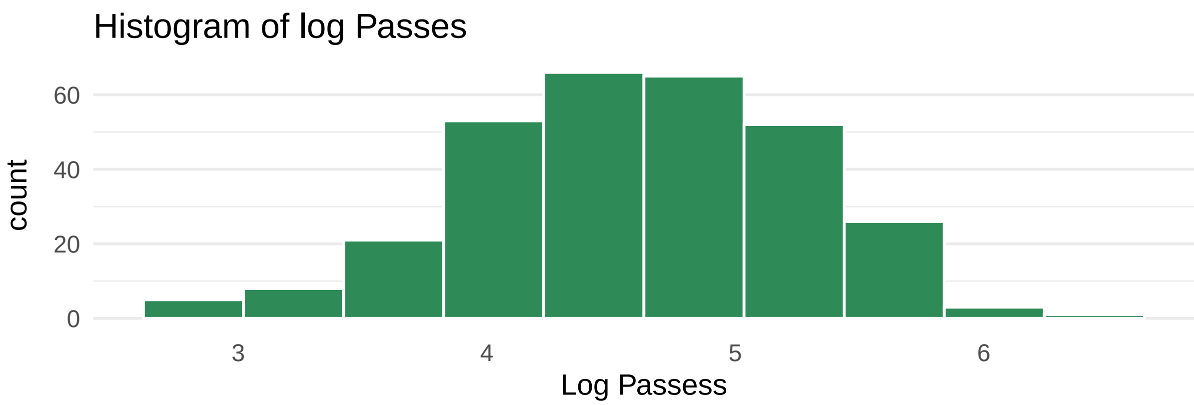
F-Tests

Global F-test

```
worldcup1 <- worldcup %>%
  select(-c(Saves, Team)) %>%
  filter(Passes > 1 ) %>%
  filter(Time > 90) %>%
  filter(Position != "Goalkeeper") %>%
  filter(Tackles > 0) %>%
  filter(Shots > 0)
```

```
g1 <- ggplot(data = worldcup1,
       aes(x = Passes)) +
  geom_histogram(color = "white",
          fill = "seagreen",
          bins = 10) +
  labs(x = "Passess",
       title = "Histogram of Passess") +
  theme_minimal() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank())

g2 <- ggplot(data = worldcup1,
       aes(x = log(Passes))) +
  geom_histogram(color = "white",
          fill = "seagreen",
          bins = 10) +
  labs(x = "Log Passess",
       title = "Histogram of log Passes") +
  theme_minimal() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank())

g1 / g2
```






```r
model_Full <- lm(log(Passes) ~ Time  + Shots + Tackles, data = worldcup1)

# Global F test
model_null <- lm(log(Passes) ~ 1, data = worldcup1)
anova(model_null, model_Full)
```

```
## Analysis of Variance Table
##
## Model 1: log(Passes) ~ 1
## Model 2: log(Passes) ~ Time + Shots + Tackles
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    299 125.29
## 2    296  41.07  3    84.219 202.33 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(model_Full)
```

```
##
## Call:
## lm(formula = log(Passes) ~ Time + Shots + Tackles, data = worldcup1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.20741 -0.22219  0.00296  0.23401  1.15294
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.4004649  0.0557615  60.982  < 2e-16 ***
```

```
## Time         0.0040118  0.0002531  15.852  < 2e-16 ***
## Shots       -0.0149556  0.0063551  -2.353  0.01926 *
## Tackles      0.0208857  0.0056646   3.687  0.00027 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3725 on 296 degrees of freedom
## Multiple R-squared:  0.6722, Adjusted R-squared:  0.6689
## F-statistic: 202.3 on 3 and 296 DF,  p-value: < 2.2e-16
```

Lets define our variables the following way:

$log(\hat{Y}) = log(\text{Passes})$
$x_1 = Time$
$x_2 = Shots$
$x_3 = Tackles$

$$log(\hat{Y}) = 3.4004649 + 0.0040118x_1 - 0.0149556x_2 + 0.0208857x_3$$
$$\therefore \text{If we want to get a value for } \hat{Y}$$
$$\hat{Y} = e^{3.4004649 + 0.0040118x_1 - 0.0149556x_2 + 0.0094249x_3}$$

For example if a player has played 100 minutes, has attempted 3 shots, and has made 10 tackles, then we would expect this player to make about 47.0413546 passes.

```r
exp(3.4004649  + 0.0040118*100 - 0.0149556*3 + 0.0094249*10)
```

```
## [1] 47.04135
```

Partial F-tests

- Testing a Subset of Slope Parameters Equal 0

R = Reduced model F = Full model N = number of observations M = number of predictor variables

$$F^* = \frac{\frac{RSS(R)-RSS(F)}{(N-M-1)_R-(N-M-1)_F}}{\frac{RSS(F)}{(N-M-1)_F}}$$

Model without `Shots` or `Tackles`

```r
model_reduced <- lm(log(Passes) ~ Time, data = worldcup1)
anova(model_reduced, model_Full)
```

```
## Analysis of Variance Table
##
## Model 1: log(Passes) ~ Time
## Model 2: log(Passes) ~ Time + Shots + Tackles
##   Res.Df   RSS Df Sum of Sq     F    Pr(>F)
## 1    298 44.70
## 2    296 41.07  2    3.6298 13.08 3.601e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(model_Full)
```

```
##
## Call:
## lm(formula = log(Passes) ~ Time + Shots + Tackles, data = worldcup1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.20741 -0.22219  0.00296  0.23401  1.15294
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.4004649  0.0557615  60.982  < 2e-16 ***
## Time         0.0040118  0.0002531  15.852  < 2e-16 ***
## Shots       -0.0149556  0.0063551  -2.353  0.01926 *
## Tackles      0.0208857  0.0056646   3.687  0.00027 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3725 on 296 degrees of freedom
## Multiple R-squared:  0.6722, Adjusted R-squared:  0.6689
## F-statistic: 202.3 on 3 and 296 DF,  p-value: < 2.2e-16
```

Model without Shots

```r
model_reduced2 <- lm(log(Passes) ~ Time + Tackles, data = worldcup1)
anova(model_reduced2, model_Full)
```

```
## Analysis of Variance Table
##
## Model 1: log(Passes) ~ Time + Tackles
## Model 2: log(Passes) ~ Time + Shots + Tackles
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1    297 41.838
## 2    296 41.070  1   0.76842 5.5382 0.01926 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(model_Full)
```

```
##
## Call:
## lm(formula = log(Passes) ~ Time + Shots + Tackles, data = worldcup1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.20741 -0.22219  0.00296  0.23401  1.15294
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.4004649  0.0557615  60.982  < 2e-16 ***
## Time         0.0040118  0.0002531  15.852  < 2e-16 ***
## Shots       -0.0149556  0.0063551  -2.353  0.01926 *
## Tackles      0.0208857  0.0056646   3.687  0.00027 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3725 on 296 degrees of freedom
## Multiple R-squared:  0.6722, Adjusted R-squared:  0.6689
## F-statistic: 202.3 on 3 and 296 DF,  p-value: < 2.2e-16
```

Adjusted R^2

```r
statedata <- data.frame(state.x77, row.names = state.abb)
head(statedata)
```

```
##    Population Income Illiteracy Life.Exp Murder HS.Grad Frost   Area
## AL       3615   3624        2.1    69.05   15.1    41.3    20  50708
## AK        365   6315        1.5    69.31   11.3    66.7   152 566432
## AZ       2212   4530        1.8    70.55    7.8    58.1    15 113417
## AR       2110   3378        1.9    70.66   10.1    39.9    65  51945
## CA      21198   5114        1.1    71.71   10.3    62.6    20 156361
## CO       2541   4884        0.7    72.06    6.8    63.9   166 103766
```

```r
lmod <- lm(Life.Exp ~ ., statedata)
summary(lmod)
```

```
##
## Call:
## lm(formula = Life.Exp ~ ., data = statedata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.48895 -0.51232 -0.02747  0.57002  1.49447
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.094e+01  1.748e+00  40.586  < 2e-16 ***
## Population   5.180e-05  2.919e-05   1.775   0.0832 .
## Income      -2.180e-05  2.444e-04  -0.089   0.9293
## Illiteracy   3.382e-02  3.663e-01   0.092   0.9269
## Murder      -3.011e-01  4.662e-02  -6.459 8.68e-08 ***
## HS.Grad      4.893e-02  2.332e-02   2.098   0.0420 *
## Frost       -5.735e-03  3.143e-03  -1.825   0.0752 .
## Area        -7.383e-08  1.668e-06  -0.044   0.9649
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7448 on 42 degrees of freedom
## Multiple R-squared:  0.7362, Adjusted R-squared:  0.6922
## F-statistic: 16.74 on 7 and 42 DF,  p-value: 2.534e-10
```

```r
library(leaps) # Regression Subset Selection
b <- regsubsets(formula(lmod),
                data=statedata)
rs <- summary(b)
rs$which # for each model of size p+1, chooses the model with the lowest RSS value.
```

```
##   (Intercept) Population Income Illiteracy Murder HS.Grad Frost  Area
## 1        TRUE      FALSE  FALSE      FALSE   TRUE   FALSE FALSE FALSE
## 2        TRUE      FALSE  FALSE      FALSE   TRUE    TRUE FALSE FALSE
## 3        TRUE      FALSE  FALSE      FALSE   TRUE    TRUE  TRUE FALSE
## 4        TRUE       TRUE  FALSE      FALSE   TRUE    TRUE  TRUE FALSE
## 5        TRUE       TRUE   TRUE      FALSE   TRUE    TRUE  TRUE FALSE
## 6        TRUE       TRUE   TRUE       TRUE   TRUE    TRUE  TRUE FALSE
## 7        TRUE       TRUE   TRUE       TRUE   TRUE    TRUE  TRUE  TRUE
```

```r
p1 <- ggplot(data = data.frame(rs$rsq), aes(x = 2:8, y =rs$rsq)) +
  geom_point(colour = "red", size = 1.5) +
  geom_label(aes(label= round(rs$rsq, 4)), size = 3, nudge_y = 0.01 ) +
  scale_x_continuous(breaks = seq(2,8,1)) +
  labs(x = "Number of parameters", y = "R^2",
       title = "R^2") +
  theme_minimal()

p2 <- ggplot(data = data.frame(rs$adjr2), aes(x = 2:8, y =rs$rsq)) +
  geom_point(colour = "red", size = 1.5) +
  geom_label(aes(label= round(rs$adjr2, 4)), size = 3, nudge_y = 0.01 ) +
  scale_x_continuous(breaks = seq(2,8,1)) +
  labs(x = "Number of parameters", y = "Adjusted R^2",
       title = "Adjusted R^2") +
  theme_minimal()

p1 / p2
```