

# 3080Project

Leon Weingartner

5/1/2021

```
# Correlation plot against every pair of variables
```

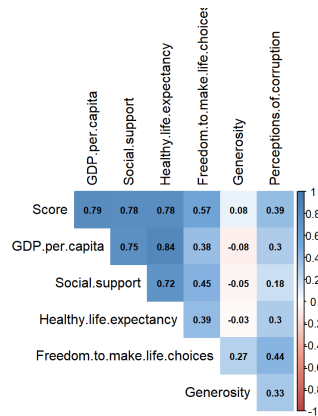
```
setwd("C:/Users/leonw/OneDrive/Desktop/Math 3800 project")  
happinessfile <- "HappinessRegressions.csv"  
happinessData <- read.csv(happinessfile)
```

```
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.0.5
```

```
## corrplot 0.84 loaded
```

```
col <- colorRampPalette(c("#BB4444", "#EE9988", "#FFFFFF", "#77AADD", "#4477AA")) #setup color palette for plot  
X <- cor(subset(happinessData, select = -c(Overall.rank, Country.or.region, Continent))) #create a subset of all columns in  
the data set except non-quantifiable ones  
corrplot(X, method = "color", col = col(200), type = "upper", number.cex = 0.7, addCoef.col = "black",  
         tl.col = "black", tl.srt = 90, sig.level = 0.01, insig = "blank", diag = FALSE) #Plot correlations
```



```
# distribution of score
```

```
setwd("C:/Users/leonw/OneDrive/Desktop/Math 3800 project")  
happinessfile <- "HappinessRegressions.csv"  
happinessData <- read.csv(happinessfile)
```

```
# plot distribution of happiness scores using histogram
```

```
x <- happinessData$Score  
h <- hist(x, breaks = 10, col = "steelblue", xlab = "Happiness Score", ylab = "Count",  
         main = "Happiness Score with Normal Curve")
```

```
# draw normal curve with same mean and sd.
```

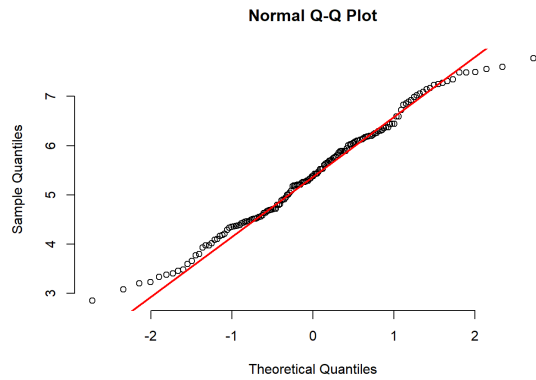
```
xfit <- seq(min(x), max(x), length = 40)  
yfit <- dnorm(xfit, mean = mean(x), sd = sd(x))  
yfit <- yfit * diff(h$mids[1:2]) * length(x)  
lines(xfit, yfit, col = "blue", lwd = 2)
```

Happiness Score with Normal Curve



```
# qq plot
```

```
qqnorm(happinessData$Score, pch = 1, frame = FALSE)  
qqline(happinessData$Score, col = "red", lwd = 2)
```



```
summary(happinessData$Score)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  2.853   4.545   5.380   5.407   6.184   7.769
```

```
setwd("C:/Users/leonw/OneDrive/Desktop/Math 3800 project")
happinessfile <- "HappinessRegressions.csv"
happinessData <- read.csv(happinessfile)
```

```
# freedom to make life choices
```

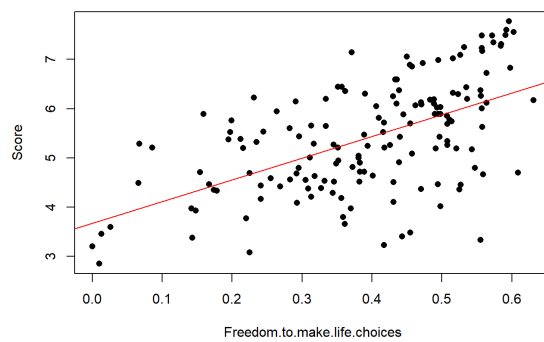
```
# simple regression line
summary(happinessData$Freedom.to.make.life.choices)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.3880  0.4170  0.3926  0.5072  0.6310
```

```
summary(lm(Score ~ Freedom.to.make.life.choices, happinessData))
```

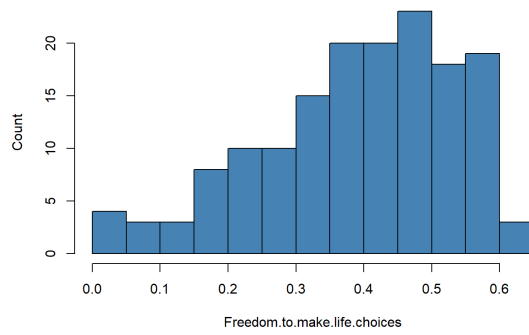
```
##
## Call:
## lm(formula = Score ~ Freedom.to.make.life.choices, data = happinessData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.7882 -0.5838  0.0149  0.7029  1.8269
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.6788     0.2155  17.075 < 2e-16 ***
## Freedom.to.make.life.choices  4.4026     0.5158   8.536 1.24e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9201 on 154 degrees of freedom
## Multiple R-squared:  0.3212, Adjusted R-squared:  0.3168
## F-statistic: 72.87 on 1 and 154 DF,  p-value: 1.238e-14
```

```
plot(Score ~ Freedom.to.make.life.choices, happinessData, pch = 16)
abline(lm(Score ~ Freedom.to.make.life.choices, happinessData), col = "red")
```



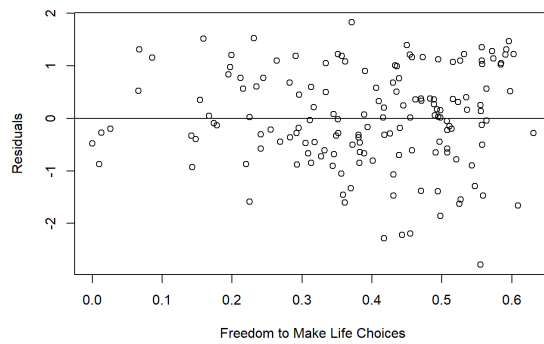
```
# distribution of Freedom.to.make.life.choices
x <- happinessData$Freedom.to.make.life.choices
h <- hist(x, breaks = 10, col = "steelblue", xlab = "Freedom.to.make.life.choices",
  ylab = "Count", main = "Freedom to Make Life Choices Distribution")
```

Freedom to Make Life Choices Distribution



```
# Plot residuals
score.lm = lm(Score ~ Freedom.to.make.life.choices, happinessData)
score.res = resid(score.lm)
plot(happinessData$Freedom.to.make.life.choices, score.res, ylab = "Residuals", xlab = "Freedom to Make Life Choices",
     main = "Happiness-Freedom to Make Life Choices Residuals")
abline(0, 0)
```

Happiness~Freedom to Make Life Choices Residuals



```
# Healthy Life Expectancy
```

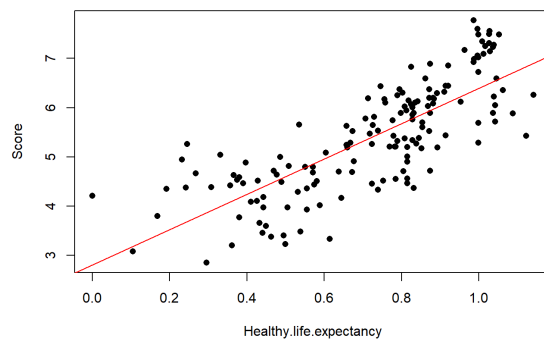
```
# simple regression line
summary(happinessData$Healthy.life.expectancy)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.0000  0.5477  0.7890  0.7252  0.8818  1.1410
```

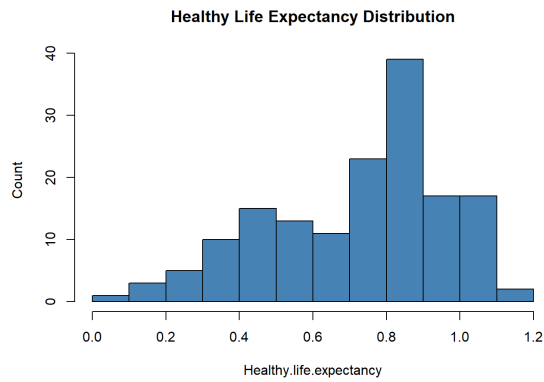
```
summary(lm(Score ~ Healthy.life.expectancy, happinessData))
```

```
##
## Call:
## lm(formula = Score ~ Healthy.life.expectancy, data = happinessData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.6743  -0.4621   0.0863   0.4864   1.5797
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.8068     0.1772   15.84 <2e-16 ***
## Healthy.life.expectancy 3.5854     0.2319   15.46 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.699 on 154 degrees of freedom
## Multiple R-squared:  0.6082, Adjusted R-squared:  0.6057
## F-statistic: 239.1 on 1 and 154 DF, p-value: < 2.2e-16
```

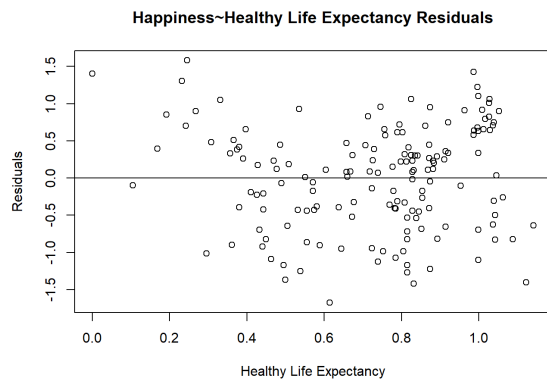
```
plot(Score ~ Healthy.life.expectancy, happinessData, pch = 16)
abline(lm(Score ~ Healthy.life.expectancy, happinessData), col = "red")
```



```
# distribution of Healthy.life.expectancy
x <- happinessData$Healthy.life.expectancy
h <- hist(x, breaks = 10, col = "steelblue", xlab = "Healthy.life.expectancy", ylab = "Count",
  main = "Healthy Life Expectancy Distribution")
```



```
# Plot residuals
score.lm = lm(Score ~ Healthy.life.expectancy, happinessData)
score.res = resid(score.lm)
plot(happinessData$Healthy.life.expectancy, score.res, ylab = "Residuals", xlab = "Healthy Life Expectancy",
  main = "Happiness-Healthy Life Expectancy Residuals")
abline(0, 0)
```



```
# Social Support

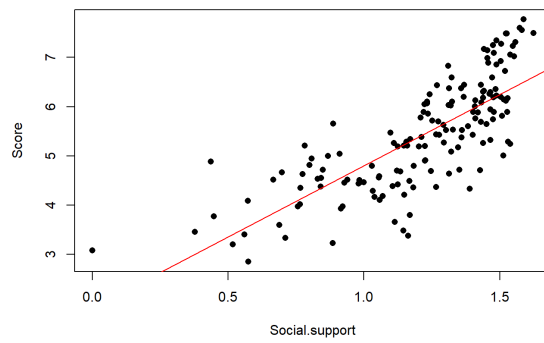
# simple regression line
summary(happinessData$Social.support)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.000   1.056   1.272   1.209   1.452   1.624
```

```
summary(lm(Score ~ Social.support, happinessData))
```

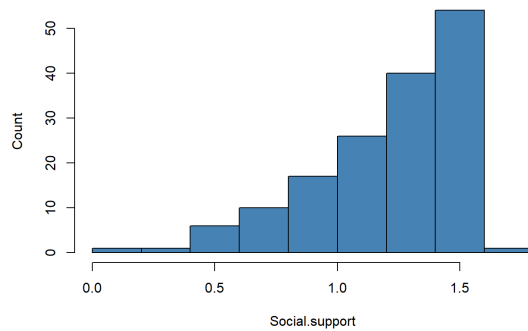
```
##
## Call:
## lm(formula = Score ~ Social.support, data = happinessData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.89465 -0.45762 -0.01993  0.54720  1.70721
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.9124     0.2349   8.14 1.25e-13 ***
## Social.support  2.8910     0.1887  15.32 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7029 on 154 degrees of freedom
## Multiple R-squared:  0.6038, Adjusted R-squared:  0.6012
## F-statistic: 234.7 on 1 and 154 DF, p-value: < 2.2e-16
```

```
plot(Score ~ Social.support, happinessData, pch = 16)
abline(lm(Score ~ Social.support, happinessData), col = "red")
```



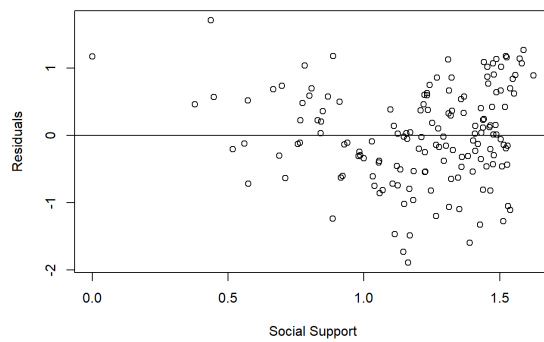
```
# distribution of Social.support
x <- happinessData$Social.support
h <- hist(x, breaks = 10, col = "steelblue", xlab = "Social.support", ylab = "Count",
  main = "Social Support Distribution")
```

**Social Support Distribution**



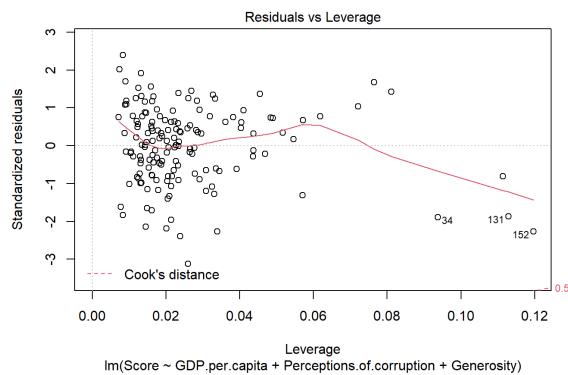
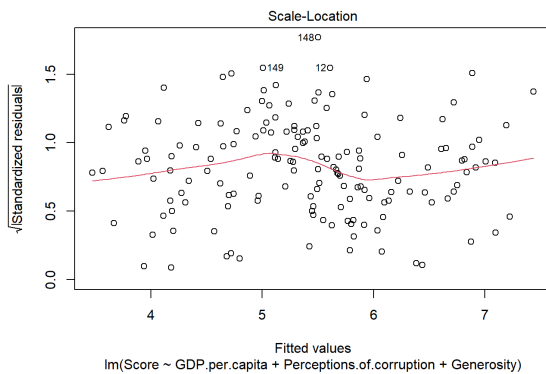
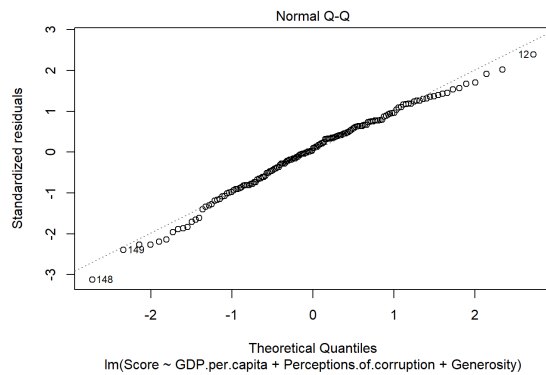
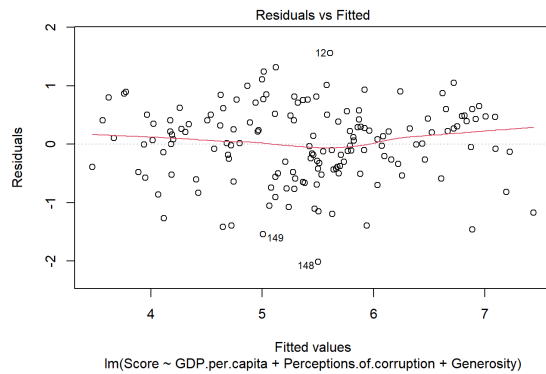
```
# Plot residuals
score.lm = lm(Score ~ Social.support, happinessData)
score.res = resid(score.lm)
plot(happinessData$Social.support, score.res, ylab = "Residuals", xlab = "Social Support",
  main = "Happiness-Social Support Residuals")
abline(0, 0)
```

**Happiness~Social Support Residuals**



```
setwd("C:/Users/leonw/OneDrive/Desktop/Math 3800 project")
happinessFile <- "HappinessRegressions.csv"
happinessData <- read.csv(happinessFile)

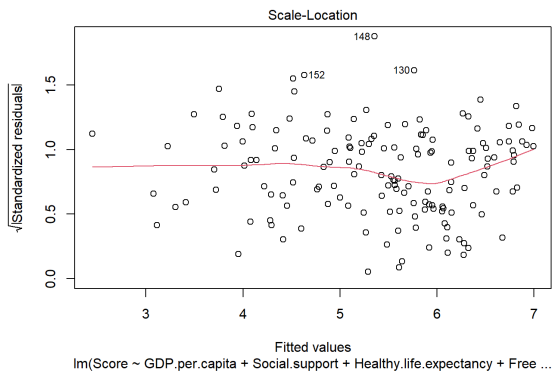
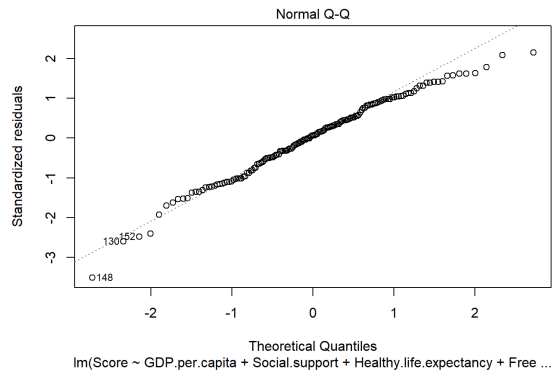
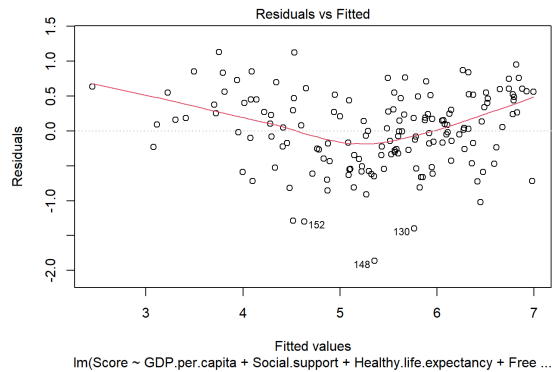
# Experimental model with variables that have low correlation
model1 <- lm(formula = Score ~ GDP.per.capita + Perceptions.of.corruption + Generosity,
  data = happinessData)
plot(model1)
```

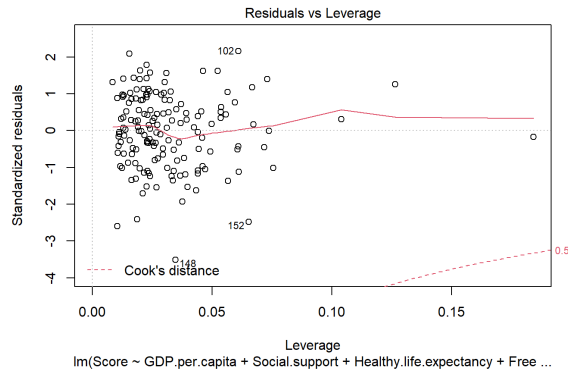


```
summary(model11)
```

```
##
## Call:
## lm(formula = Score ~ GDP.per.capita + Perceptions.of.corruption +
## Generosity, data = happinessData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.01491 -0.42212  0.04152  0.44160  1.55664
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.1050      0.1722   18.031  <2e-16 ***
## GDP.per.capita    2.1341      0.1408   15.160  <2e-16 ***
## Perceptions.of.corruption 1.4850      0.6256    2.374  0.0189 *
## Generosity        1.1158      0.5944    1.877  0.0624 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6533 on 152 degrees of freedom
## Multiple R-squared:  0.6622, Adjusted R-squared:  0.6556
## F-statistic: 99.34 on 3 and 152 DF,  p-value: < 2.2e-16

# summary of final model using 4 significant variables
model2 <- lm(formula = Score ~ GDP.per.capita + Social.support + Healthy.life.expectancy +
Freedom.to.make.life.choices, data = happinessData)
plot(model2)
```





```
summary(model2)
```

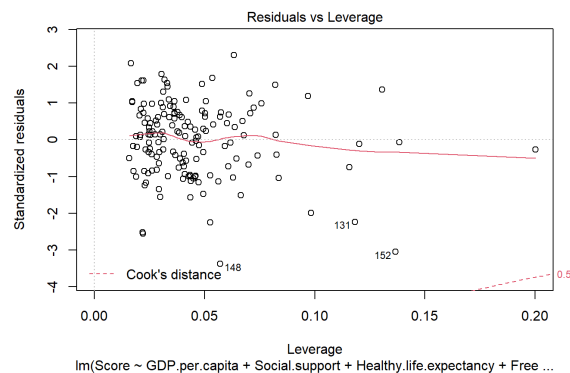
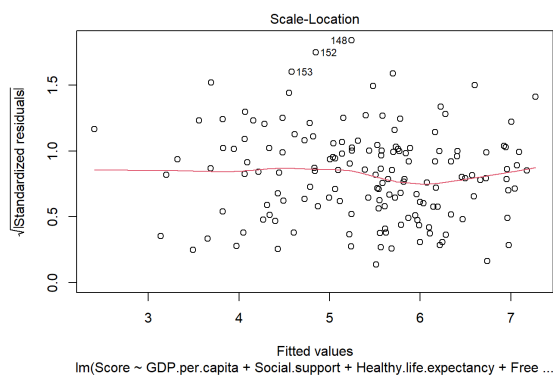
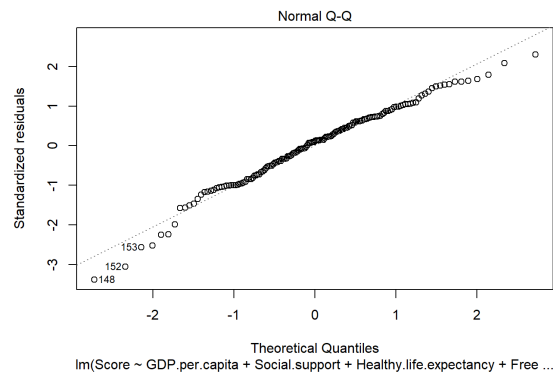
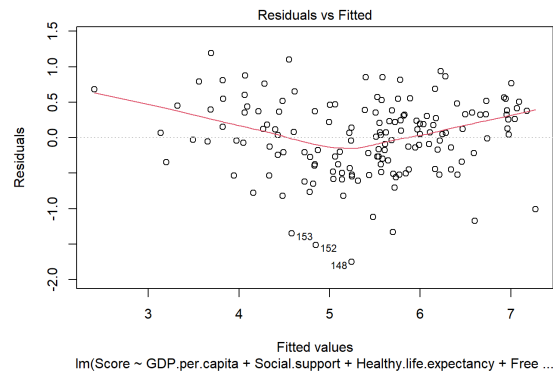
```
##
## Call:
## lm(formula = Score ~ GDP.per.capita + Social.support + Healthy.life.expectancy +
##   Freedom.to.make.life.choices, data = happinessData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.86584 -0.34594  0.03403  0.43676  1.13076
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.8921     0.1994   9.491 < 2e-16 ***
## GDP.per.capita    0.8105     0.2165   3.745 0.000256 ***
## Social.support    1.0166     0.2347   4.331 2.70e-05 ***
## Healthy.life.expectancy 1.1414     0.3373   3.384 0.000910 ***
## Freedom.to.make.life.choices 1.8458     0.3404   5.423 2.28e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5398 on 151 degrees of freedom
## Multiple R-squared:  0.7709, Adjusted R-squared:  0.7649
## F-statistic: 127 on 4 and 151 DF,  p-value: < 2.2e-16
```

```
# data of the country surname from 2015
d <- data.frame(GDP.per.capita = c(0.99534), Social.support = c(0.972), Healthy.life.expectancy = c(0.6082),
  Freedom.to.make.life.choices = c(0.5966))
# predict happiness of surname and find prediction interval
predict(model2, newdata = d, interval = "predict")
```

```
##          fit          lwr          upr
## 1 5.482456 4.389013 6.575898
```

```
# summary of model including all 6 variables
model3 <- lm(formula = Score ~ GDP.per.capita + Social.support + Healthy.life.expectancy +
  Freedom.to.make.life.choices + Generosity + Perceptions.of.corruption, data = happinessData)
plot(model3)
```



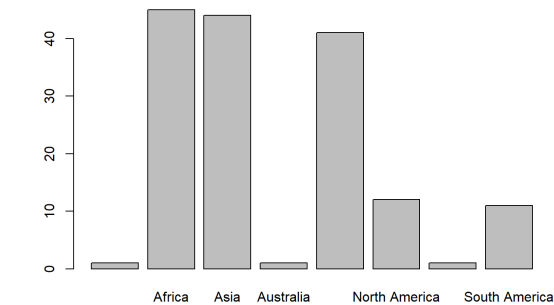


```
summary(model3)
```

```
##
## Call:
## lm(formula = Score ~ GDP.per.capita + Social.support + Healthy.life.expectancy +
##   Freedom.to.make.life.choices + Generosity + Perceptions.of.corruption,
##   data = happinessData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.75384 -0.35386  0.05703  0.36695  1.19059
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.7952      0.2111   8.505 1.77e-14 ***
## GDP.per.capita      0.7754      0.2182   3.553 0.000510 ***
## Social.support      1.1242      0.2369   4.745 4.83e-06 ***
## Healthy.life.expectancy 1.0781      0.3345   3.223 0.001560 **
## Freedom.to.make.life.choices 1.4548      0.3753   3.876 0.000159 ***
## Generosity          0.4898      0.4977   0.984 0.326709
## Perceptions.of.corruption 0.9723      0.5424   1.793 0.075053 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5335 on 149 degrees of freedom
## Multiple R-squared:  0.7792, Adjusted R-squared:  0.7703
## F-statistic: 87.62 on 6 and 149 DF,  p-value: < 2.2e-16

setwd("C:/Users/leonw/OneDrive/Desktop/Math 3800 project")
happinessfile <- "HappinessRegressions.csv"
happinessData <- read.csv(happinessfile)

# visualize count of countries by continent
x <- happinessData$Continent
barplot(table(x))
```



```
# happiness distribution of countries in Europe
Europe <- subset(happinessData, Continent == "Europe")
h <- hist(Europe$Score, xlim = c(2, 9), breaks = seq(2, 9), col = "steelblue", xlab = "Happiness Score",
  ylab = "Count", main = "Happiness Score in Europe")
```



```
print("Europe")

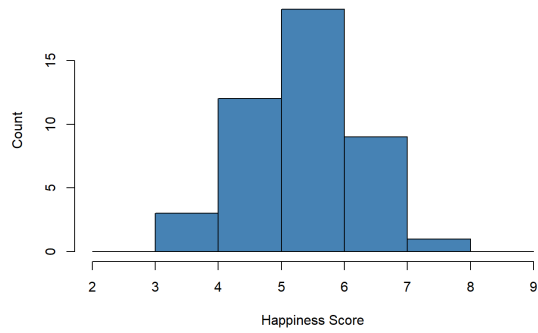
## [1] "Europe"

summary(Europe$Score)

##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  4.332   5.603   6.149   6.268   7.021   7.769

# happiness distribution of countries in Asia
Asia <- subset(happinessData, Continent == "Asia")
h <- hist(Asia$Score, xlim = c(2, 9), breaks = seq(2, 9), col = "steelblue", xlab = "Happiness Score",
  ylab = "Count", main = "Happiness Score in Asia")
```

Happiness Score in Asia



```
print("Asia")
```

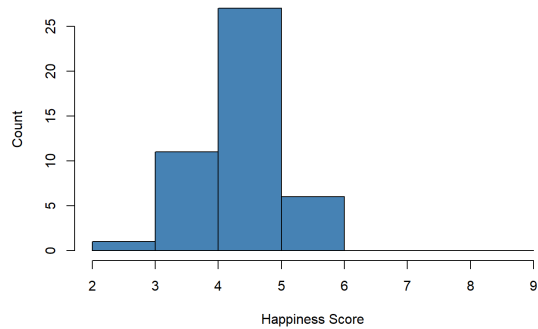
```
## [1] "Asia"
```

```
summary(Asia$Score)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 3.203   4.699   5.254   5.265   5.888   7.139
```

```
# happiness distribution of countries in Africa
Africa <- subset(happinessData, Continent == "Africa")
h <- hist(Africa$Score, xlim = c(2, 9), breaks = seq(2, 9), col = "steelblue", xlab = "Happiness Score",
          ylab = "Count", main = "Happiness Score in Africa")
```

Happiness Score in Africa



```
print("Africa")
```

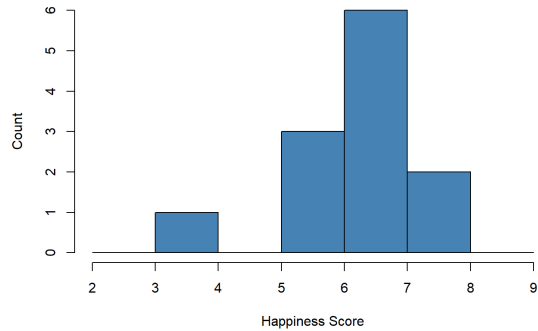
```
## [1] "Africa"
```

```
summary(Africa$Score)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 2.853   3.975   4.461   4.368   4.722   5.888
```

```
# happiness distribution of countries in North America
NAM <- subset(happinessData, Continent == "North America")
h <- hist(NAM$Score, xlim = c(2, 9), breaks = seq(2, 9), col = "steelblue", xlab = "Happiness Score",
          ylab = "Count", main = "Happiness Score in North America")
```

Happiness Score in North America



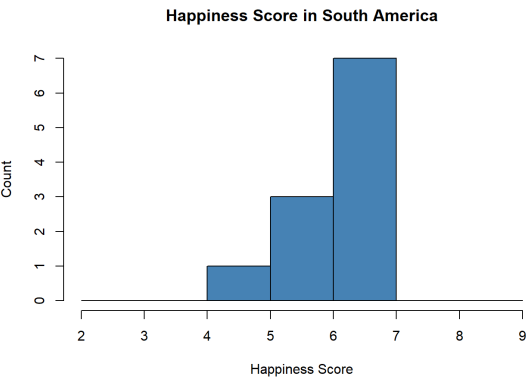
```
print("North America")
```

```
## [1] "North America"
```

```
summary(NAM$Score)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 3.597  5.883   6.287   6.152   6.669   7.278

# happiness distribution of countries in South America
SAm <- subset(happinessData, Continent == "South America")
h <- hist(SAm$Score, xlim = c(2, 9), breaks = seq(2, 9), col = "steelblue", xlab = "Happiness Score",
          ylab = "Count", main = "Happiness Score in South America")
```



```
print("South America")

## [1] "South America"

summary(SAm$Score)

##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 4.707  5.761   6.086   5.945   6.242   6.444
```