Predicting Happiness

An Attempt

Alyssa Andrichik, Eva Licht, and Joe Yalowitz

Abstract

We are investigating the World Happiness Index in order to build a model that can predict a country's Happiness Score based on demographic and geographic factors such as literacy levels, cell-phone use, birth rate, death rate, GDP per capita, perceived corruption, etc. We will build an array of linear models, simple and with interaction, and use other regression analysis tools such as Ridge, Lasso, and PCA to understand our data. We will also use regression trees, random forests, and boosted trees to develop prediction methods for our research question. We have determined by running our training data and validated with our test data, that region is an optimal predictor of happiness.

–

Introduction

Happiness can define a country. Beyond the great importance of individual and community well being which accompanies happiness and positive emotions, studies have shown that positive emotions contribute to "broadening workers' individual mindsets, enabling them to build up their personal resources in terms of enhanced sensitivity and positive attitudes toward their workplace," and can increase productivity (De Satio 2019). Those who identify as "happy" have also been cited to perform more "pro-social" behaviors, such as volunteering (Arafa, 2019). Predicting a country's happiness can aid governments in supporting their citizens, and ensuring greater well being. In terms of analysis, we employed both data model analysis and algorithmic model analysis. Two primary research questions guided our project. First, can we use a mixture of demographic and geographic data to predict Happiness for a country. Second, is there a significant difference in Happiness Score between regions of the world, and is region a significant predictor of Happiness.

–

The Data

Our final dataset pulls data from a wide variety of sources. We obtained our "happiness" data from the Gallup World Poll. Our demographic data on countries came from the US Government. Data on corruption came from Transparency International. We combined these disparate sources into one data set with 120 observations and 22 variables. Each observation refers to a country of the world. These combinations created our complete dataset, but required immense data wrangling. As the sources differed, merging by country resulted in errors because each data set recorded country name differently. We had to mutate the data to eliminate these differences: changing all three data sets to reporting "United States" rather than "The United States" or "United States of America." Additionally, the data was collected by several different organizations, and some data had commas to signify decimals, rather than periods. Variable names had to be normalized and checked for any possible causes of error; for example, GDP Per Capita was originally reported as "GDP ($ per capita)" and the dollar sign prompted errors in the code. After fixing the variable names, merging, and checking the data reporting format, the data was ready for its initial analysis.

Exploratory Data Analysis

```
##
## Call:
## lm(formula = HappinessScore ~ Infantmortalityper1000births +
##     Literacy, data = happy_country)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.91891 -0.58185 -0.02146  0.61982  1.70423
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   4.142e+00  4.580e-01   9.043 3.89e-15 ***
## Infantmortalityper1000births -1.249e-04  2.712e-05  -4.604 1.06e-05 ***
## Literacy                      2.091e-03  4.755e-04   4.397 2.43e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.844 on 117 degrees of freedom
## Multiple R-squared:  0.4896, Adjusted R-squared:  0.4808
## F-statistic: 56.11 on 2 and 117 DF,  p-value: < 2.2e-16
```

```
names(happy_country)
```

```
##  [1] "Country"                       "HappinessScore"
##  [3] "Region"                        "Population"
##  [5] "Areasqmi"                      "PopDensitypersqmi"
##  [7] "Coastlinecoastbyarearatio"     "Netmigration"
##  [9] "Infantmortalityper1000births" "GDP_percapita"
```

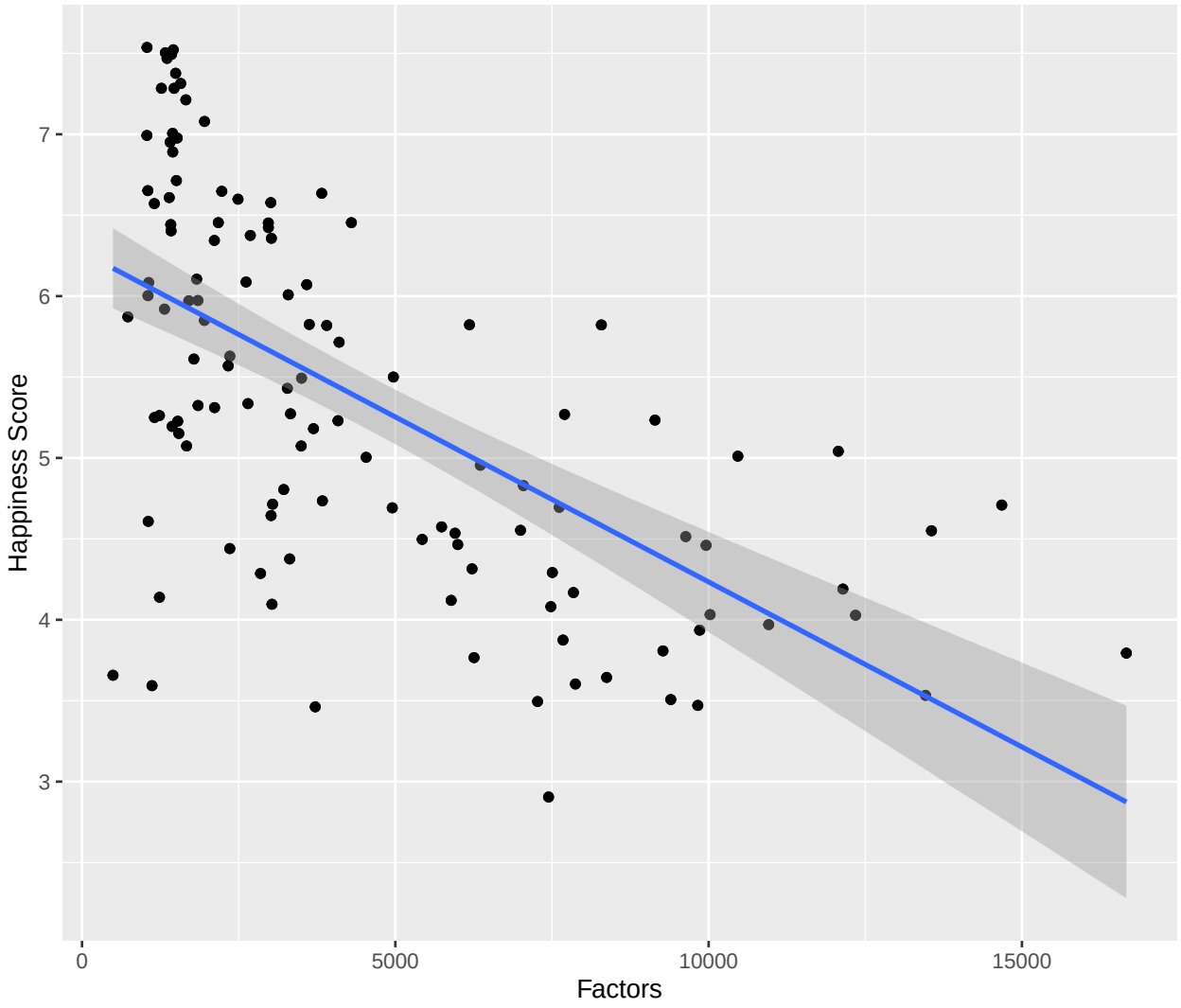


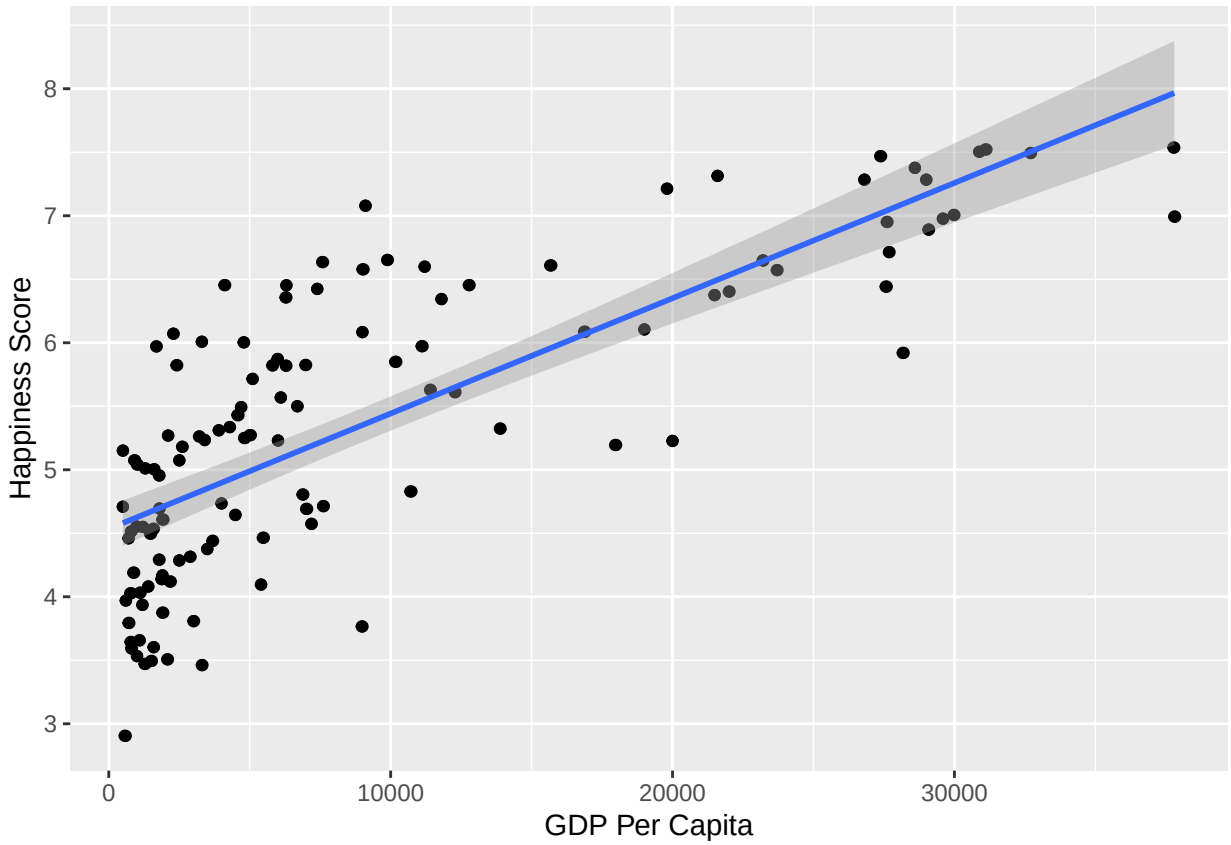
Figure 1: plotting example

```
## [11] "Literacy"                    "Phonesper1000"
## [13] "Arable"                      "Crops"
## [15] "Other"                       "Climate"
## [17] "Birthrate"                   "Deathrate"
## [19] "Agriculture"                 "Industry"
## [21] "Service"                     "PercievedCorruptionScore"
```

```
mod5<- lm(HappinessScore~GDP_percapita, data=happy_country)
summary(mod5)
```

```
##
## Call:
## lm(formula = HappinessScore ~ GDP_percapita, data = happy_country)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.68442 -0.54668 -0.05663  0.46238  1.71796
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.535e+00  9.248e-02   49.04   <2e-16 ***
## GDP_percapita 9.078e-05  6.826e-06   13.30   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7441 on 118 degrees of freedom
## Multiple R-squared:  0.5998, Adjusted R-squared:  0.5964
## F-statistic: 176.9 on 1 and 118 DF,  p-value: < 2.2e-16
```

```
ggplot(happy_country, aes(x=GDP_percapita, y=HappinessScore)) +
  geom_point()+
  labs(x="GDP Per Capita", y="Happiness Score") +
  geom_jitter()+
  stat_smooth(method = "lm")
```

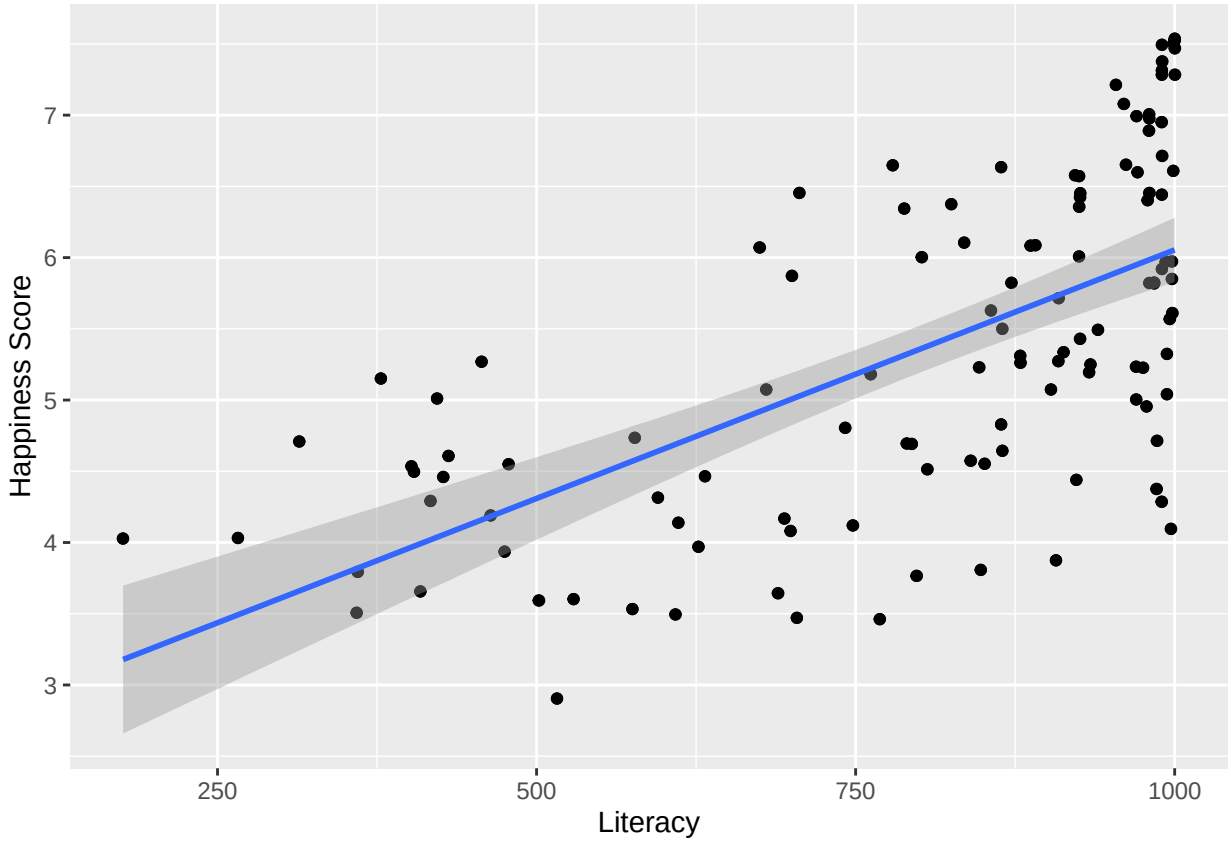



```
lm1<- lm(HappinessScore~Literacy, data= happy_country)
summary(lm1)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Literacy, data = happy_country)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.94783 -0.71092 -0.05079  0.74260  1.48270
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.5644058  0.3289310   7.796 2.80e-12 ***
## Literacy    0.0034899  0.0003959   8.816 1.24e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9133 on 118 degrees of freedom
## Multiple R-squared:  0.3971, Adjusted R-squared:  0.392
## F-statistic: 77.72 on 1 and 118 DF,  p-value: 1.243e-14
```

```
lmplot1<- ggplot(happy_country, aes(x = Literacy, y = HappinessScore)) +
  geom_point() +
  labs(x="Literacy", y="Happiness Score")+
  geom_jitter() +
```

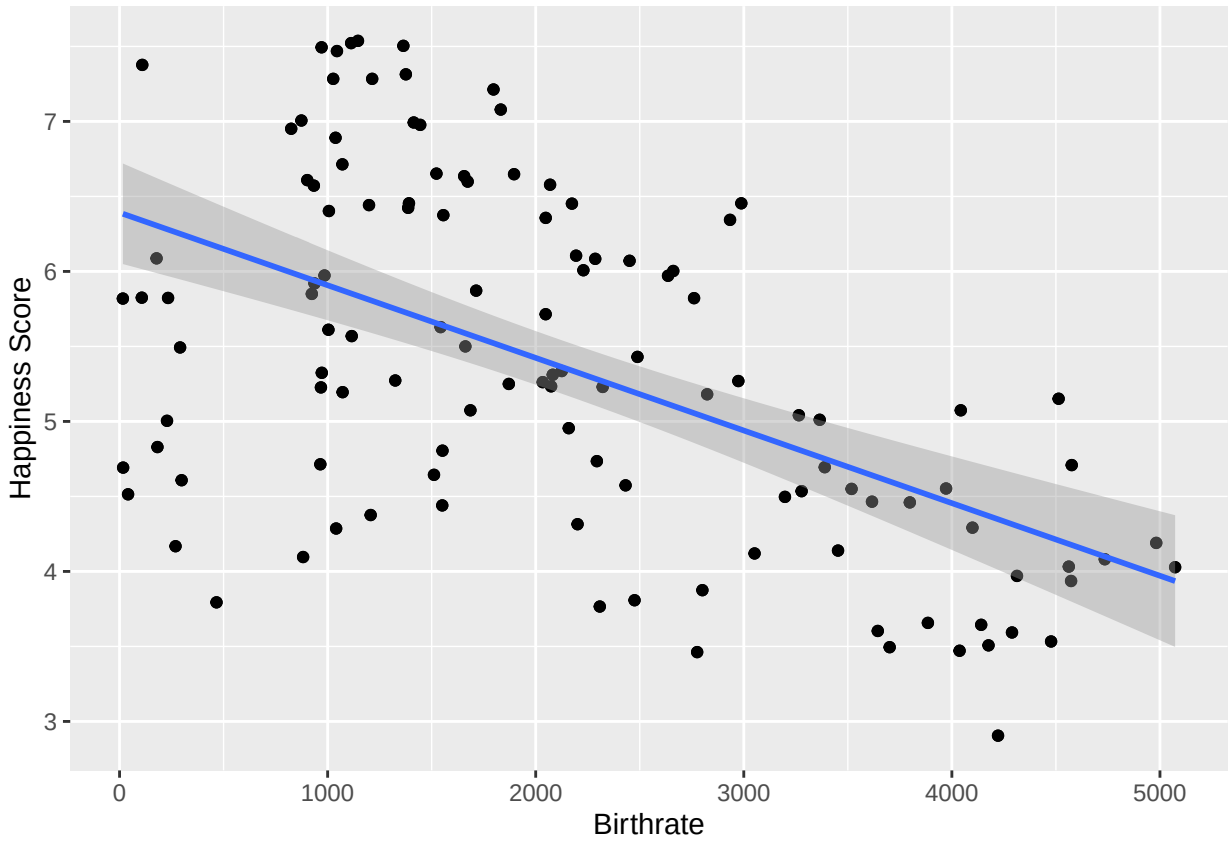
```
  stat_smooth(method = "lm")
lmplot1
```




```
lm2<- lm(HappinessScore~Birthrate, data= happy_country)
summary(lm2)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Birthrate, data = happy_country)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.37237 -0.69253 -0.06449  0.80929  1.77243
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.392e+00  1.702e-01  37.555  < 2e-16 ***
## Birthrate   -4.842e-04  6.842e-05  -7.077 1.14e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9856 on 118 degrees of freedom
## Multiple R-squared:  0.2979, Adjusted R-squared:  0.292
## F-statistic: 50.08 on 1 and 118 DF,  p-value: 1.137e-10
```

```
lmplot2<- ggplot(happy_country, aes(x = Birthrate, y = HappinessScore)) +
  geom_point() +
  labs(x="Birthrate", y="Happiness Score")+
  geom_jitter() +
  stat_smooth(method = "lm")
lmplot2
```

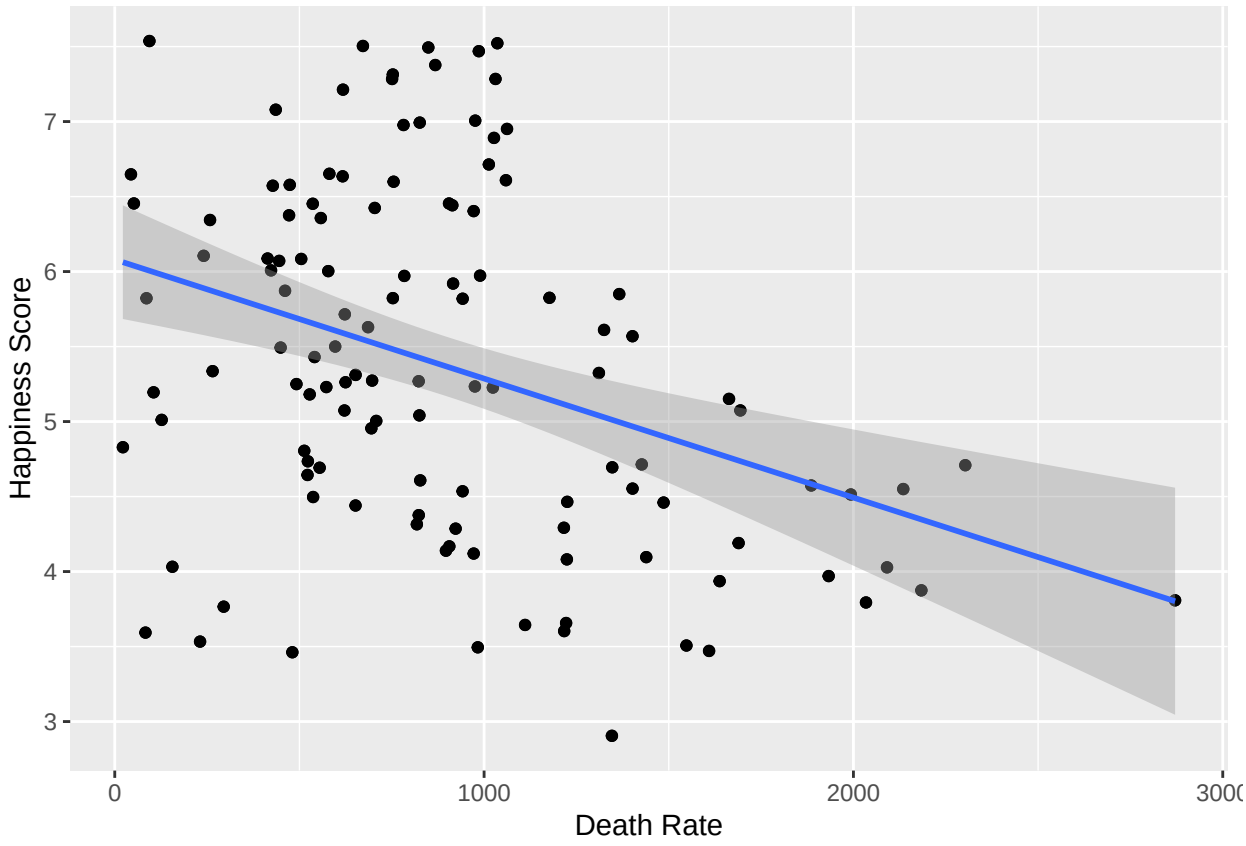



```
lm3<- lm(HappinessScore~Deathrate, data= happy_country)
summary(lm3)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Deathrate, data = happy_country)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.42118 -0.81773 -0.05652  0.68903  2.26394
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.0800321  0.1948483  31.204  < 2e-16 ***
## Deathrate   -0.0007934  0.0001868  -4.248 4.32e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.095 on 118 degrees of freedom
```

```
## Multiple R-squared:  0.1327, Adjusted R-squared:  0.1253
## F-statistic: 18.05 on 1 and 118 DF,  p-value: 4.322e-05
```

```
lmplot3<- ggplot(happy_country, aes(x = Deathrate, y = HappinessScore)) +
  geom_point() +
  labs(x="Death Rate", y="Happiness Score")+
  geom_jitter() +
  stat_smooth(method = "lm")
lmplot3
```



PCA

```
pca1 <- prcomp(num_happy, center=TRUE, scale. =  TRUE)
summary(pca1)
```

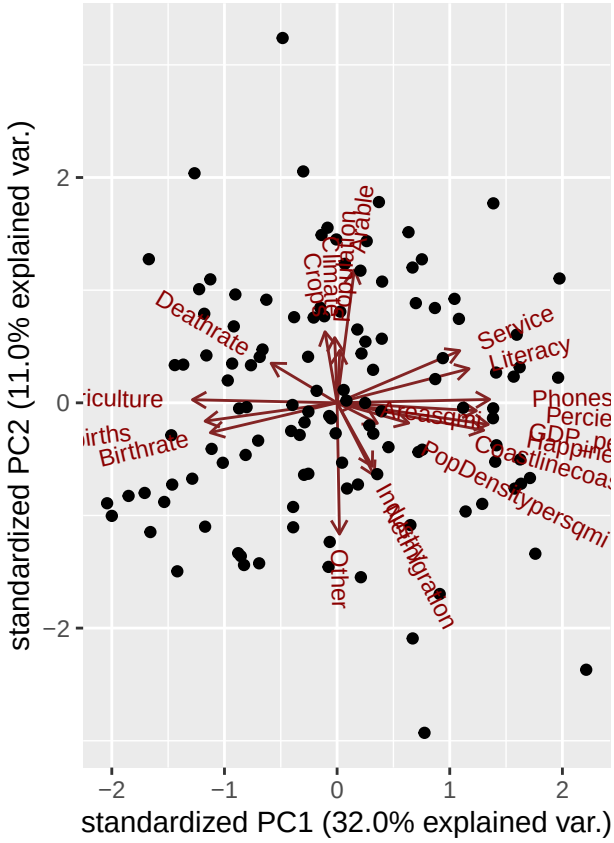
```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5     PC6    PC7
## Standard deviation     2.5289 1.4861 1.4330 1.26159 1.23050 1.08556 0.9402
## Proportion of Variance 0.3198 0.1104 0.1027 0.07958 0.07571 0.05892 0.0442
## Cumulative Proportion  0.3198 0.4302 0.5329 0.61245 0.68816 0.74708 0.7913
##                           PC8     PC9    PC10    PC11    PC12    PC13
## Standard deviation     0.82234 0.78462 0.7510 0.67712 0.62004 0.60078
## Proportion of Variance 0.03381 0.03078 0.0282 0.02292 0.01922 0.01805
```

```
## Cumulative Proportion  0.82509 0.85587 0.8841 0.90700 0.92622 0.94427
##                           PC14    PC15    PC16    PC17    PC18    PC19
## Standard deviation     0.52661 0.48194 0.46201 0.45422 0.36352 0.23045
## Proportion of Variance 0.01387 0.01161 0.01067 0.01032 0.00661 0.00266
## Cumulative Proportion  0.95813 0.96975 0.98042 0.99074 0.99734 1.00000
##                           PC20
## Standard deviation     0.004019
## Proportion of Variance 0.000000
## Cumulative Proportion  1.000000
```

```
str(pca1)
```

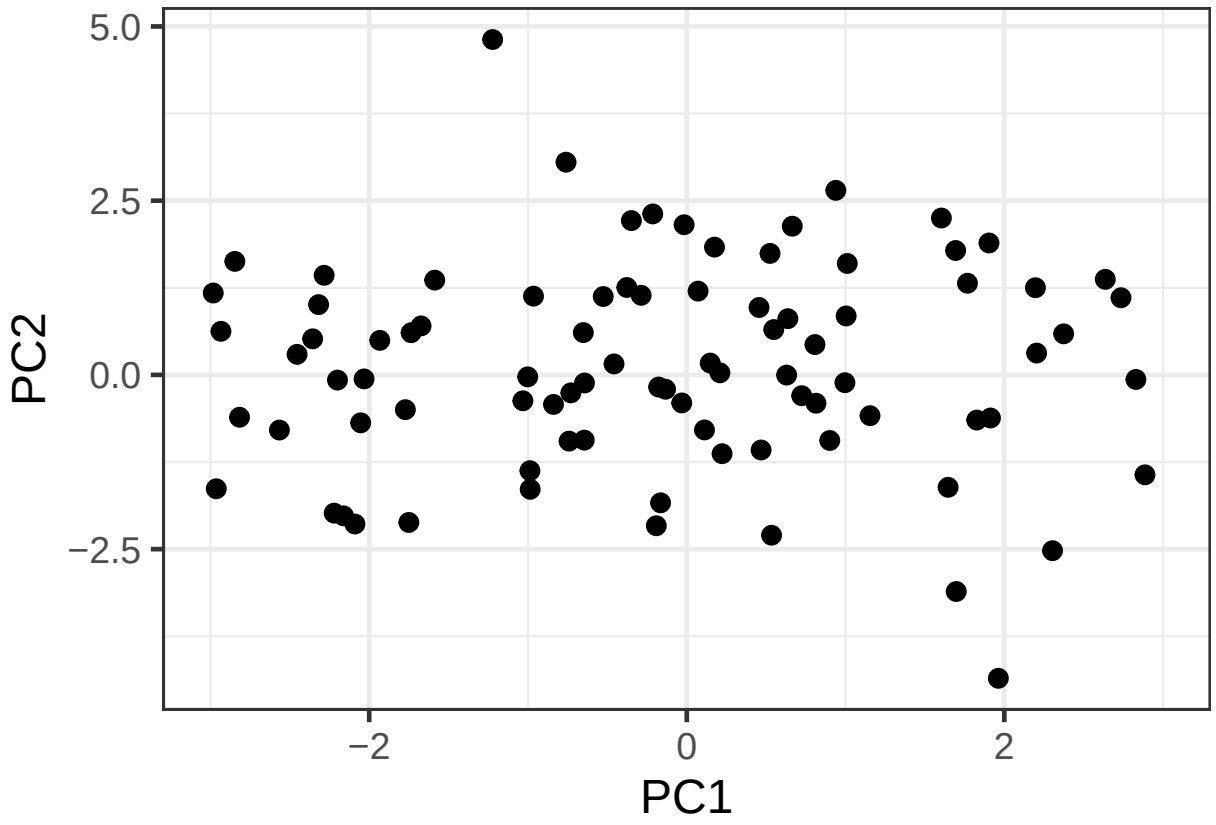
```
## List of 5
##  $ sdev    : num [1:20] 2.53 1.49 1.43 1.26 1.23 ...
##  $ rotation: num [1:20, 1:20] 0.33775 0.00626 0.04316 0.09277 0.16504 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:20] "HappinessScore" "Population" "Areasqmi" "PopDensitypersqmi" ...
##   .. ..$ : chr [1:20] "PC1" "PC2" "PC3" "PC4" ...
##  $ center  : Named num [1:20] 5.37 4.86e+07 7.88e+05 1.64e+02 1.75 ...
##   ..- attr(*, "names")= chr [1:20] "HappinessScore" "Population" "Areasqmi" "PopDensitypersqmi" ...
##  $ scale   : Named num [1:20] 1.17 1.59e+08 1.65e+06 6.03e+02 4.21 ...
##   ..- attr(*, "names")= chr [1:20] "HappinessScore" "Population" "Areasqmi" "PopDensitypersqmi" ...
##  $ x       : num [1:120, 1:20] -3.587 -0.526 -0.192 1.154 -0.651 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:20] "PC1" "PC2" "PC3" "PC4" ...
##  - attr(*, "class")= chr "prcomp"
```

```
ggbiplot(pca1)
```

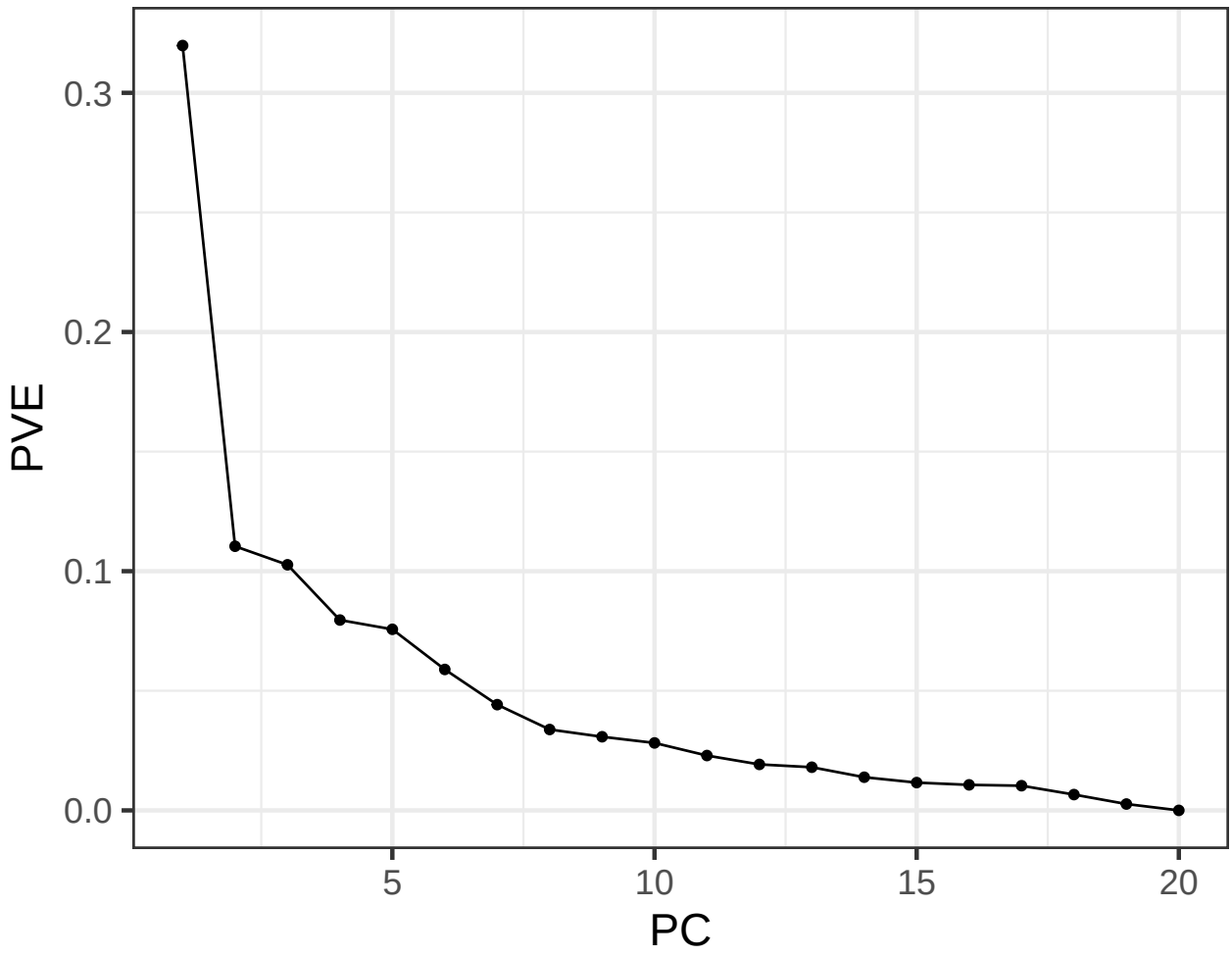



```
dat <- as.data.frame(pca1$x)
PCAPlot <- ggplot(dat, aes(x = PC1, y = PC2)) +
  geom_point(size = 3) +
  xlim(c(-3, 3)) +
  theme_bw(base_size = 18)
PCAPlot
```

```
## Warning: Removed 33 rows containing missing values (geom_point).
```



Scree Plot



Breakdown of Variables and Regions Data

Top Correlated Varaiables: Perceived Corruption (0.8832017), Net Migration (0.8384467), and Industry(0.8170564).

```
#Linear Model to Predict Eastern European Happiness
EastEurolm <- lm(HappinessScore ~ PercievedCorruptionScore + Netmigration +
                 Industry + Coastlinecoastbyarearatio, data = eastern_europe)
summary(EastEurolm)
```

```
##
## Call:
## lm(formula = HappinessScore ~ PercievedCorruptionScore + Netmigration +
##     Industry + Coastlinecoastbyarearatio, data = eastern_europe)
##
## Residuals:
##         1         2         3         4         5         6
## -0.001902 -0.007578 -0.022118 -0.015840  0.007886  0.039551
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)               0.346151   0.301958   1.146   0.4567
## PercievedCorruptionScore  0.042532   0.003829  11.108   0.0572 .
## Netmigration              0.164704   0.015014  10.970   0.0579 .
## Industry                  9.417019   0.848428  11.099   0.0572 .
## Coastlinecoastbyarearatio 1.369153   0.125781  10.885   0.0583 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.04927 on 1 degrees of freedom
## Multiple R-squared:  0.9992, Adjusted R-squared:  0.9959
## F-statistic: 302.5 on 4 and 1 DF,  p-value: 0.0431
```

Adjusted R-squared: 0.9959 p-value: 0.0431

```
#Western Europe Region
western_europe <- filter(happy_country, Region == "WESTERN EUROPE")
western_europe_nocat <- subset(western_europe, select = -c(Region, Country, Climate))
western_europe_cor <- western_europe_nocat %>%
  cor(western_europe_nocat)
```

Top Correlated Variables: Percieved Corruption (0.870394734), GDP per capita (0.854352684), Crops (-0.829523968).

```
#Linear Model to Predict Western European Happiness
WestEurolm <- lm(HappinessScore ~ PercievedCorruptionScore + GDP_percapita +
                Literacy + Agriculture + Deathrate, data = western_europe)
summary(WestEurolm)
```

```
##
## Call:
## lm(formula = HappinessScore ~ PercievedCorruptionScore + GDP_percapita +
##     Literacy + Agriculture + Deathrate, data = western_europe)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33542 -0.11413 -0.00060  0.09072  0.51110
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -5.498e+00  7.559e+00  -0.727  0.48373
## PercievedCorruptionScore  3.931e-02  1.019e-02   3.860  0.00316 **
## GDP_percapita             5.831e-05  3.030e-05   1.925  0.08314 .
## Literacy                  7.274e-03  8.651e-03   0.841  0.42010
## Agriculture               7.336e+00  4.032e+00   1.819  0.09888 .
## Deathrate                 4.646e-04  3.402e-04   1.366  0.20194
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2522 on 10 degrees of freedom
## Multiple R-squared:  0.9244, Adjusted R-squared:  0.8866
## F-statistic: 24.46 on 5 and 10 DF,  p-value: 2.623e-05
```

Adjusted R-squared: 0.8866 p-value: 2.623e-05

```r
#Latin America and Caribbean Region
latin_america_carib <- filter(happy_country, Region == "LATIN AMER. & CARIB")
latin_america_carib_nocat <- subset(latin_america_carib, select = -c(Region, Country, Climate))
latin_america_carib_cor <- latin_america_carib_nocat %>%
  cor(latin_america_carib_nocat)
```

Top Correlated Variables: Phonesper1000 (0.7221507), GDP_percapita (0.6595595), Literacy (0.6108366)

```r
#Linear Model to Predict Latin America and Caribbean Happiness
LACablm <- lm(HappinessScore ~ Phonesper1000 + Deathrate + Crops +
                Arable, data = latin_america_carib)
summary(LACablm)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Phonesper1000 + Deathrate + Crops +
##     Arable, data = latin_america_carib)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.87995 -0.15358  0.05095  0.16703  0.52144
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    6.0373385  0.2743938  22.002 7.88e-13 ***
## Phonesper1000  0.0066053  0.0009959   6.632 7.98e-06 ***
## Deathrate     -0.0016867  0.0003677  -4.588 0.000356 ***
## Crops         -0.0965245  0.0332576  -2.902 0.010943 *
## Arable         0.0290974  0.0171181   1.700 0.109804
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.344 on 15 degrees of freedom
## Multiple R-squared:  0.8495, Adjusted R-squared:  0.8094
## F-statistic: 21.17 on 4 and 15 DF,  p-value: 4.992e-06
```

Adjusted R-squared: 0.8094 p-value: 4.992e-06

```r
# Africa Region
happy_country2 <- happy_country
happy_country2$Region[happy_country2$Region == "NORTHERN AFRICA"] <- "AFRICA"
happy_country2$Region[happy_country2$Region == "SUB-SAHARAN AFRICA"] <- "AFRICA"
africa <- filter(happy_country2, Region == "AFRICA")
africa_nocat <- subset(africa, select = -c(Region, Country, Climate))
africa_cor <- africa_nocat %>%
  cor(africa_nocat)
```

Top Correlated Variables: Phonesper1000 (0.53851034), GDP_percapita (0.43636820), Birthrate (-0.43576128).

```r
#Linear Model to Predict Africa Happiness
Africalm <- lm(HappinessScore ~ Phonesper1000 + Birthrate + Crops +
                 PercievedCorruptionScore, data = africa)
summary(Africalm)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Phonesper1000 + Birthrate + Crops +
##     PercievedCorruptionScore, data = africa)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.84985 -0.37228  0.02929  0.32410  1.05239
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                5.081e+00  4.991e-01  10.181 3.01e-11 ***
## Phonesper1000              5.905e-03  1.964e-03   3.006   0.0053 **
## Birthrate                 -1.093e-04  8.957e-05  -1.220   0.2319
## Crops                     -3.811e-02  2.109e-02  -1.807   0.0809 .
## PercievedCorruptionScore -1.593e-02  9.022e-03  -1.766   0.0876 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4998 on 30 degrees of freedom
## Multiple R-squared:  0.4447, Adjusted R-squared:  0.3706
## F-statistic: 6.006 on 4 and 30 DF,  p-value: 0.00113
```

Adjusted R-squared: 0.3706 p-value: 0.00113

```r
# ASIA (EX. NEAR EAST) Region
asiaNE <- filter(happy_country2, Region == "ASIA (EX. NEAR EAST)")
asiaNE_nocat <- subset(asiaNE, select = -c(Region, Country, Climate))
asiaNE_cor <- asiaNE_nocat %>%
  cor(asiaNE_nocat)
```

Top Correlated Variables: Agriculture (-0.82178348), PercievedCorruptionScore (0.66801346), GDP_percapita (0.66188480).

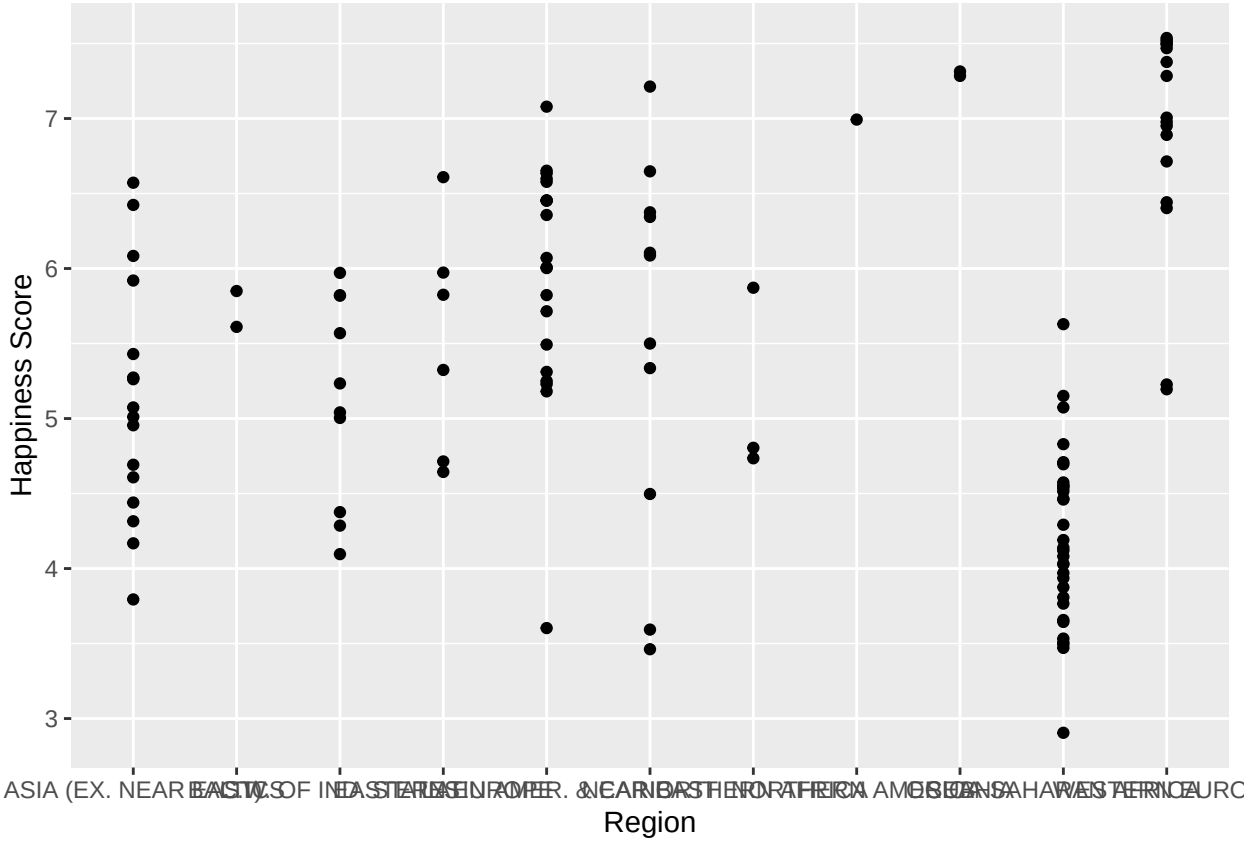
```r
#Linear Model to Predict ASIA (EX. NEAR EAST) Happiness
asiaNElm <- lm(HappinessScore ~ Agriculture + Phonesper1000 + Birthrate +
                 GDP_percapita, data = asiaNE)
summary(asiaNElm)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Agriculture + Phonesper1000 + Birthrate +
##     GDP_percapita, data = asiaNE)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.81632 -0.10865  0.04691  0.27222  0.90366
```

```
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.578e+00  6.569e-01   8.492 2.03e-06 ***
## Agriculture   -5.146e+00  2.050e+00  -2.510   0.0274 *
## Phonesper1000 -6.601e-04  1.885e-03  -0.350   0.7323
## Birthrate      2.022e-04  1.360e-04   1.487   0.1629
## GDP_percapita  3.284e-05  3.238e-05   1.014   0.3304
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.457 on 12 degrees of freedom
## Multiple R-squared:  0.7442, Adjusted R-squared:  0.6589
## F-statistic: 8.728 on 4 and 12 DF,  p-value: 0.001531
```

Adjusted R-squared: 0.6589 p-value: 0.001531

```
#Happiness Score sorted by Region
RegionPlot<- ggplot(happy_country, aes(x = Region, y = HappinessScore)) +
 geom_point() +
 labs(x="Region", y="Happiness Score")
RegionPlot
```



Modeling

Ridge and Lasso

```
library(broom)
xnew <- model.matrix(HappinessScore~., num_happy)[,-1]
ynew <- num_happy$HappinessScore
lambdanew <- 10^seq(10, -2, length = 100)
library(glmnet)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
## Loading required package: foreach
```

```
##
## Attaching package: 'foreach'
```

```
## The following objects are masked from 'package:purrr':
##
##     accumulate, when
```

```
## Loaded glmnet 2.0-16
```

```
set.seed(489)
trainnew <- sample(1:nrow(xnew), nrow(xnew)/2)
testnew<- (-trainnew)
ytestnew <- ynew[testnew]
#OLS
happylm <- lm(HappinessScore~., data = num_happy)
coef(happylm)
```

```
##                   (Intercept)                    Population
##                 -1.356365e+02                 -6.171659e-10
##                       Areasqmi              PopDensitypersqmi
##                  4.813662e-08                 -6.304624e-05
##     Coastlinecoastbyarearatio                   Netmigration
##                 -3.089872e-05                  4.155625e-03
## Infantmortalityper1000births                  GDP_percapita
##                 -5.444791e-05                  4.418409e-05
##                      Literacy                 Phonesper1000
##                  8.304203e-04                 -3.903325e-06
##                        Arable                         Crops
##                 -5.591452e-03                 -1.723483e-02
##                         Other                       Climate
##                  1.703962e-05                  1.736171e-02
##                     Birthrate                     Deathrate
##                 -3.154149e-05                 -2.826559e-04
##                   Agriculture                      Industry
```

```
##                  1.394043e+02                  1.400957e+02
##                       Service     PercievedCorruptionScore
##                  1.402940e+02                  6.893027e-03
```

```
sldhappy <- summary(happylm)
sldhappy
```

```
##
## Call:
## lm(formula = HappinessScore ~ ., data = num_happy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.01482 -0.35888  0.00955  0.42737  1.37830
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -1.356e+02  6.136e+01  -2.210   0.0294 *
## Population                    -6.172e-10  5.966e-10  -1.034   0.3034
## Areasqmi                       4.814e-08  5.263e-08   0.915   0.3626
## PopDensitypersqmi             -6.305e-05  1.389e-04  -0.454   0.6510
## Coastlinecoastbyarearatio     -3.090e-05  2.138e-02  -0.001   0.9988
## Netmigration                   4.156e-03  1.974e-02   0.211   0.8337
## Infantmortalityper1000births -5.445e-05  2.652e-05  -2.053   0.0426 *
## GDP_percapita                  4.418e-05  2.023e-05   2.184   0.0313 *
## Literacy                       8.304e-04  4.966e-04   1.672   0.0976 .
## Phonesper1000                 -3.903e-06  9.400e-04  -0.004   0.9967
## Arable                        -5.591e-03  5.907e-03  -0.947   0.3461
## Crops                         -1.723e-02  1.835e-02  -0.939   0.3499
## Other                          1.704e-05  3.311e-05   0.515   0.6079
## Climate                        1.736e-02  1.769e-02   0.982   0.3287
## Birthrate                     -3.154e-05  7.373e-05  -0.428   0.6697
## Deathrate                     -2.827e-04  1.431e-04  -1.976   0.0509 .
## Agriculture                    1.394e+02  6.147e+01   2.268   0.0255 *
## Industry                       1.401e+02  6.143e+01   2.281   0.0247 *
## Service                        1.403e+02  6.151e+01   2.281   0.0247 *
## PercievedCorruptionScore       6.893e-03  6.146e-03   1.122   0.2647
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6486 on 100 degrees of freedom
## Multiple R-squared:  0.7423, Adjusted R-squared:  0.6933
## F-statistic: 15.16 on 19 and 100 DF,  p-value: < 2.2e-16
```

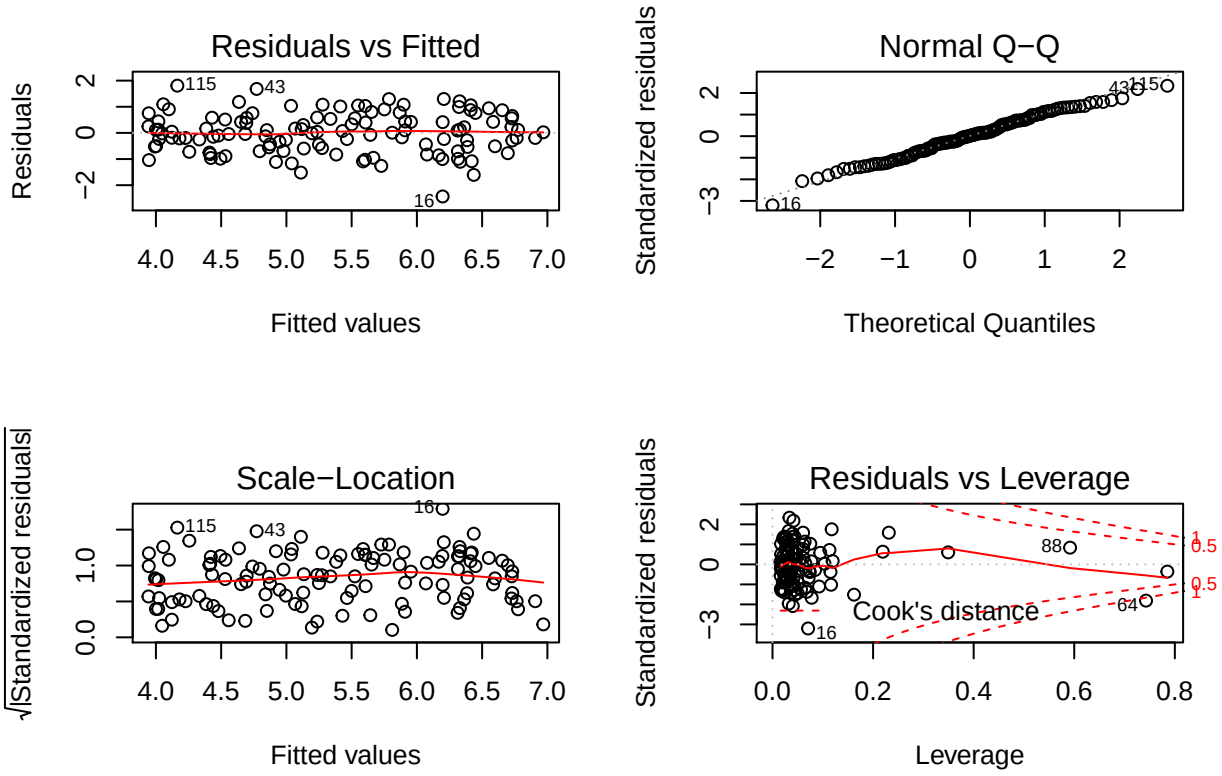
```
olscoeftab<- coef(sldhappy)
```

```
olscoeftab
```

```
##                                   Estimate   Std. Error     t value
## (Intercept)                   -1.356365e+02 6.136217e+01 -2.21042600
## Population                    -6.171659e-10 5.966479e-10 -1.03438879
## Areasqmi                       4.813662e-08 5.262784e-08  0.91466083
```

```
## PopDensitypersqmi            -6.304624e-05 1.389446e-04 -0.45375083
## Coastlinecoastbyarearatio    -3.089872e-05 2.137668e-02 -0.00144544
## Netmigration                  4.155625e-03 1.973986e-02  0.21051949
## Infantmortalityper1000births -5.444791e-05 2.651520e-05 -2.05346073
## GDP_percapita                 4.418409e-05 2.022695e-05  2.18441691
## Literacy                      8.304203e-04 4.965761e-04  1.67229215
## Phonesper1000                -3.903325e-06 9.399736e-04 -0.00415259
## Arable                       -5.591452e-03 5.906876e-03 -0.94660046
## Crops                        -1.723483e-02 1.834990e-02 -0.93923313
## Other                         1.703962e-05 3.310719e-05  0.51468041
## Climate                       1.736171e-02 1.768677e-02  0.98162102
## Birthrate                    -3.154149e-05 7.373128e-05 -0.42778981
## Deathrate                    -2.826559e-04 1.430526e-04 -1.97588832
## Agriculture                   1.394043e+02 6.146660e+01  2.26796814
## Industry                      1.400957e+02 6.142608e+01  2.28072088
## Service                       1.402940e+02 6.150698e+01  2.28094470
## PercievedCorruptionScore      6.893027e-03 6.145905e-03  1.12156423
##                                Pr(>|t|)
## (Intercept)                  0.02935500
## Population                   0.30344947
## Areasqmi                     0.36256995
## PopDensitypersqmi            0.65099139
## Coastlinecoastbyarearatio    0.99884959
## Netmigration                 0.83369061
## Infantmortalityper1000births 0.04263811
## GDP_percapita                0.03126829
## Literacy                     0.09759285
## Phonesper1000                0.99669499
## Arable                       0.34612419
## Crops                        0.34987419
## Other                        0.60791206
## Climate                      0.32865555
## Birthrate                    0.66972401
## Deathrate                    0.05092280
## Agriculture                  0.02548170
## Industry                     0.02468668
## Service                      0.02467292
## PercievedCorruptionScore     0.26473375
```

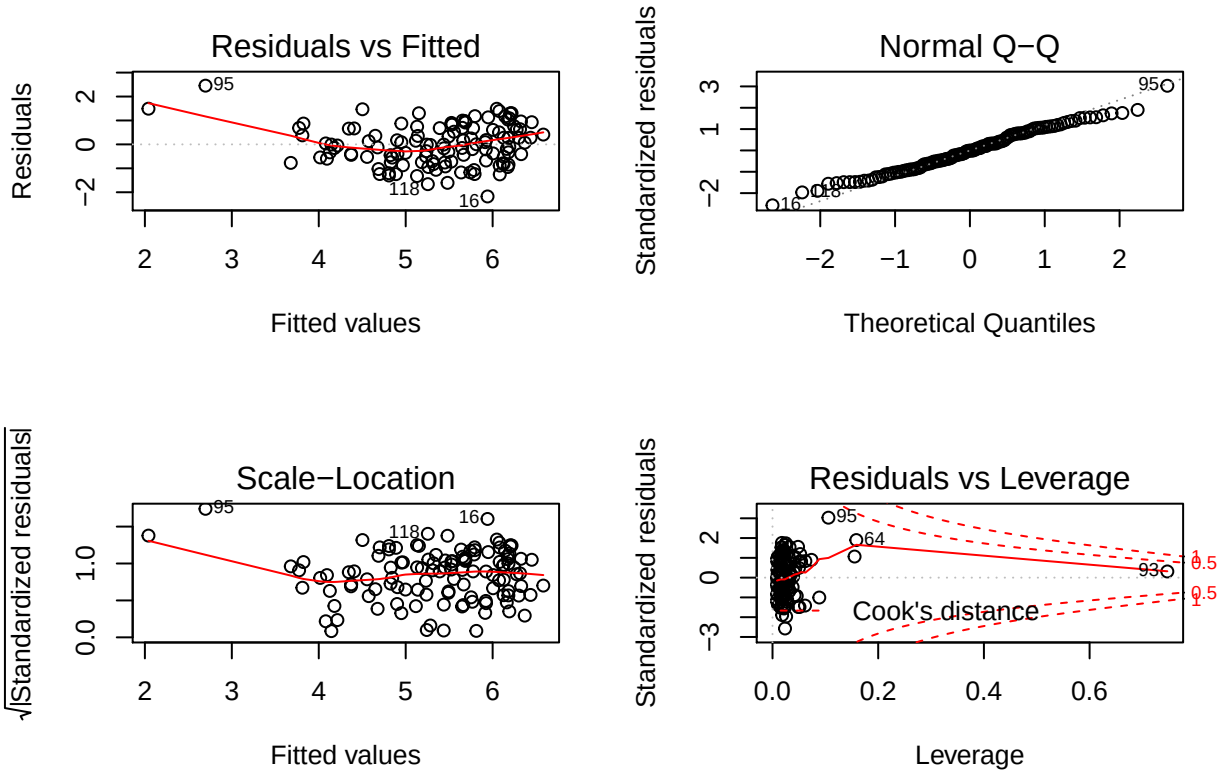
```r
#OLS with interaction
olsmod<- lm(HappinessScore~Agriculture*Industry*Service, data=num_happy)
par(mfrow = c(2, 2))
OLS<- plot(olsmod)
```




```
OLSpred<-predict(olsmod, newdata=num_happy[testnew,])
OLSMSE<- mean((OLSpred-ytestnew)^2)
OLSMSE
```

```
## [1] 0.4590039
```

```
#OLS without interaction
olsmod2<- lm(HappinessScore~Agriculture+Industry+Service, data=num_happy)
par(mfrow = c(2, 2))
OLS2<- plot(olsmod2)
```




```
OLSpred2<- predict(olsmod2, newdata = num_happy[testnew,])
OLS2MSE <- mean((OLSpred2-ytestnew)^2)
OLS2MSE
```

```
## [1] 0.576744
```

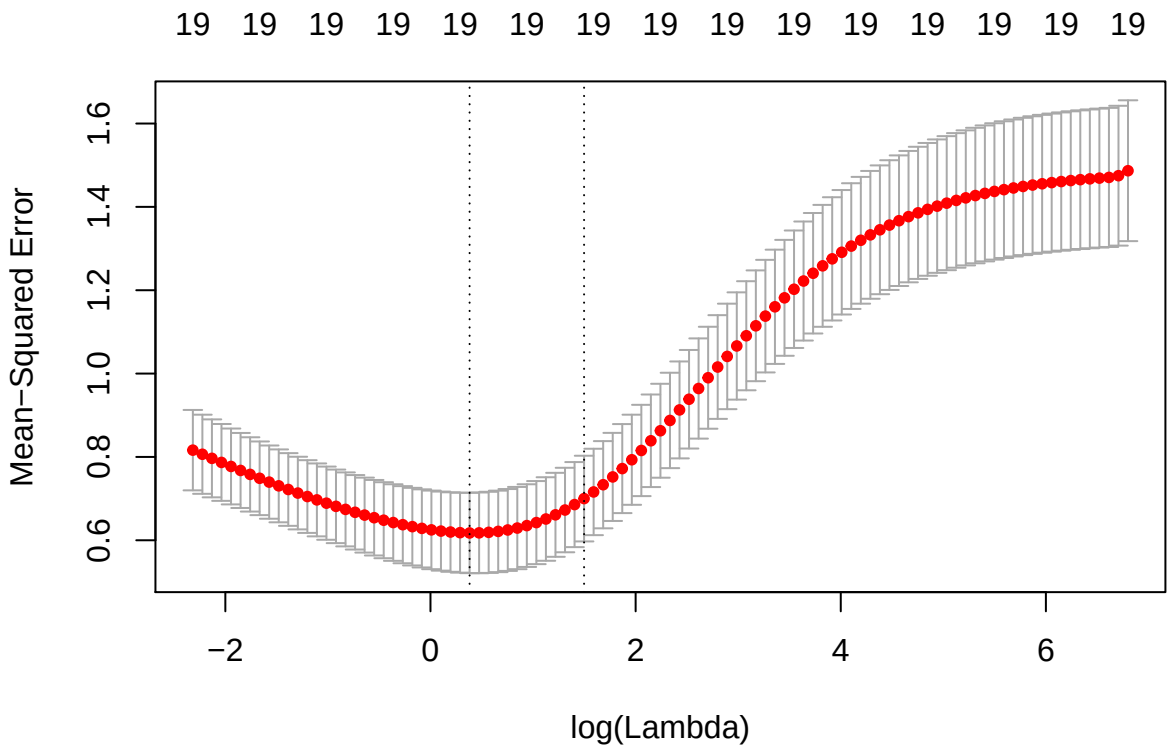
```
#Clearly interaction is much better
#ridge
happyridge <- glmnet(xnew, ynew, alpha = 0, lambda = lambdanew)
predict(happyridge, s = 0, exact = FALSE, type = 'coefficients')[1:20,]
```

```
##                   (Intercept)                    Population
##                  3.838453e+00                  -5.488042e-10
##                      Areasqmi                PopDensitypersqmi
##                  4.390964e-08                  -5.299996e-05
##     Coastlinecoastbyarearatio                     Netmigration
##                  7.310422e-04                   1.908362e-03
## Infantmortalityper1000births                     GDP_percapita
##                 -4.619738e-05                   3.815745e-05
##                       Literacy                    Phonesper1000
##                  8.134111e-04                   4.137226e-04
##                         Arable                           Crops
##                 -5.824861e-03                  -1.400190e-02
##                          Other                         Climate
##                  2.852235e-05                   1.901622e-02
##                     Birthrate                      Deathrate
##                 -1.626183e-05                  -2.133644e-04
##                    Agriculture                       Industry
```

```
##                -3.250528e-01                 4.780759e-01
##                      Service   PercievedCorruptionScore
##                 4.963853e-01                 7.608010e-03
```

```
happylm1 <- lm(HappinessScore~., data = num_happy)
happyridge1 <- glmnet(xnew[trainnew,], ynew[trainnew], alpha = 0, lambda = lambdanew)
#find the best lambda from our list via cross-validation
```

```
cv.out <- cv.glmnet(xnew[trainnew,], ynew[trainnew], alpha = 0)
plot(cv.out)
```




```
bestlamnew <- cv.out$lambda.min
bestlamnew
```

```
## [1] 1.464055
```

```
#make predictions
rpred1<- predict(happyridge, s=bestlamnew, newx = xnew[testnew,])
ridge.prednew <- predict(happyridge1, s = bestlamnew, newx = xnew[testnew,])
s.prednew <- predict(happylm1, newdata = num_happy[testnew,])
#check MSE
rmse1<- mean((rpred1-ytestnew)^2)
rmse1
```

```
## [1] 0.3551861
```

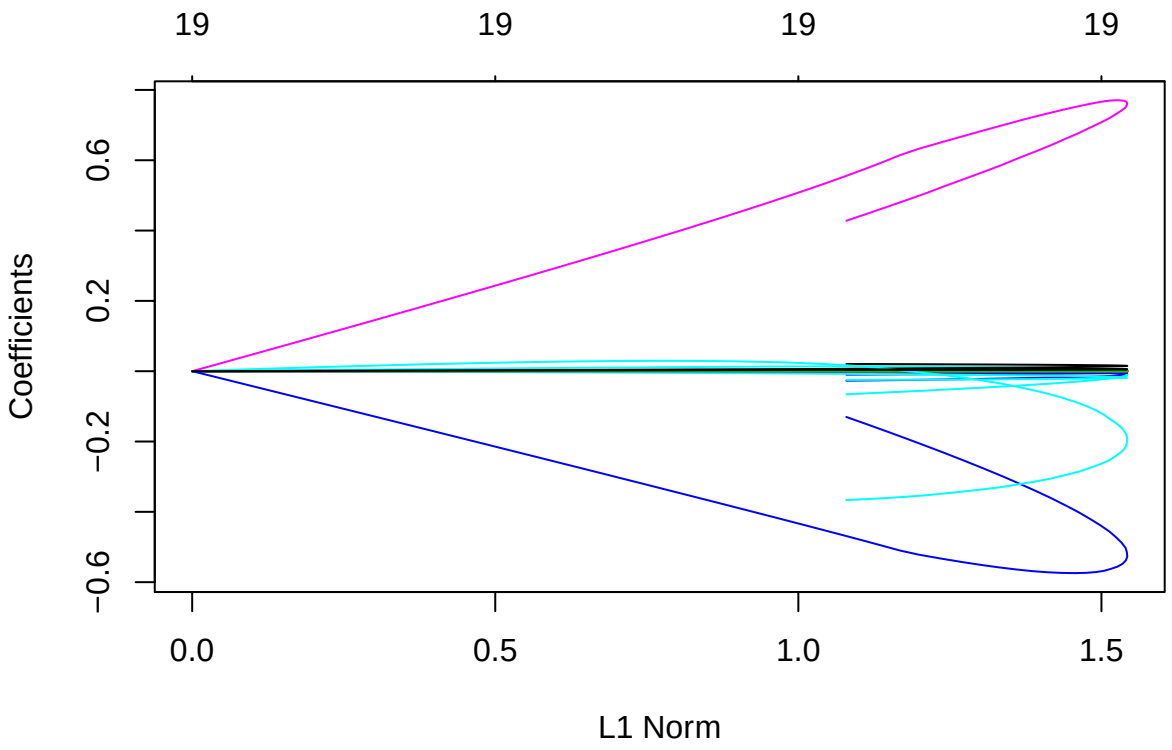
```
MSEnew<- mean((s.prednew-ytestnew)^2)
MSEnew
```

```
## [1] 0.2539225
```

```
RidgeMSE<- mean((ridge.prednew-ytestnew)^2)
RidgeMSE
```

```
## [1] 0.4433365
```

```
out <- glmnet(xnew[trainnew,],ynew[trainnew],alpha = 0)
plot(out)
```

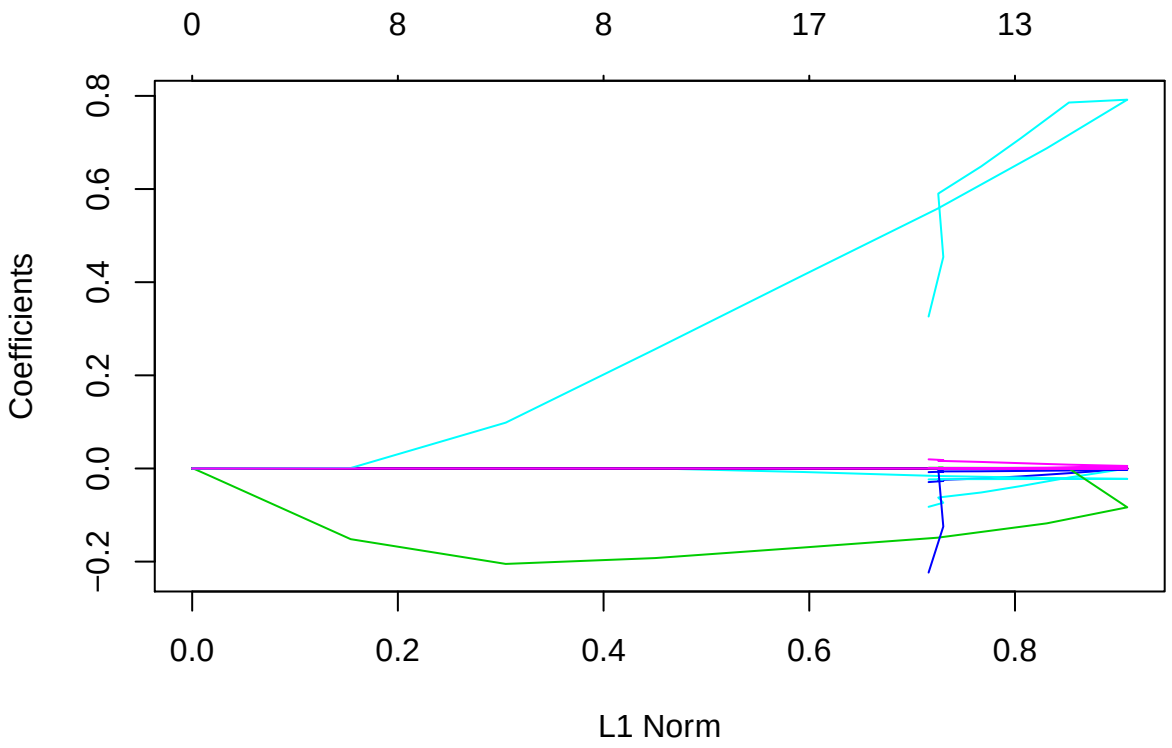



```
ridgepredictnew<-predict(happyridge1, type = "coefficients", s = bestlamnew)[1:20,]
ridgepredictnew
```

```
##                   (Intercept)                    Population
##                  4.664019e+00                 -1.879487e-10
##                      Areasqmi               PopDensitypersqmi
##                  1.659712e-08                 -8.173089e-05
##     Coastlinecoastbyarearatio                   Netmigration
##                 -1.420121e-03                  7.543369e-03
## Infantmortalityper1000births                   GDP_percapita
##                 -3.579342e-05                  1.349721e-05
##                       Literacy                  Phonesper1000
##                  5.504922e-04                  6.432719e-04
##                         Arable                         Crops
```

```
##                -2.807108e-03                -1.333805e-02
##                        Other                      Climate
##                 1.434557e-05                 1.140728e-02
##                    Birthrate                    Deathrate
##                -7.650915e-05                -1.593895e-04
##                  Agriculture                     Industry
##                -5.727619e-01                -7.594614e-02
##                      Service  PercievedCorruptionScore
##                 7.425383e-01                 4.973838e-03
```

```
##Lasso
lasso.mod <- glmnet(xnew[trainnew,], ynew[trainnew], alpha = 1, lambda = lambdanew)
plot(lasso.mod)
```




```
lasso.pred <- predict(lasso.mod, s = bestlamnew, newx = xnew[testnew,])
LassoMSE<-mean((lasso.pred-ytestnew)^2)
LassoMSE
```

```
## [1] 1.370192
```

```
lasso.coef  <- predict(lasso.mod, type = 'coefficients', s = bestlamnew)[1:20,]
lasso.coef
```

```
##                   (Intercept)                    Population
##                       5.57865                       0.00000
##                      Areasqmi             PopDensitypersqmi
##                       0.00000                       0.00000
##     Coastlinecoastbyarearatio                   Netmigration
```

```
##                       0.00000                       0.00000
## Infantmortalityper1000births                GDP_percapita
##                       0.00000                       0.00000
##                      Literacy                Phonesper1000
##                       0.00000                       0.00000
##                        Arable                         Crops
##                       0.00000                       0.00000
##                         Other                       Climate
##                       0.00000                       0.00000
##                     Birthrate                     Deathrate
##                       0.00000                       0.00000
##                   Agriculture                      Industry
##                       0.00000                       0.00000
##                       Service      PercievedCorruptionScore
##                       0.00000                       0.00000
```

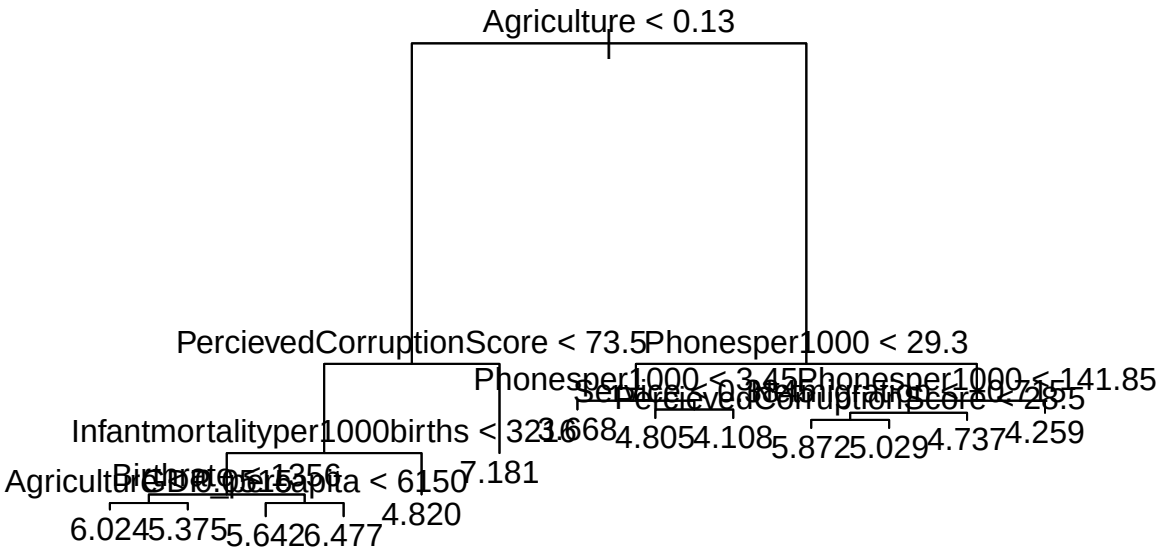
Regression Trees

```
library(tree)
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

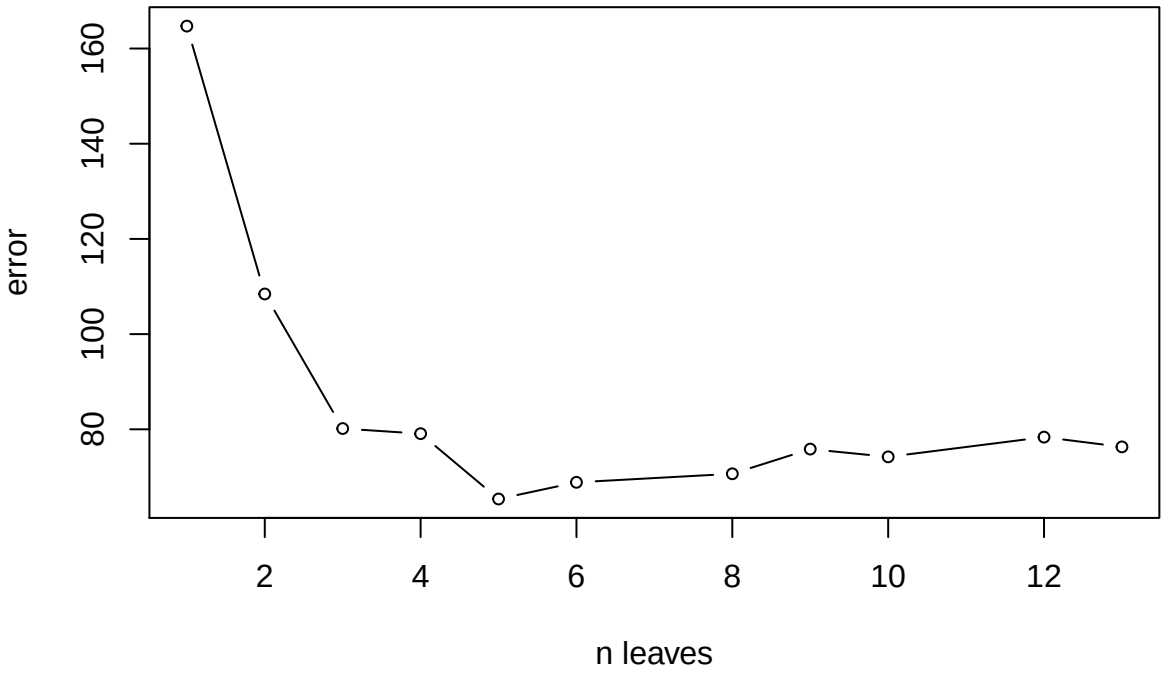
```
regtree<- tree(HappinessScore ~ .-HappinessScore,data=num_happy)
plot(regtree)
text(regtree, pretty=0)
```



```r
#Prune
tcv <- cv.tree(regtree, FUN = prune.tree)
tcv
```

```
## $size
##  [1] 13 12 10  9  8  6  5  4  3  2  1
##
## $dev
##  [1]  76.30909  78.35814  74.21352  75.84469  70.65663  68.86167  65.35966
##  [8]  79.08520  80.16017 108.43339 164.71068
##
## $k
##  [1]      -Inf  1.662375  1.950806  2.188930  2.263204  2.795167  3.095401
##  [8]  9.459158 10.620794 22.912871 82.290626
##
## $method
## [1] "deviance"
##
## attr(,"class")
## [1] "prune"         "tree.sequence"
```

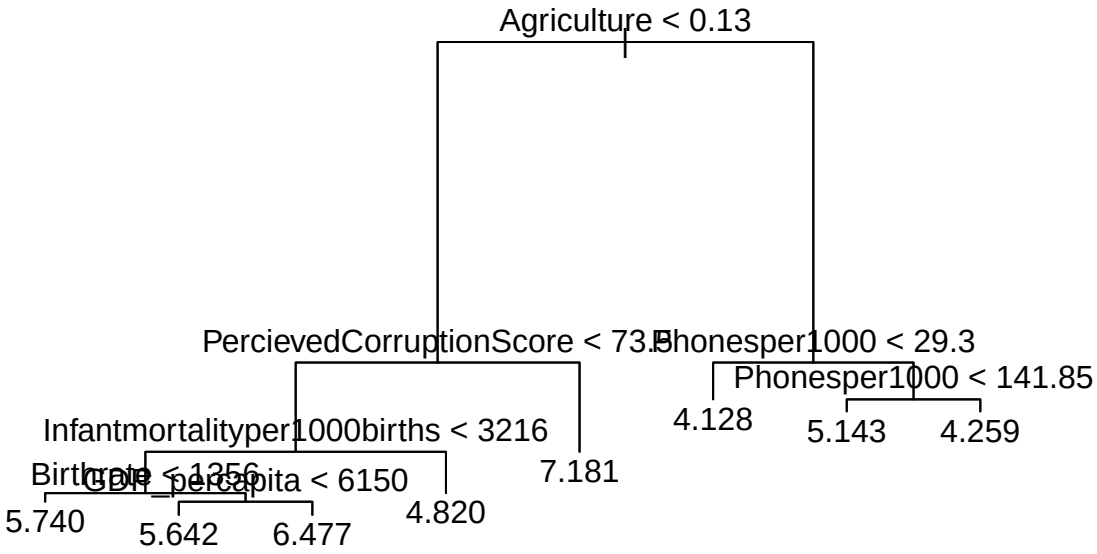
```r
plot(tcv$size, tcv$dev, type = "b", xlab = "n leaves", ylab = "error", cex=.75)
```




```r
tcv$size[which.min(tcv$dev)]
```

```
## [1] 5
```

```r
tprune <- prune.tree(regtree, best = 8)
plot(tprune)
text(tprune, pretty = 0)
```




```
tree_est <- predict(tprune, newdata=num_happy)
MSE_test<- mean((tree_est - num_happy$HappinessScore)^2)
MSE_test
```

```
## [1] 0.2440981
```

Boosted Tree ## Boost

```
set.seed(1)
train <- sample(1:nrow(num_happy), nrow(num_happy) * .75)
traind <- num_happy[train, ]
testd <- num_happy[-train, ]
library(gbm)
```

```
## Loaded gbm 2.1.5
```

```
boost1 <- gbm(HappinessScore ~ .-HappinessScore,data=traind,
                     distribution = "multinomial", n.trees = 50,
                     shrinkage = 0.1, interaction.depth = 1)
pred66 <- predict(boost1, newdata = testd, n.trees = 50, type = "response", shrinkage = 0.1, interactio
MSE_test2<- mean((pred66 - testd$HappinessScore)^2)
MSE_test2
```

```
## [1] 28.22945
```

Random Forest

```
library(randomForest)
```

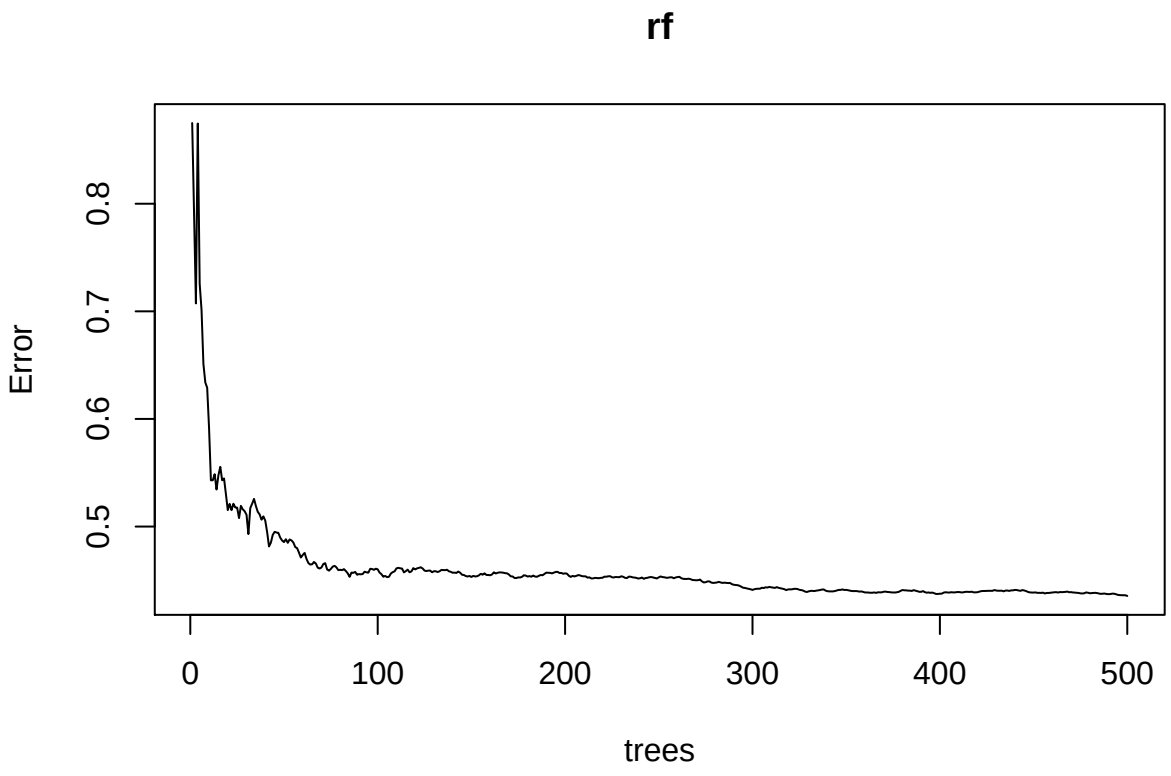
```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:dplyr':
##
##     combine

## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
rf <- randomForest(HappinessScore ~ .-HappinessScore, data = traind, importance = TRUE)
rf
```

```
##
## Call:
##  randomForest(formula = HappinessScore ~ . - HappinessScore, data = traind,      importance = TRUE)
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 6
##
##           Mean of squared residuals: 0.4355214
##                     % Var explained: 69.64
```

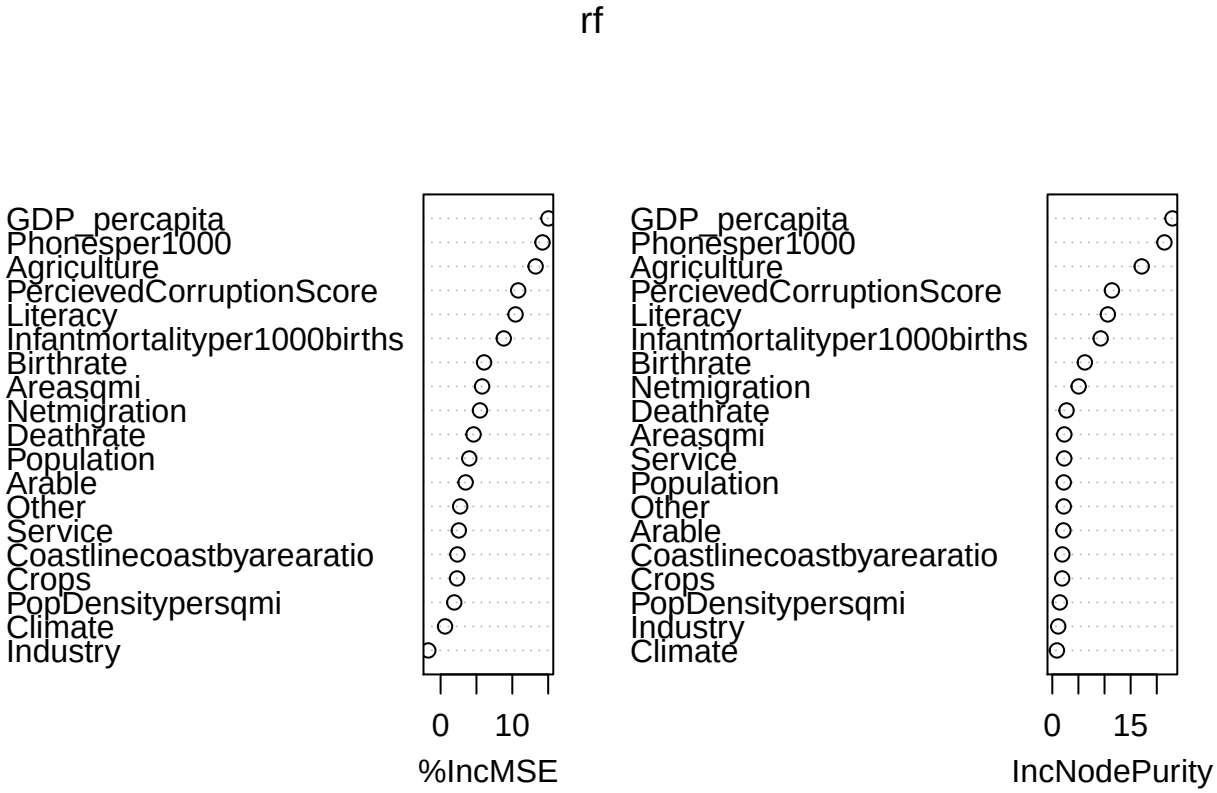
```
plot(rf)
```



```
testrf <- predict(rf, newdata=testd)
MSE2<- mean((testrf - testd$HappinessScore)^2)
MSE2
```

```
## [1] 0.4458951
```

```
varImpPlot(rf)
```



LINEAR MODELS

```
#Region Linear Model
alymod<- lm(HappinessScore ~ Region, data = happy_country)
summary(alymod)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Region, data = happy_country)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.34440 -0.43511  0.03602  0.50886  1.65300
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 5.134765   0.185105  27.740  < 2e-16 ***
## RegionBALTICS               0.595735   0.570532   1.044 0.298715
## RegionC.W. OF IND. STATES -0.012965   0.304159  -0.043 0.966079
```

```
## RegionEASTERN EUROPE       0.380069   0.362415   1.049 0.296631
## RegionLATIN AMER. & CARIB  0.812635   0.251770   3.228 0.001649 **
## RegionNEAR EAST            0.425235   0.295325   1.440 0.152765
## RegionNORTHERN AFRICA      0.002569   0.477939   0.005 0.995722
## RegionNORTHERN AMERICA     1.858235   0.785334   2.366 0.019739 *
## RegionOCEANIA              2.164235   0.570532   3.793 0.000244 ***
## RegionSUB-SAHARAN AFRICA  -0.941421   0.229056  -4.110 7.69e-05 ***
## RegionWESTERN EUROPE       1.739798   0.265837   6.545 2.01e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7632 on 109 degrees of freedom
## Multiple R-squared:  0.6111, Adjusted R-squared:  0.5754
## F-statistic: 17.13 on 10 and 109 DF,  p-value: < 2.2e-16
```

Adjusted R-squared: 0.5754

```
#GDP + Arable Land + Infant Mortality + Percieved Corruption
alymod0 <- lm(HappinessScore ~ GDP_percapita + Arable +
                Infantmortalityper1000births + PercievedCorruptionScore,
              data = happy_country)
summary(alymod0)
```

```
##
## Call:
## lm(formula = HappinessScore ~ GDP_percapita + Arable + Infantmortalityper1000births +
##     PercievedCorruptionScore, data = happy_country)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7165 -0.4287  0.0272  0.4243  1.4488
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   5.088e+00  2.269e-01  22.428  < 2e-16 ***
## GDP_percapita                 5.787e-05  1.267e-05   4.566 1.26e-05 ***
## Arable                       -1.233e-02  4.358e-03  -2.830  0.00549 **
## Infantmortalityper1000births -1.023e-04  2.025e-05  -5.054 1.65e-06 ***
## PercievedCorruptionScore      7.209e-03  5.920e-03   1.218  0.22581
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6694 on 115 degrees of freedom
## Multiple R-squared:  0.6844, Adjusted R-squared:  0.6734
## F-statistic: 62.35 on 4 and 115 DF,  p-value: < 2.2e-16
```

Adjusted R-squared: 0.6734

```
#Region + GDP + Infant Mortality + Percieved Corruption
alymod2 <- lm(HappinessScore ~ Region + GDP_percapita +
                Infantmortalityper1000births + PercievedCorruptionScore,
              data = happy_country)
summary(alymod2)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Region + GDP_percapita + Infantmortalityper1000births +
##     PercievedCorruptionScore, data = happy_country)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.74707 -0.34632  0.04397  0.41472  1.17032
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   4.450e+00  2.498e-01  17.813  < 2e-16 ***
## RegionBALTICS                -6.787e-02  4.455e-01  -0.152   0.8792
## RegionC.W. OF IND. STATES     2.615e-01  2.338e-01   1.119   0.2659
## RegionEASTERN EUROPE         -4.507e-02  2.803e-01  -0.161   0.8726
## RegionLATIN AMER. & CARIB     8.270e-01  1.950e-01   4.241 4.77e-05 ***
## RegionNEAR EAST              -7.129e-02  2.389e-01  -0.298   0.7660
## RegionNORTHERN AFRICA         3.110e-02  3.660e-01   0.085   0.9324
## RegionNORTHERN AMERICA       -1.031e+00  7.072e-01  -1.458   0.1479
## RegionOCEANIA                 2.304e-01  4.832e-01   0.477   0.6345
## RegionSUB-SAHARAN AFRICA     -4.824e-01  1.856e-01  -2.600   0.0107 *
## RegionWESTERN EUROPE         -3.541e-01  3.166e-01  -1.118   0.2660
## GDP_percapita                 8.002e-05  1.580e-05   5.063 1.75e-06 ***
## Infantmortalityper1000births -2.952e-05  2.028e-05  -1.456   0.1484
## PercievedCorruptionScore      7.340e-03  5.569e-03   1.318   0.1903
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5788 on 106 degrees of freedom
## Multiple R-squared:  0.7825, Adjusted R-squared:  0.7558
## F-statistic: 29.34 on 13 and 106 DF,  p-value: < 2.2e-16
```

Adjusted R-squared: 0.7558

```
#Region + GDP + Arable Land + Percieved Corruption
alymod3 <- lm(HappinessScore ~ Region + GDP_percapita + Arable +
                PercievedCorruptionScore, data = happy_country)
summary(alymod3)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Region + GDP_percapita + Arable +
##     PercievedCorruptionScore, data = happy_country)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.74005 -0.37312  0.04332  0.37535  1.31064
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                4.461e+00  2.409e-01  18.519  < 2e-16 ***
## RegionBALTICS              4.291e-02  4.412e-01   0.097  0.92271
## RegionC.W. OF IND. STATES  2.531e-01  2.323e-01   1.090  0.27840
```

```
## RegionEASTERN EUROPE       1.856e-01  2.917e-01   0.636  0.52604
## RegionLATIN AMER. & CARIB  8.128e-01  1.937e-01   4.196 5.66e-05 ***
## RegionNEAR EAST           -1.060e-01  2.389e-01  -0.444  0.65802
## RegionNORTHERN AFRICA      9.807e-03  3.638e-01   0.027  0.97854
## RegionNORTHERN AMERICA    -1.002e+00  7.029e-01  -1.426  0.15681
## RegionOCEANIA              1.764e-01  4.807e-01   0.367  0.71432
## RegionSUB-SAHARAN AFRICA  -6.068e-01  1.800e-01  -3.371  0.00105 **
## RegionWESTERN EUROPE      -2.995e-01  3.167e-01  -0.946  0.34642
## GDP_percapita              8.288e-05  1.529e-05   5.420 3.77e-07 ***
## Arable                    -7.973e-03  4.236e-03  -1.882  0.06252 .
## PercievedCorruptionScore   7.392e-03  5.533e-03   1.336  0.18439
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.575 on 106 degrees of freedom
## Multiple R-squared:  0.7853, Adjusted R-squared:  0.759
## F-statistic: 29.83 on 13 and 106 DF,  p-value: < 2.2e-16
```

Adjusted R-squared: 0.7590

```
#Region + GDP + Arable Land + Infant Mortality
alymod4 <- lm(HappinessScore ~ Region + GDP_percapita + Arable +
                Infantmortalityper1000births, data = happy_country)
summary(alymod4)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Region + GDP_percapita + Arable +
##     Infantmortalityper1000births, data = happy_country)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.61211 -0.34943  0.06897  0.38631  1.21569
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   4.903e+00  2.163e-01  22.671  < 2e-16 ***
## RegionBALTICS                 7.436e-02  4.326e-01   0.172 0.863849
## RegionC.W. OF IND. STATES     2.030e-01  2.296e-01   0.884 0.378523
## RegionEASTERN EUROPE          1.537e-01  2.901e-01   0.530 0.597351
## RegionLATIN AMER. & CARIB     7.123e-01  1.956e-01   3.641 0.000421 ***
## RegionNEAR EAST              -2.232e-01  2.324e-01  -0.960 0.339049
## RegionNORTHERN AFRICA        -1.194e-01  3.644e-01  -0.328 0.743759
## RegionNORTHERN AMERICA       -1.089e+00  6.858e-01  -1.589 0.115127
## RegionOCEANIA                 2.229e-01  4.758e-01   0.469 0.640351
## RegionSUB-SAHARAN AFRICA     -5.290e-01  1.840e-01  -2.875 0.004881 **
## RegionWESTERN EUROPE         -3.070e-01  3.132e-01  -0.980 0.329295
## GDP_percapita                 8.892e-05  1.189e-05   7.479 2.28e-11 ***
## Arable                       -9.393e-03  4.272e-03  -2.199 0.030065 *
## Infantmortalityper1000births -3.770e-05  2.032e-05  -1.855 0.066308 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.5706 on 106 degrees of freedom
## Multiple R-squared:  0.7886, Adjusted R-squared:  0.7627
## F-statistic: 30.42 on 13 and 106 DF,  p-value: < 2.2e-16
```

Adjusted R-squared: 0.7627

```
#Region + GDP + Arable Land + Infant Mortality + Percieved Corruption
alymod5 <- lm(HappinessScore ~ Region + GDP_percapita + Arable +
              Infantmortalityper1000births + PercievedCorruptionScore,
              data = happy_country)
summary(alymod5)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Region + GDP_percapita + Arable +
##     Infantmortalityper1000births + PercievedCorruptionScore,
##     data = happy_country)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.56448 -0.33297  0.05501  0.38725  1.25961
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   4.690e+00  2.683e-01  17.476  < 2e-16 ***
## RegionBALTICS                -2.957e-02  4.380e-01  -0.068 0.946297
## RegionC.W. OF IND. STATES     2.343e-01  2.300e-01   1.019 0.310526
## RegionEASTERN EUROPE          1.488e-01  2.891e-01   0.515 0.607898
## RegionLATIN AMER. & CARIB     7.375e-01  1.958e-01   3.767 0.000273 ***
## RegionNEAR EAST              -1.524e-01  2.375e-01  -0.642 0.522499
## RegionNORTHERN AFRICA        -8.966e-02  3.637e-01  -0.247 0.805753
## RegionNORTHERN AMERICA       -9.055e-01  6.970e-01  -1.299 0.196721
## RegionOCEANIA                 1.769e-01  4.753e-01   0.372 0.710513
## RegionSUB-SAHARAN AFRICA     -5.255e-01  1.833e-01  -2.867 0.005014 **
## RegionWESTERN EUROPE         -2.636e-01  3.137e-01  -0.840 0.402704
## GDP_percapita                 7.520e-05  1.568e-05   4.796 5.36e-06 ***
## Arable                       -9.379e-03  4.256e-03  -2.204 0.029732 *
## Infantmortalityper1000births -3.748e-05  2.025e-05  -1.852 0.066906 .
## PercievedCorruptionScore      7.311e-03  5.471e-03   1.336 0.184327
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5685 on 105 degrees of freedom
## Multiple R-squared:  0.7921, Adjusted R-squared:  0.7644
## F-statistic: 28.58 on 14 and 105 DF,  p-value: < 2.2e-16
```

Adjusted R-squared: 0.7644

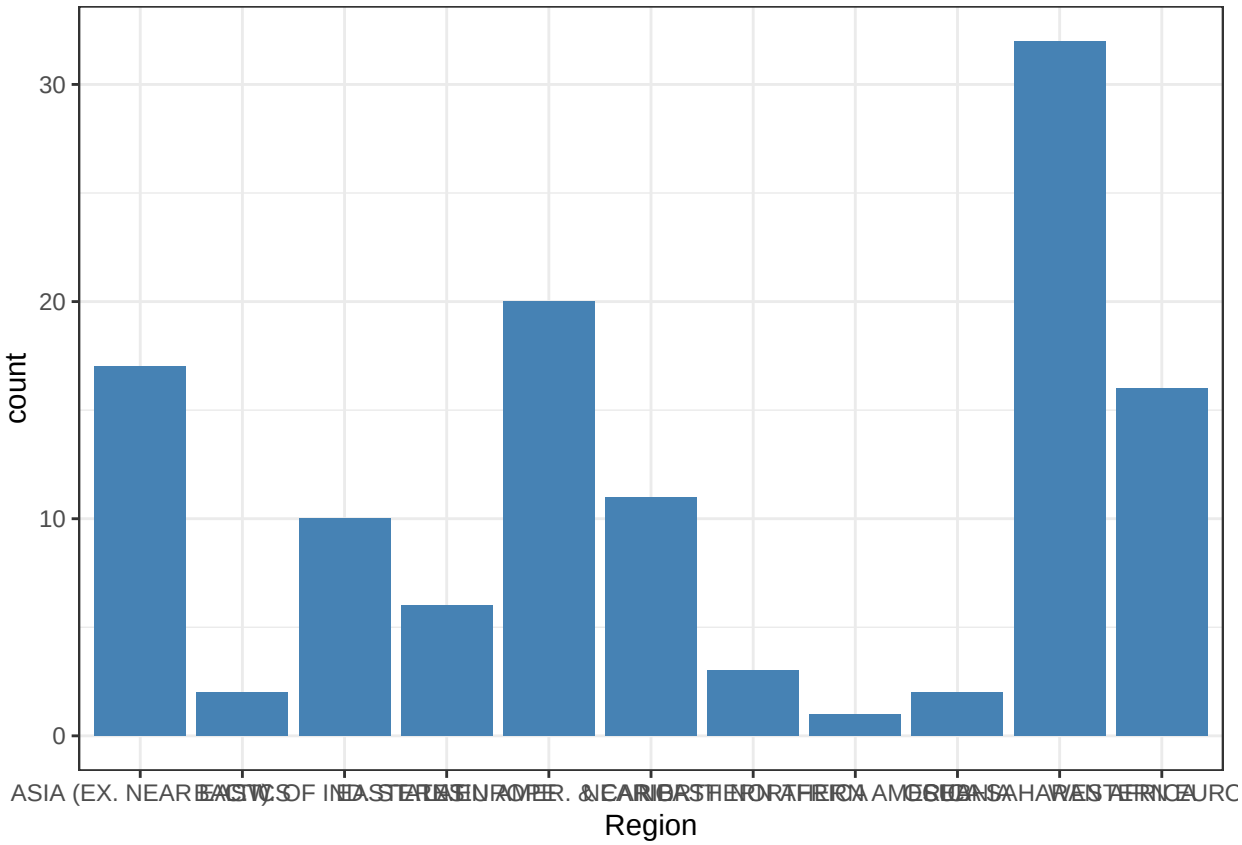
```
#Region + GDP + Arable Land + Infant Mortality + Percieved Corruption + Coastline Area
alymod6 <- lm(HappinessScore ~ Region + GDP_percapita + Arable +
              Infantmortalityper1000births + PercievedCorruptionScore +
                Coastlinecoastbyarearatio, data = happy_country)
summary(alymod6)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Region + GDP_percapita + Arable +
##     Infantmortalityper1000births + PercievedCorruptionScore +
##     Coastlinecoastbyarearatio, data = happy_country)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.46905 -0.34778  0.04081  0.39245  1.20608
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  4.711e+00  2.688e-01  17.524  < 2e-16 ***
## RegionBALTICS               -3.740e-02  4.377e-01  -0.085 0.932070
## RegionC.W. OF IND. STATES    1.965e-01  2.324e-01   0.846 0.399709
## RegionEASTERN EUROPE         7.450e-02  2.968e-01   0.251 0.802315
## RegionLATIN AMER. & CARIB    7.118e-01  1.971e-01   3.612 0.000469 ***
## RegionNEAR EAST             -1.809e-01  2.388e-01  -0.758 0.450295
## RegionNORTHERN AFRICA       -1.326e-01  3.655e-01  -0.363 0.717451
## RegionNORTHERN AMERICA      -1.122e+00  7.244e-01  -1.549 0.124338
## RegionOCEANIA                7.531e-02  4.840e-01   0.156 0.876642
## RegionSUB-SAHARAN AFRICA    -5.478e-01  1.843e-01  -2.972 0.003676 **
## RegionWESTERN EUROPE        -3.785e-01  3.308e-01  -1.144 0.255226
## GDP_percapita                8.056e-05  1.642e-05   4.905 3.46e-06 ***
## Arable                      -9.095e-03  4.261e-03  -2.135 0.035141 *
## Infantmortalityper1000births -3.826e-05  2.024e-05  -1.890 0.061537 .
## PercievedCorruptionScore     7.186e-03  5.467e-03   1.314 0.191588
## Coastlinecoastbyarearatio   -1.559e-02  1.435e-02  -1.086 0.279878
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.568 on 104 degrees of freedom
## Multiple R-squared:  0.7945, Adjusted R-squared:  0.7648
## F-statistic:  26.8 on 15 and 104 DF,  p-value: < 2.2e-16
```

Adjusted R-squared: 0.7648

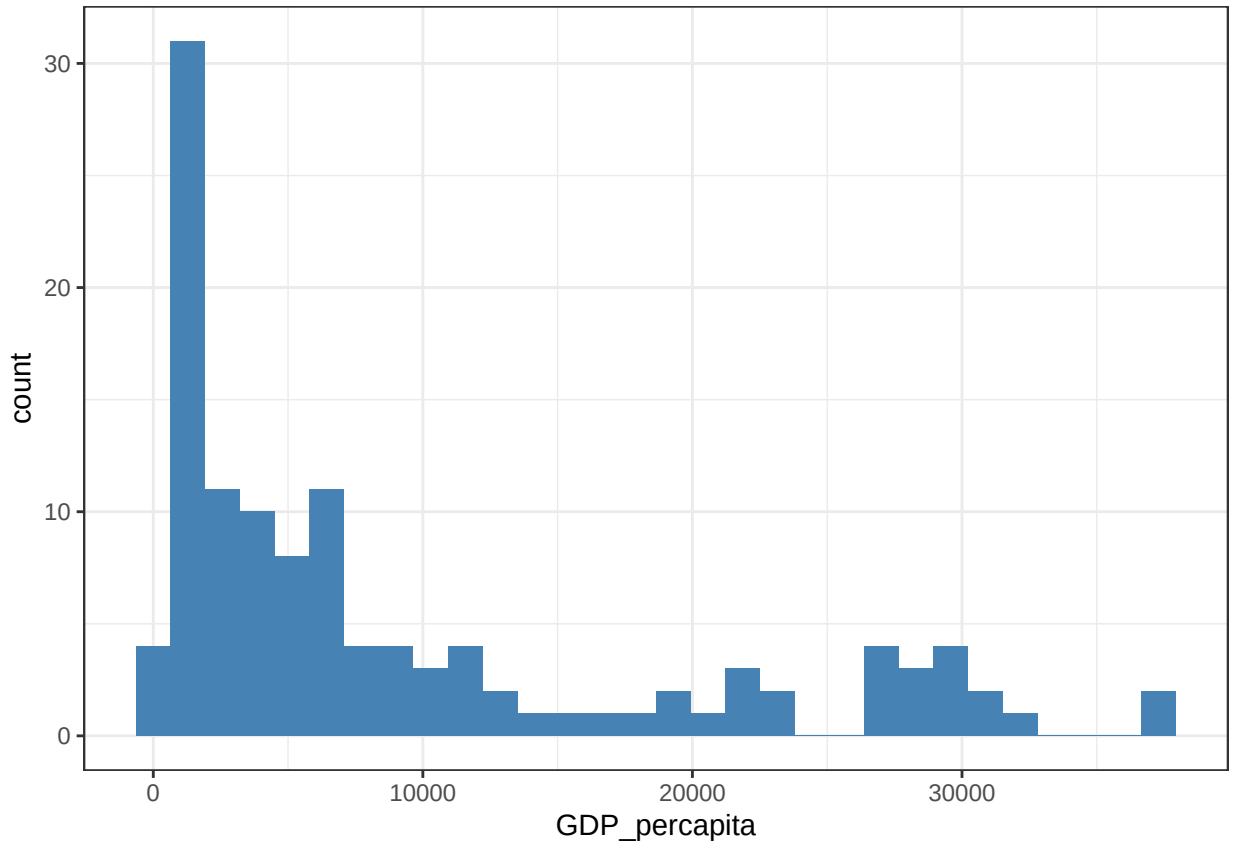
```
#Normality of Region
ggplot(happy_country, aes(x = Region)) +
geom_histogram(fill = "steelblue", stat="count") +
theme_bw()
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

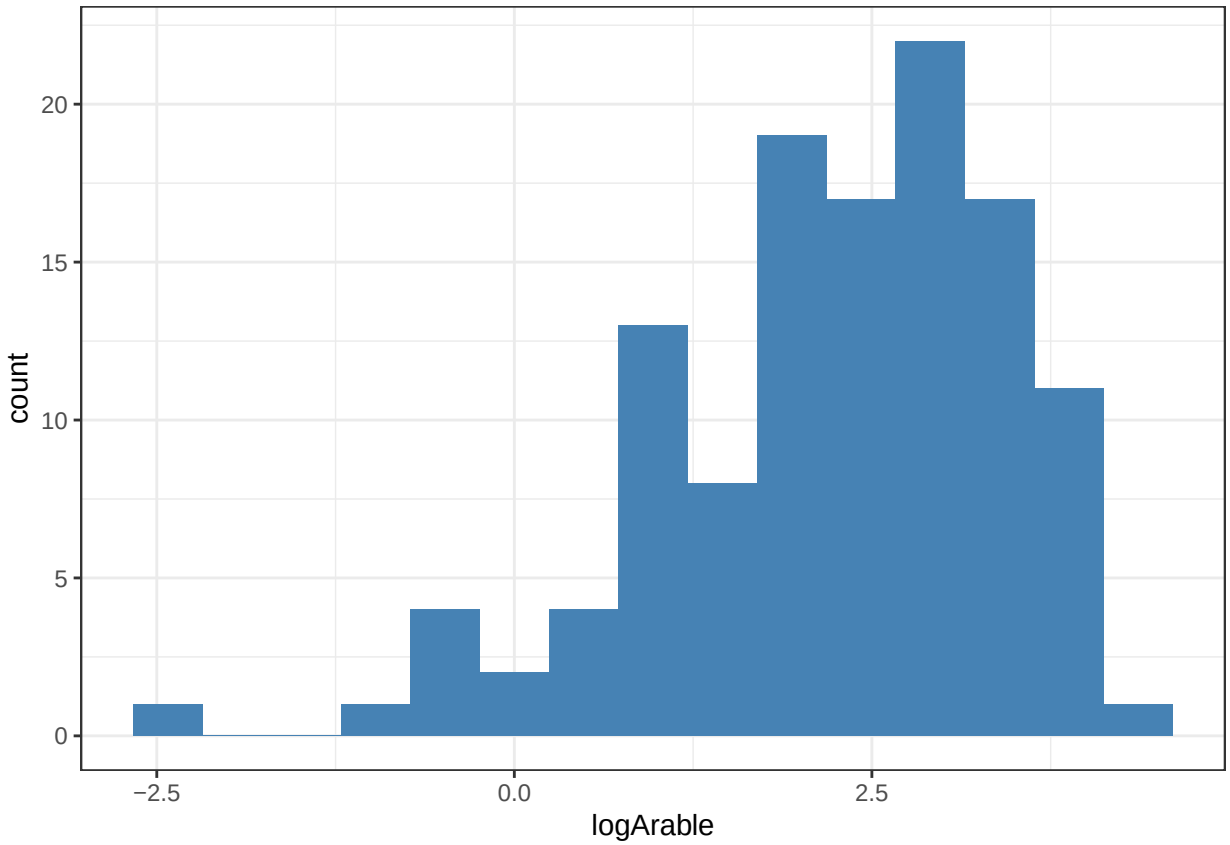



```
#Normality of GDP per capita
logGDP <- log(happy_country$GDP_percapita)
ggplot(happy_country, aes(x = GDP_percapita)) +
geom_histogram(fill = "steelblue") +
theme_bw()
```

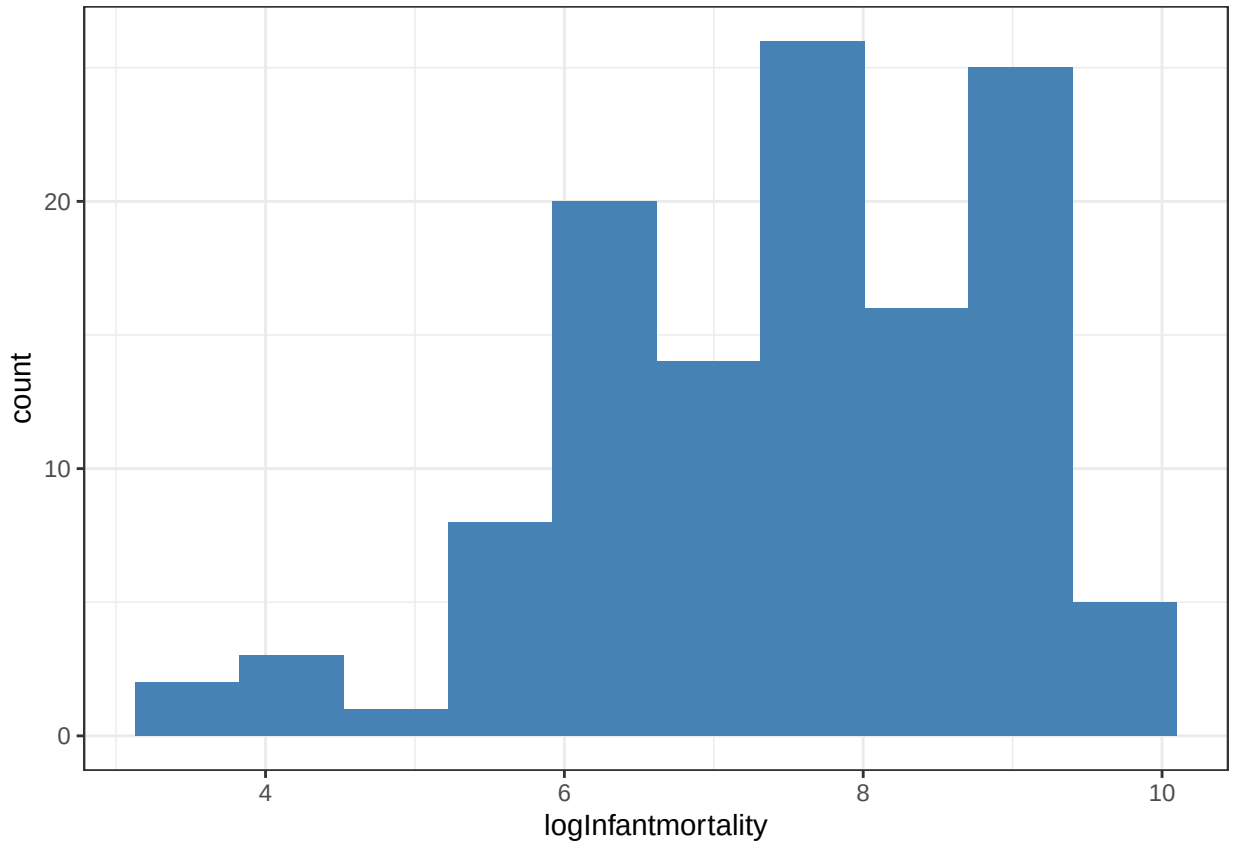
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



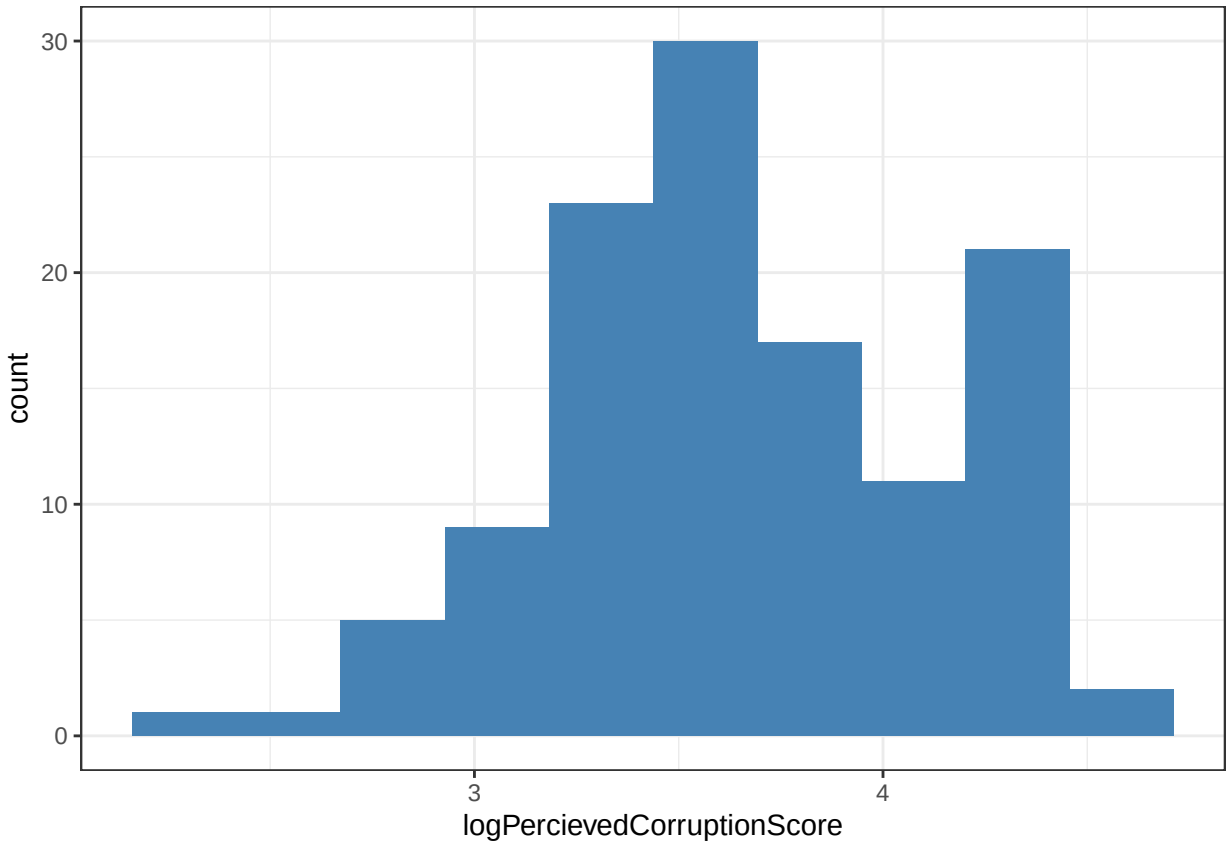

```
#Normality of Arable
logArable <- log(happy_country$Arable)
ggplot(happy_country, aes(x = logArable)) +
geom_histogram(fill = "steelblue", bins = "15") +
theme_bw()
```




```
#Normality of Infantmortalityper1000births
logInfantmortality <- log(happy_country$Infantmortalityper1000births)
ggplot(happy_country, aes(x = logInfantmortality)) +
geom_histogram(fill = "steelblue", bins = "10") +
theme_bw()
```

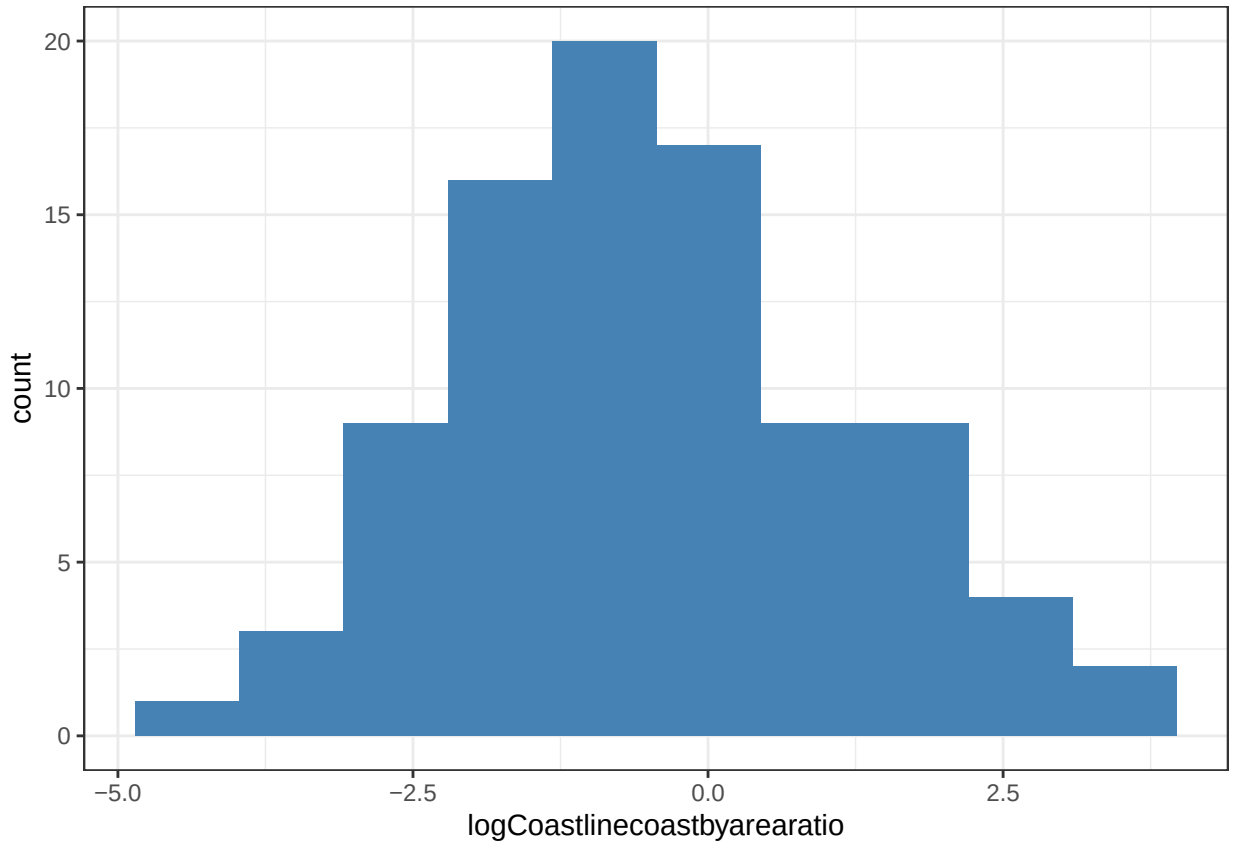



```
#Normality of PercievedCorruptionScore
logPercievedCorruptionScore <- log(happy_country$PercievedCorruptionScore)
ggplot(happy_country, aes(x = logPercievedCorruptionScore)) +
geom_histogram(fill = "steelblue", bins = "10") +
theme_bw()
```




```
#Normality of Coastlinecoastbyarearatio
logCoastlinecoastbyarearatio <- log(happy_country$Coastlinecoastbyarearatio)
lnCoastlinecoastbyarearatio <- log1p(happy_country$Coastlinecoastbyarearatio)
ggplot(happy_country, aes(x = logCoastlinecoastbyarearatio)) +
geom_histogram(fill = "steelblue", bins = "10") +
theme_bw()
```

```
## Warning: Removed 30 rows containing non-finite values (stat_bin).
```




```
#Region + GDP + log Arable Land + log Infant Mortality + log Percieved Corruption + Coastline Area
alymod7 <- lm(HappinessScore ~ Region + GDP_percapita +
                log(Arable) + log(Infantmortalityper1000births) +
                log(PercievedCorruptionScore) + Coastlinecoastbyarearatio,
              data = happy_country)
summary(alymod7)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Region + GDP_percapita + log(Arable) +
##     log(Infantmortalityper1000births) + log(PercievedCorruptionScore) +
##     Coastlinecoastbyarearatio, data = happy_country)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.56049 -0.34415  0.00454  0.40700  1.22812
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       4.597e+00  7.920e-01   5.805 7.05e-08
## RegionBALTICS                     3.770e-02  4.391e-01   0.086  0.93174
## RegionC.W. OF IND. STATES         2.424e-01  2.341e-01   1.036  0.30280
## RegionEASTERN EUROPE              7.437e-03  2.914e-01   0.026  0.97969
## RegionLATIN AMER. & CARIB         7.899e-01  1.926e-01   4.101 8.19e-05
## RegionNEAR EAST                  -1.044e-01  2.403e-01  -0.435  0.66480
## RegionNORTHERN AFRICA            -1.421e-01  3.702e-01  -0.384  0.70180
```

```
## RegionNORTHERN AMERICA            -1.225e+00  7.398e-01  -1.655  0.10088
## RegionOCEANIA                      2.264e-01  4.881e-01   0.464  0.64366
## RegionSUB-SAHARAN AFRICA          -5.667e-01  1.852e-01  -3.060  0.00281
## RegionWESTERN EUROPE              -3.569e-01  3.406e-01  -1.048  0.29720
## GDP_percapita                      8.134e-05  1.592e-05   5.110 1.47e-06
## log(Arable)                       -7.420e-02  4.931e-02  -1.505  0.13541
## log(Infantmortalityper1000births) -9.304e-02  5.589e-02  -1.665  0.09902
## log(PercievedCorruptionScore)      2.679e-01  1.900e-01   1.410  0.16145
## Coastlinecoastbyarearatio         -1.654e-02  1.449e-02  -1.142  0.25603
##
## (Intercept)                       ***
## RegionBALTICS
## RegionC.W. OF IND. STATES
## RegionEASTERN EUROPE
## RegionLATIN AMER. & CARIB         ***
## RegionNEAR EAST
## RegionNORTHERN AFRICA
## RegionNORTHERN AMERICA
## RegionOCEANIA
## RegionSUB-SAHARAN AFRICA          **
## RegionWESTERN EUROPE
## GDP_percapita                     ***
## log(Arable)
## log(Infantmortalityper1000births) .
## log(PercievedCorruptionScore)
## Coastlinecoastbyarearatio
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5734 on 104 degrees of freedom
## Multiple R-squared:  0.7905, Adjusted R-squared:  0.7603
## F-statistic: 26.17 on 15 and 104 DF,  p-value: < 2.2e-16
```

```
set.seed(1)
trainS <- sample(1:nrow(happy_country), nrow(happy_country) * .75)
trainD <- happy_country[trainS, ]
testD <- happy_country[-trainS, ]
```

```
#Train 1
linearmodel1 <- lm(HappinessScore ~ Region + GDP_percapita + Arable +
              Infantmortalityper1000births + PercievedCorruptionScore +
                Coastlinecoastbyarearatio, data = trainD)
summary(linearmodel1)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Region + GDP_percapita + Arable +
##     Infantmortalityper1000births + PercievedCorruptionScore +
##     Coastlinecoastbyarearatio, data = trainD)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.3554 -0.3363  0.0000  0.3443  0.9978
```

```
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   5.396e+00  3.076e-01  17.542  < 2e-16 ***
## RegionBALTICS                -1.426e-01  4.264e-01  -0.334  0.73896
## RegionC.W. OF IND. STATES    -9.271e-02  2.749e-01  -0.337  0.73690
## RegionEASTERN EUROPE         -3.151e-01  3.309e-01  -0.952  0.34408
## RegionLATIN AMER. & CARIB     4.370e-01  2.233e-01   1.957  0.05409 .
## RegionNEAR EAST              -5.772e-01  2.722e-01  -2.120  0.03736 *
## RegionNORTHERN AFRICA        -8.057e-01  4.170e-01  -1.932  0.05720 .
## RegionNORTHERN AMERICA       -2.133e+00  7.649e-01  -2.789  0.00672 **
## RegionOCEANIA                 1.238e-01  6.004e-01   0.206  0.83714
## RegionSUB-SAHARAN AFRICA     -8.159e-01  2.259e-01  -3.612  0.00055 ***
## RegionWESTERN EUROPE         -1.051e+00  3.911e-01  -2.687  0.00891 **
## GDP_percapita                 1.128e-04  1.975e-05   5.710 2.21e-07 ***
## Arable                       -1.494e-02  5.369e-03  -2.782  0.00684 **
## Infantmortalityper1000births -4.997e-05  2.340e-05  -2.135  0.03604 *
## PercievedCorruptionScore     -3.142e-03  6.149e-03  -0.511  0.61093
## Coastlinecoastbyarearatio    -4.441e-02  1.549e-02  -2.866  0.00541 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5303 on 74 degrees of freedom
## Multiple R-squared:  0.8388, Adjusted R-squared:  0.8061
## F-statistic: 25.67 on 15 and 74 DF,  p-value: < 2.2e-16
```

```
#Test 1
predict(linearmodel1, newdata = testD)
```

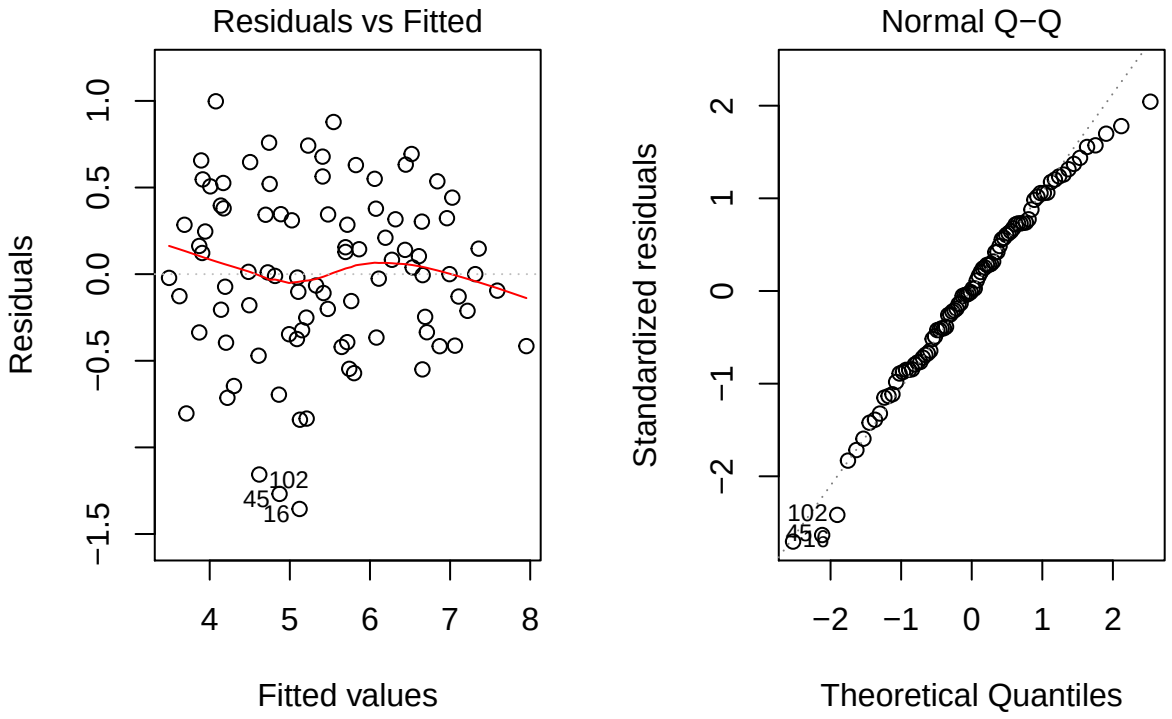
```
##        1        3        4        6       10       11       12       14
## 4.431964 5.112212 6.706248 8.412914 4.545656 5.345090 7.011217 4.784295
##       29       38       46       51       56       58       65       69
## 5.993547 4.798663 5.713753 5.746855 7.799920 5.651579 4.099085 4.330875
##       70       74       77       83       85       89       91       93
## 4.512688 4.964263 5.596258 6.030616 4.870401 4.974067 5.899259 3.695544
##       98      109      110      116      118      120
## 5.324847 3.930979 4.856577 6.249142 4.770795 4.262980
```

```
training_MSE <- mean(linearmodel1$residuals^2)
training_MSE
```

```
## [1] 0.2312365
```

```
#Residuals 1
par(mfrow = c(1, 2))
plot(linearmodel1, 1:2)
```

```
## Warning: not plotting observations with leverage one:
##   13, 68
```




```
#Train 2 with the Log variables
linearmodel2 <- lm(HappinessScore ~ Region + GDP_percapita +
                log(Arable) + log(Infantmortalityper1000births) +
                log(PercievedCorruptionScore) + Coastlinecoastbyarearatio,
                data = trainD)
summary(linearmodel2)
```

```
##
## Call:
## lm(formula = HappinessScore ~ Region + GDP_percapita + log(Arable) +
##     log(Infantmortalityper1000births) + log(PercievedCorruptionScore) +
##     Coastlinecoastbyarearatio, data = trainD)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.49193 -0.32924  0.00406  0.40521  0.94725
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       6.391e+00  9.261e-01   6.901 1.50e-09
## RegionBALTICS                    -1.245e-01  4.397e-01  -0.283 0.777799
## RegionC.W. OF IND. STATES         1.414e-02  2.806e-01   0.050 0.959945
## RegionEASTERN EUROPE             -4.768e-01  3.301e-01  -1.444 0.152882
## RegionLATIN AMER. & CARIB         5.613e-01  2.266e-01   2.476 0.015555
## RegionNEAR EAST                  -4.600e-01  2.827e-01  -1.627 0.107977
## RegionNORTHERN AFRICA            -6.235e-01  4.295e-01  -1.452 0.150781
## RegionNORTHERN AMERICA           -2.334e+00  8.054e-01  -2.898 0.004932
## RegionOCEANIA                     2.284e-01  6.164e-01   0.370 0.712079
## RegionSUB-SAHARAN AFRICA         -8.265e-01  2.228e-01  -3.710 0.000398
## RegionWESTERN EUROPE             -1.092e+00  4.177e-01  -2.613 0.010851
```

```
## GDP_percapita                       1.114e-04  1.951e-05   5.712 2.19e-07
## log(Arable)                        -8.075e-02  5.756e-02  -1.403 0.164823
## log(Infantmortalityper1000births) -1.268e-01  6.446e-02  -1.968 0.052831
## log(PercievedCorruptionScore)      -1.157e-01  2.146e-01  -0.539 0.591584
## Coastlinecoastbyarearatio          -4.371e-02  1.604e-02  -2.726 0.008007
##
## (Intercept)                        ***
## RegionBALTICS
## RegionC.W. OF IND. STATES
## RegionEASTERN EUROPE
## RegionLATIN AMER. & CARIB          *
## RegionNEAR EAST
## RegionNORTHERN AFRICA
## RegionNORTHERN AMERICA             **
## RegionOCEANIA
## RegionSUB-SAHARAN AFRICA           ***
## RegionWESTERN EUROPE               *
## GDP_percapita                      ***
## log(Arable)
## log(Infantmortalityper1000births) .
## log(PercievedCorruptionScore)
## Coastlinecoastbyarearatio          **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5492 on 74 degrees of freedom
## Multiple R-squared:  0.8272, Adjusted R-squared:  0.7921
## F-statistic: 23.61 on 15 and 74 DF,  p-value: < 2.2e-16
```

```
#Test 2
predict(linearmodel2, newdata = testD)
```

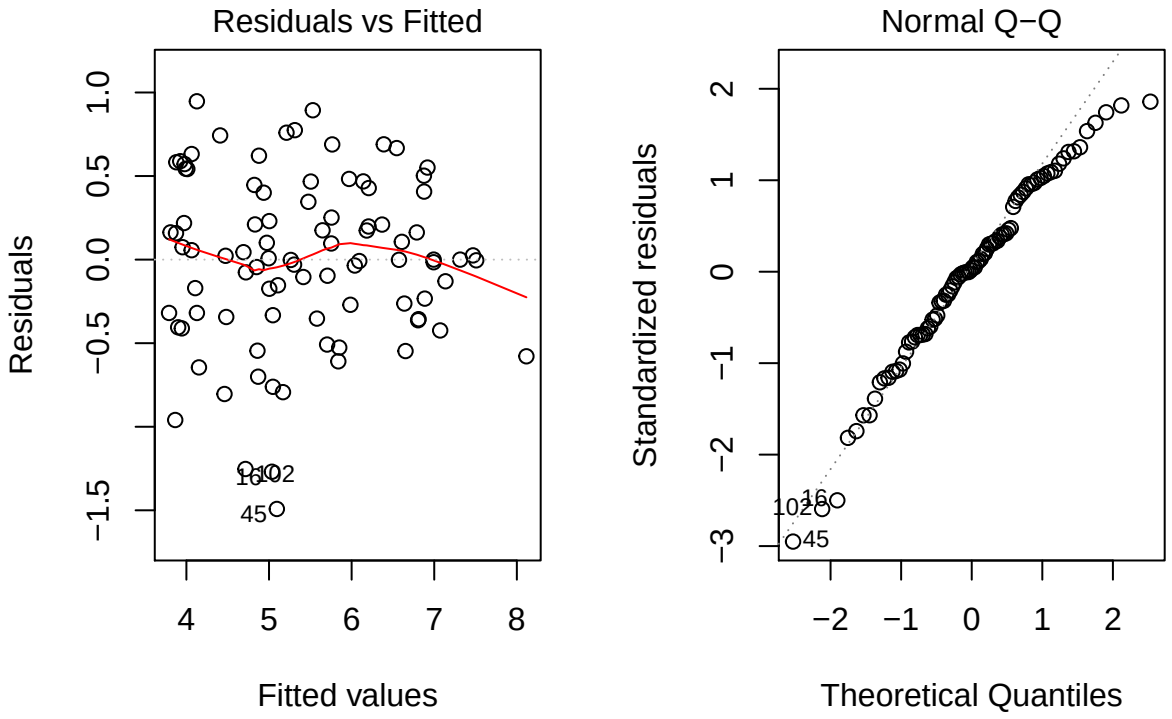
```
##        1        3        4        6       10       11       12       14
## 4.723765 5.499874 6.636875 8.402106 5.049476 5.460658 6.999434 4.789443
##       29       38       46       51       56       58       65       69
## 6.407044 4.654075 5.625601 5.539193 7.757600 5.530032 3.984103 4.311067
##       70       74       77       83       85       89       91       93
## 4.760414 4.826762 5.565651 5.927065 4.744554 4.943540 5.836949 3.831039
##       98      109      110      116      118      120
## 5.158978 3.961732 5.301684 6.366531 4.786467 4.128675
```

```
training_MSE2 <- mean(linearmodel2$residuals^2)
training_MSE2
```

```
## [1] 0.247965
```

```
#Residuals 2
par(mfrow = c(1, 2))
plot(linearmodel2, 1:2)
```

```
## Warning: not plotting observations with leverage one:
##   13, 68
```



Discussion

References

Arafa, S. (2019, April 5). Why Governments Should Care More about Happiness. Greater Good. https://greatergood.berkeley.edu/article/item/why_governments_should_care_more_about_happiness De Stasio, S., Fiorilli, C., Benevene, P., Boldrini, F., Ragni, B., Pepe, A., & Maldonado Briegas, J. J. (2019). Subjective Happiness and Compassion Are Enough to Increase Teachers' Work Engagement? Frontiers in Psychology, 10. https://doi.org/10.3389/fpsyg.2019.02268 e.V, T. I. (n.d.). Corruption Perceptions Index 2017. Retrieved December 4, 2019, from Www.transparency.org website Lasso, Fernando. "Countries of the World Data (World Factbook US Government)." Erasmus University, 26 Apr. 2018. "World Happiness Report (Gallup World Poll)." Sustainable Development Solutions Network Updates, 28 Feb. 2017.