Final Project Report

Global Life Expectancy Predictive Models

James Ross

STAT 5428

Introduction to Statistical Analysis

December 9, 2020

Introduction

The goal of this project was to build a statistical model that accurately predicts life expectancy based on several key factors. I was most interested at looking at this problem from a global perspective, and for that reason the models’ estimates were based on global life expectancy, not life expectancy per country. It is important to understand these factors and their influence so as to improve real life methods, and ideally have a positive impact on global health.

The data used for this report was gathered from the Kaggle database, which provides thousands of free datasets that are easily accessed. The observations were collected by the WHO (World Health Organization), a globally trusted organization. As such it is safe to assume that most of the data is accurate and not corrupted. For the purposes of my study, I removed a few variables from the data set that contained a significant amount of missing values or repetitive to another variable (several of the measures in the data set were describing the exact same thing).

The strategy used here was to construct three different models using different techniques. The techniques used were multiple linear regression, ridge regression and random forest regression, implementing R to build them.

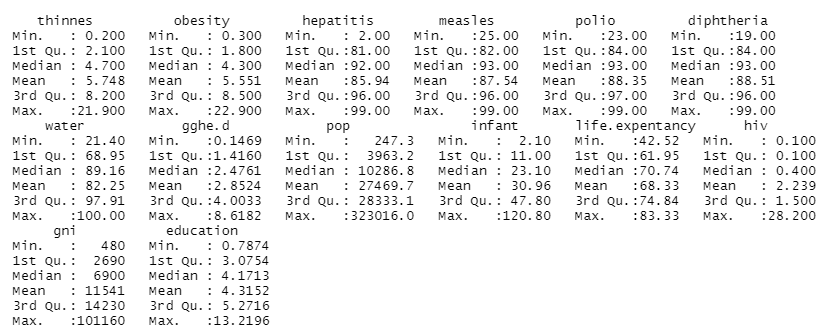
Data

The data set used had some 3,111 observations and 16 variables. It is important to understand the data used in this process. A large number of missing values can be difficult to deal with when fitting models, and can lead to misleading results. Therefore, the first step was to clean the data by filling in the missing values.

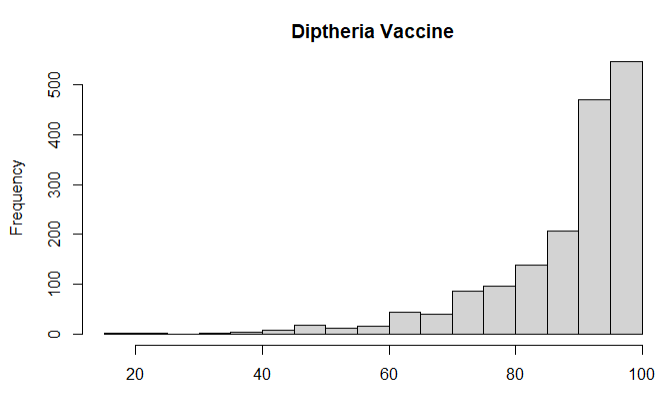
The approach used here was to replace them with their means based on feature’s respective country. This method replaces the missing values with more meaningful substitutes, than other methods. As some countries reported less, and had significantly different values, taking the mean of the entire variable and replacing missing values with that mean, would lead to inaccuracies and give more weight to the values of the countries that reported more often. Or give outliers more influence.

After the missing values for variables with large amounts of NAs were reasonably replaced, it was safe to remove the remaining rows with NA values while maintaining the integrity of the data. This left us with 1,685 observations.

Now that the working data set is ready for use, it is important to understand the behavior of the data. Here is an R output that gives the 5 number summary (mean, median, minimum value, maximum value, and Inner Quartile Range), for each variable:



The explanation for the variables names pictured above are as follows: Thinness and obesity were measures how many people per 100 fit into these categories as defined by the WHO. Hepatitis, measles and polio, and diptheria as the percentage of people that had received the respective vaccine. Access to clean drinking water, gghe.d represents percent of national GDP spent on health care, pop, population, infant stands for infant mortality rate, hiv is the prevalence of HIV, gni for Gross National Income per capita, and education is the percent of national GDP spent on education. As you can see several variables were skewed, the significant difference in mean and median suggest this. In order to visualize the distribution of each variable, I plotted histograms for each, all of which can be seen in the appendix. Here I will show one of the more skewed variables which was vaccine prevalence.



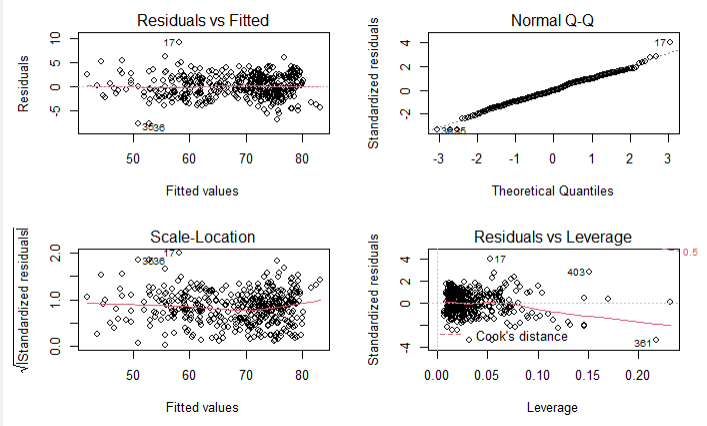
In fact, most of the distributions were slightly skewed. A normal distribution would be preferred, in attempt to achieve this a min-max transformation on the data was performed, this process scales each feature in the dataset, but it was ineffective. The next step was to begin building my models using the original working data set.

Methods

The techniques used were multiple regression, ridge regression, and random forest regression. Setting life expectancy as the target variable, and all but country and year as predictors. Country and year were not utilized because the goal was to predict global life expectancy, while the year the data was reported would not have much value, since the analysis was not in the time-series fashion.

To build these models data was split into training and testing subsets. In order to avoid bias every fourth row was chosen for the training set and the remainder was used for the test set. The reason for incrementing the selection this way was that the data was recorded based on country, so every 8 rows of the dataset related to one country, and random selection may have produced inaccurate results. The resulting training set contain 422 observations, and the testing set 1,263. The training set was used to build the model, and the test set would be used later on to test the predictive accuracy of the three models.

The first model fit was a multiple linear regression model. Implementing R following results were produced. The significant variables, based on respective p-values, were: hiv, infant mortality, healthcare spending, water quality, obesity and thinness. To understand the model looking at the coefficients and intercept is practical. According to the model the average global life expectancy is 69 years. HIV and infant mortality have a negative impact on life expectancy, while health care spending, water quality, thinness and obesity all have positive impacts. It is somewhat surprising to see that thinness and obesity factors have a positive effect, as you would rationally expect the opposite. That may suggest some bias in our model. To determine if our model met the assumptions for linear regression the diagnostic plots, as pictured below, were reviewed.



We can see from the Residuals vs fitted plot that there is a horizontal line without any pattern suggesting we meet the assumption of a linear relationship. From the Q-Q plot we can see that residuals closely follow the fitted line and are therefore normally distributed. From the scale-location plot we see that there is a slight curve and values seem to be more populated to the right, this may suggest that the assumption of homogeneity of variance has been violated. Finally looking at the Residuals vs Leverage plot, we notice a significant downward slope, which indicates that outliers may be effecting our model.

Based on the diagnostic plots, some tuning of the model would be necessary, in order to do this, the linear regression model was refit using only the variables that were deemed significant in the original model. This had a positive impact on the model as the diagnostic plots, pictured in the appendix, show. Both, the homogeneity of variance and independence of residuals assumptions appear to met. While it did not change the accuracy of the model much, or the coefficient values, the assumptions being met was enough to indicate this to be a better model than the original.

Next I used R to build a ridge regression model. Tuning the parameter lambda by cross-validation in order to attain an optimal value to be used in the final ridge regression model. While scaling the predictor variables using the center method. The ridge regression model did not provide much difference from the linear option. They significant variables were the same in both, and the R-squared values were basically identical, as well as the coefficients.

For the final model a random forest was fit to the data. By implementing the train function in the caret pack of R, our random forest model was fit using cross-validation. By averaging the results of a number of decision trees, this method produced the most accurate model.

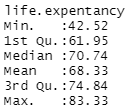
Conclusion

After optimizing the three models it was time to compare their predictive value using the test subset. The metric used to compare the models was RMSE, which measures, roughly, how much the predicted outcome differs from the actual outcome. In this case we used the models built from the training data and tested their predictive performance against the test data. The RMSEs were calculated using the R functions predict and RMSE. The predict function produces an outcome by using the data from the test set in the model. RMSE then takes these predicted values and compares them to the actual values. The results were as follows:

* Linear Regression RMSE: 2.334
* Ridge Regression RMSE: 2.302
* Random Forest Regression RMSE: 1.344

By this measure our random forest regression model produced the most accurate predictive value.

To understand what an RMSE value of 1.344, means in this case, it is useful to look at the range and inner quartile range of the target variable. Pictured here is the five number summary for life expectancy:



The life expectancy values range from 42 to 83, a difference of 41. And the inner quartile range is from 62 to 75, a 13 year difference. An RMSE of 1.344 is a indicates an accurate model, as there is such a broad range of life expectancies.

Perhaps the reason the models fit so well, is the importance of life expectancy, and therefore the data used in this report was likely already formatted well, and the variables chosen were already proven to affect life expectancy. However, whatever the cause, the value of understanding what factors effect the health of the globe cannot be overstated. It is useful to have an accurate prediction model to understand what can be done to improve people’s wellbeing.

**Appendix**

library(ggplot2)  
library(caret)

## Loading required package: lattice

library(dplyr)

##   
## Attaching package: 'dplyr'

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

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

library(data.table)

##   
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':  
##   
## between, first, last

library(countrycode)  
library(glmnet)

## Loading required package: Matrix

## Loaded glmnet 4.0-2

library(lmridge)  
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

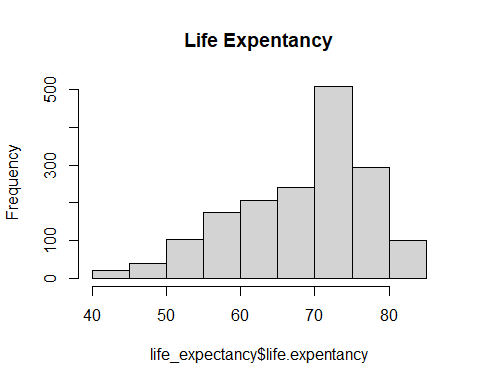
library(mlbench)  
library(e1071)  
  
  
setwd("C:\\Users\\jmros\\OneDrive\\Desktop\\Statistical Analysis")  
life\_expectancy <- read.csv("who\_life\_exp.csv", header = TRUE, stringsAsFactors = T)  
  
summary(life\_expectancy)

## country year thinnes obesity   
## AFG : 17 Min. :2000 Min. : 0.100 Min. : 0.100   
## AGO : 17 1st Qu.:2004 1st Qu.: 1.800 1st Qu.: 2.000   
## ALB : 17 Median :2008 Median : 3.800 Median : 5.200   
## ARE : 17 Mean :2008 Mean : 5.312 Mean : 5.972   
## ARG : 17 3rd Qu.:2012 3rd Qu.: 7.800 3rd Qu.: 8.900   
## ARM : 17 Max. :2016 Max. :28.100 Max. :26.700   
## (Other):3009 NA's :34 NA's :34   
## hepatitis measles polio diphtheria   
## Min. : 2.00 Min. :16.00 Min. : 8.00 Min. :19.00   
## 1st Qu.:81.00 1st Qu.:79.00 1st Qu.:81.00 1st Qu.:82.00   
## Median :92.00 Median :92.00 Median :93.00 Median :93.00   
## Mean :85.44 Mean :85.54 Mean :86.61 Mean :86.42   
## 3rd Qu.:97.00 3rd Qu.:96.00 3rd Qu.:97.00 3rd Qu.:97.00   
## Max. :99.00 Max. :99.00 Max. :99.00 Max. :99.00   
## NA's :569 NA's :19 NA's :19 NA's :19   
## water gghe.d pop infant   
## Min. : 18.70 Min. : 0.06236 Min. : 76 Min. : 1.60   
## 1st Qu.: 71.66 1st Qu.: 1.53344 1st Qu.: 2195 1st Qu.: 8.00   
## Median : 91.99 Median : 2.60130 Median : 8544 Median : 19.50   
## Mean : 83.33 Mean : 3.12293 Mean : 37076 Mean : 30.49   
## 3rd Qu.: 98.55 3rd Qu.: 4.27811 3rd Qu.: 25096 3rd Qu.: 48.05   
## Max. :100.00 Max. :12.06273 Max. :1414049 Max. :142.40   
## NA's :32 NA's :100 NA's :37   
## life.expentancy hiv gni education   
## Min. :39.44 Min. : 0.100 Min. : 420 Min. : 0.7874   
## 1st Qu.:62.84 1st Qu.: 0.100 1st Qu.: 2970 1st Qu.: 3.2628   
## Median :71.41 Median : 0.400 Median : 8340 Median : 4.4254   
## Mean :68.96 Mean : 2.038 Mean : 14965 Mean : 4.5329   
## 3rd Qu.:75.57 3rd Qu.: 1.500 3rd Qu.: 20483 3rd Qu.: 5.4950   
## Max. :83.98 Max. :28.200 Max. :122670 Max. :14.0591   
## NA's :741 NA's :117 NA's :1286

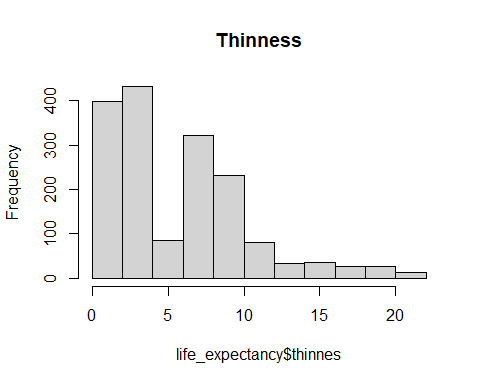
#get mean values based on country  
life\_expectancy <- life\_expectancy %>%  
 group\_by(country) %>%   
 mutate(education = replace(education, is.na(education), mean(education, na.rm = TRUE)))  
  
life\_expectancy <- life\_expectancy %>%  
 group\_by(country) %>%   
 mutate(hiv = replace(hiv, is.na(hiv), mean(hiv, na.rm = TRUE)))  
  
life\_expectancy <- life\_expectancy %>%  
 group\_by(country) %>%   
 mutate(gghe.d = replace(gghe.d, is.na(gghe.d), mean(gghe.d, na.rm = TRUE)))  
  
life\_expectancy <- life\_expectancy %>%  
 group\_by(country) %>%   
 mutate(gni = replace(gni, is.na(gni), mean(gni, na.rm = TRUE)))  
  
drops <- c("country", "year")  
life\_expectancy <- life\_expectancy[,!(names(life\_expectancy) %in% drops)]

life\_expectancy <- na.omit(life\_expectancy)

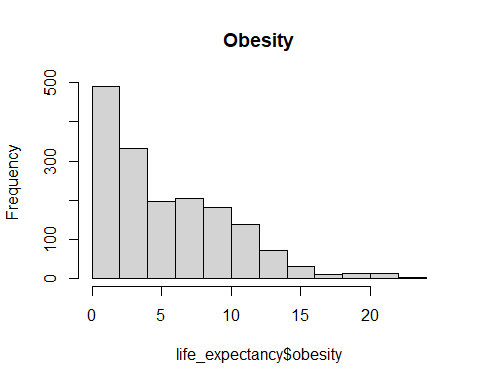
hist(life\_expectancy$life.expentancy, main = "Life Expentancy")



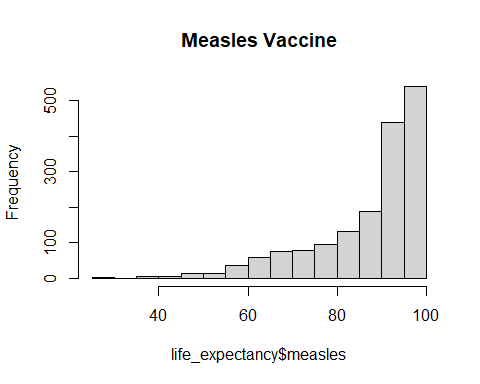
hist(life\_expectancy$thinnes, main = "Thinness")



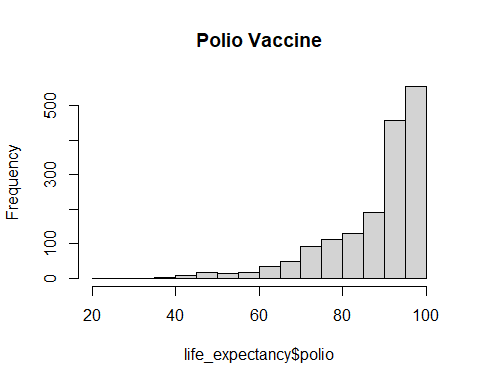
hist(life\_expectancy$obesity, main = "Obesity")



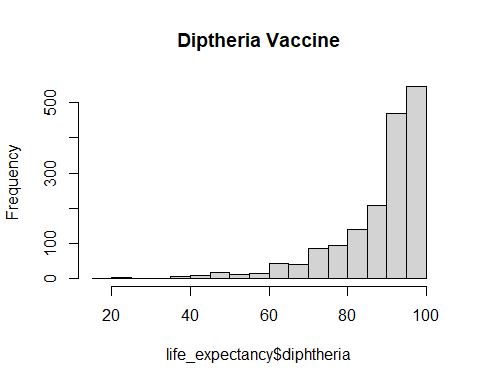
hist(life\_expectancy$measles, main = "Measles Vaccine")



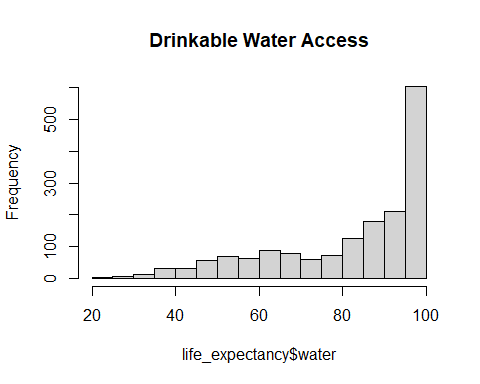
hist(life\_expectancy$polio, main = "Polio Vaccine")



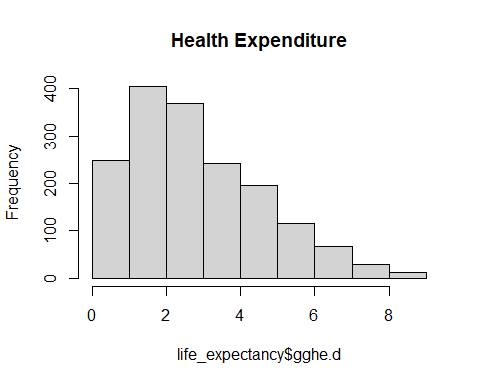
hist(life\_expectancy$diphtheria, main = "Diptheria Vaccine")



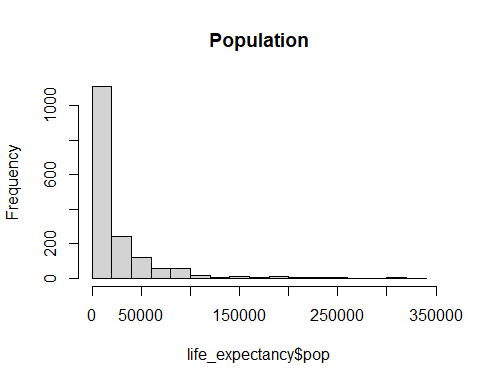
hist(life\_expectancy$water, main = "Drinkable Water Access")



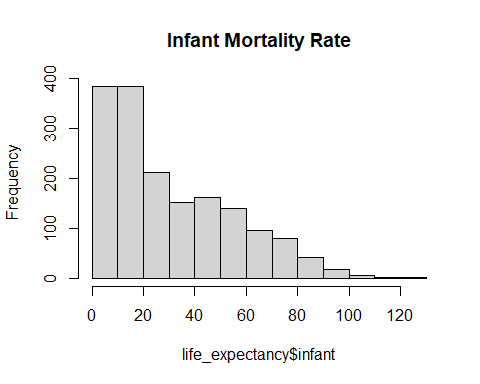
hist(life\_expectancy$gghe.d, main = "Health Expenditure")



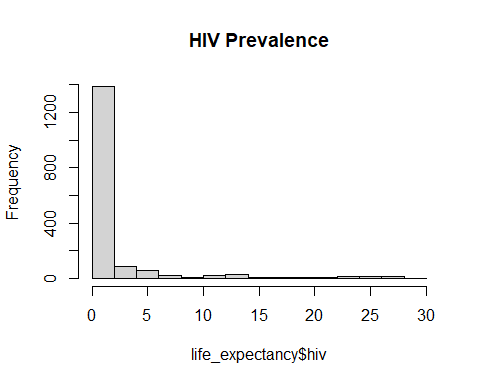
hist(life\_expectancy$pop, main = "Population")



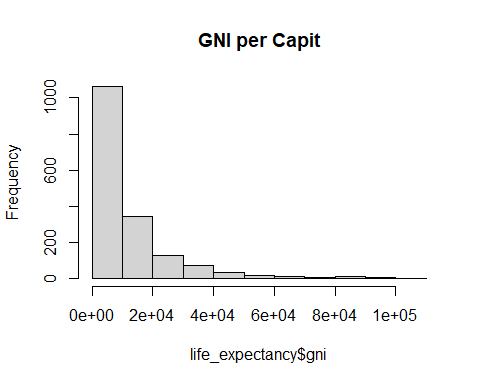
hist(life\_expectancy$infant, main = "Infant Mortality Rate")



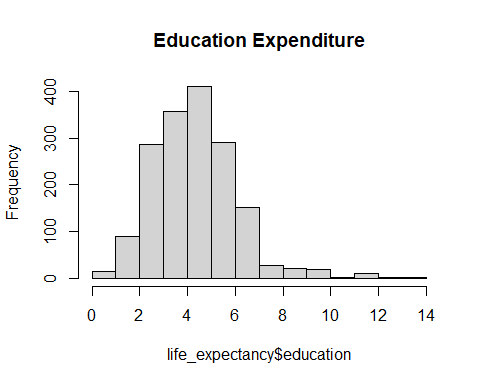
hist(life\_expectancy$hiv, main = "HIV Prevalence")



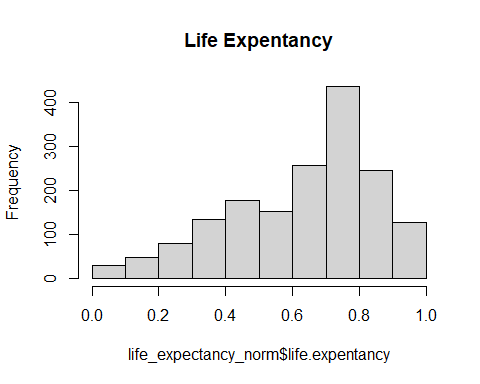
hist(life\_expectancy$gni, main = "GNI per Capit")



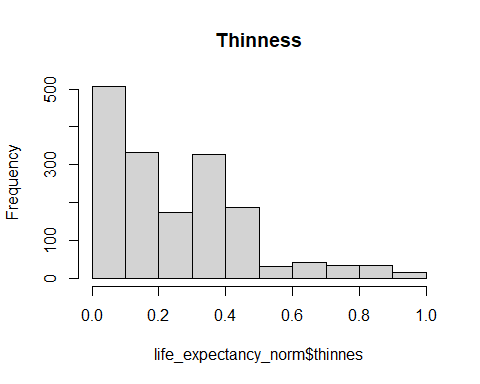
hist(life\_expectancy$education, main = "Education Expenditure")



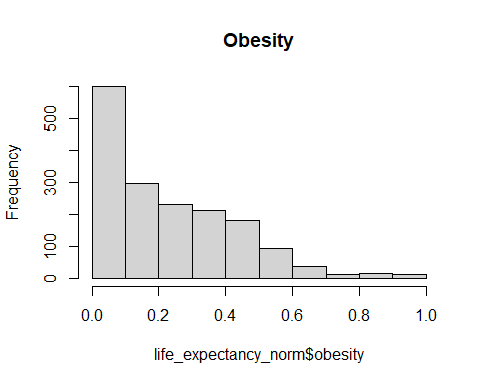
#Create the min-max normalization function  
min\_max\_norm <- function(x){  
 return((x-min(x))/(max(x)-min(x)))  
}   
  
#apply to the data  
  
life\_expectancy\_norm <- as.data.frame(lapply(life\_expectancy, min\_max\_norm))  
  
#next look at the plots  
  
hist(life\_expectancy\_norm$life.expentancy, main = "Life Expentancy")



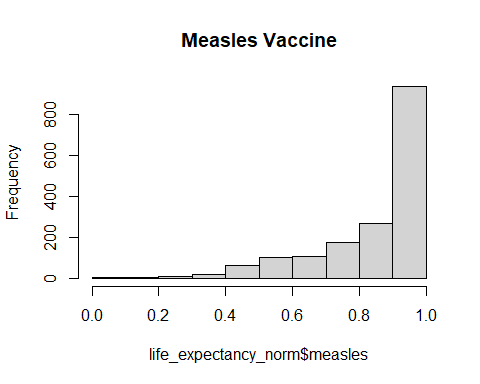
hist(life\_expectancy\_norm$thinnes, main = "Thinness")



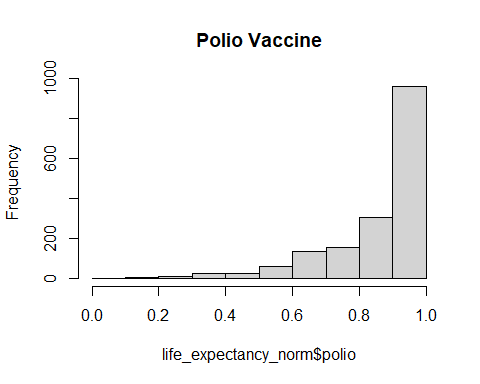
hist(life\_expectancy\_norm$obesity, main = "Obesity")



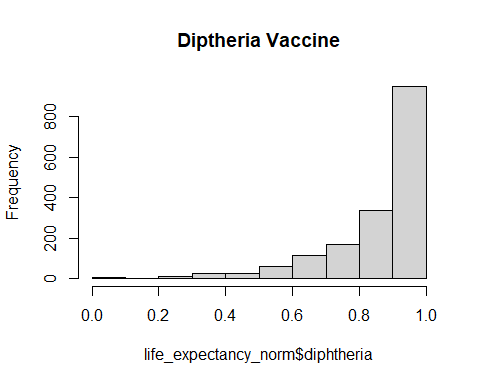
hist(life\_expectancy\_norm$measles, main = "Measles Vaccine")



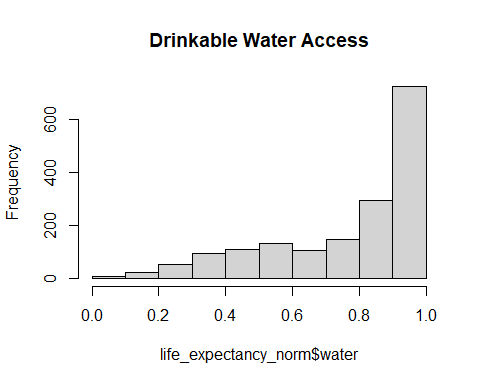
hist(life\_expectancy\_norm$polio, main = "Polio Vaccine")



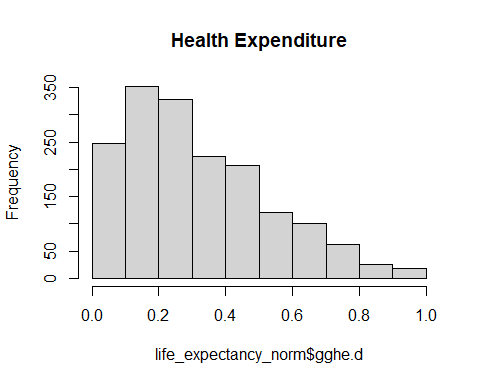
hist(life\_expectancy\_norm$diphtheria, main = "Diptheria Vaccine")



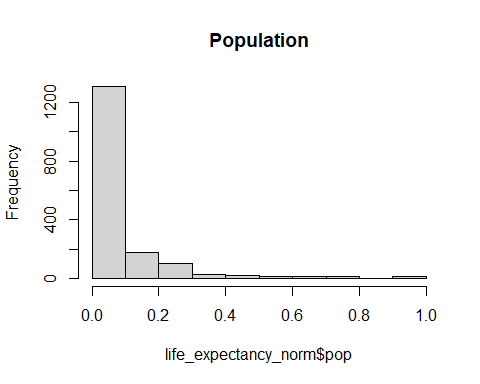
hist(life\_expectancy\_norm$water, main = "Drinkable Water Access")



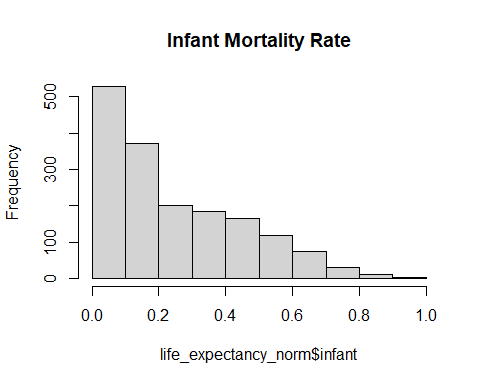
hist(life\_expectancy\_norm$gghe.d, main = "Health Expenditure")



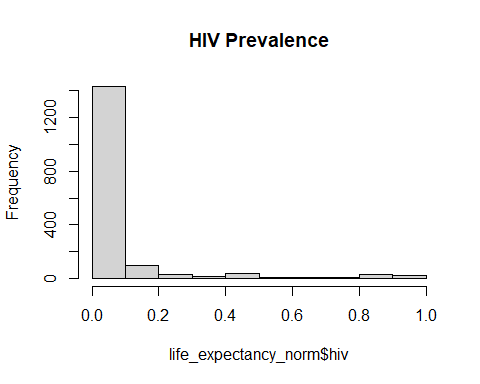
hist(life\_expectancy\_norm$pop, main = "Population")



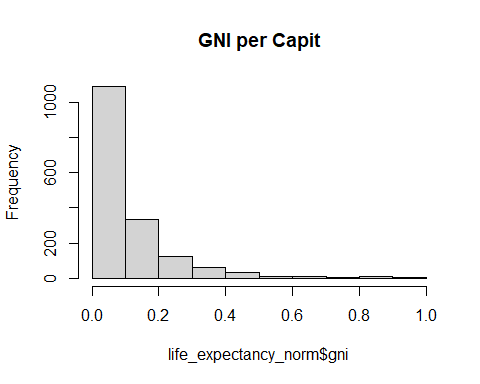
hist(life\_expectancy\_norm$infant, main = "Infant Mortality Rate")



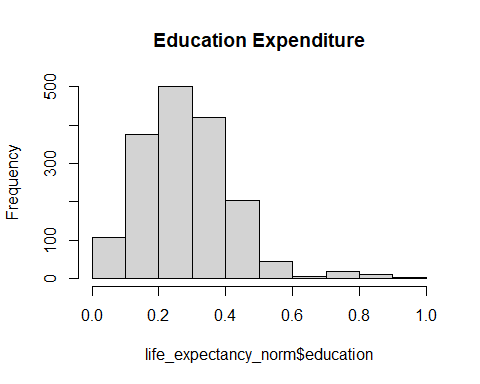
hist(life\_expectancy\_norm$hiv, main = "HIV Prevalence")



hist(life\_expectancy\_norm$gni, main = "GNI per Capit")



hist(life\_expectancy\_norm$education, main = "Education Expenditure")



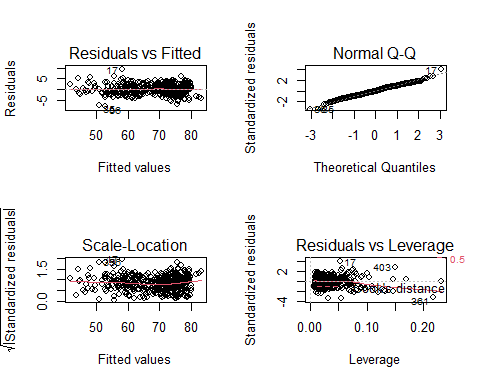
set.seed(1)  
train <- life\_expectancy[seq(1, nrow(life\_expectancy), 4),]  
test <- anti\_join(life\_expectancy, train)

## Joining, by = c("thinnes", "obesity", "hepatitis", "measles", "polio", "diphtheria", "water", "gghe.d", "pop", "infant", "life.expentancy", "hiv", "gni", "education")

mult\_reg <- lm(life.expentancy ~ ., train)  
summary(mult\_reg)

##   
## Call:  
## lm(formula = life.expentancy ~ ., data = train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -7.8205 -1.5863 -0.0732 1.6901 9.1772   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.920e+01 1.744e+00 39.673 < 2e-16 \*\*\*  
## thinnes 1.057e-01 3.925e-02 2.692 0.0074 \*\*   
## obesity 2.279e-01 4.424e-02 5.151 4.04e-07 \*\*\*  
## hepatitis -8.846e-03 1.257e-02 -0.704 0.4819   
## measles -3.937e-03 2.280e-02 -0.173 0.8630   
## polio -8.342e-03 3.968e-02 -0.210 0.8336   
## diphtheria 1.334e-02 4.147e-02 0.322 0.7479   
## water 4.969e-02 1.151e-02 4.317 1.99e-05 \*\*\*  
## gghe.d 5.145e-01 1.062e-01 4.846 1.79e-06 \*\*\*  
## pop 3.291e-06 2.719e-06 1.210 0.2269   
## infant -2.174e-01 1.094e-02 -19.867 < 2e-16 \*\*\*  
## hiv -5.504e-01 2.904e-02 -18.954 < 2e-16 \*\*\*  
## gni 1.144e-05 1.220e-05 0.937 0.3492   
## education 2.268e-02 8.393e-02 0.270 0.7871   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.365 on 408 degrees of freedom  
## Multiple R-squared: 0.9323, Adjusted R-squared: 0.9302   
## F-statistic: 432.4 on 13 and 408 DF, p-value: < 2.2e-16

par(mfrow = c(2,2))  
plot(mult\_reg)



mult\_reg\_prediction <- predict(mult\_reg, test)  
RMSE(test$life.expentancy, mult\_reg\_prediction)

## [1] 2.292882

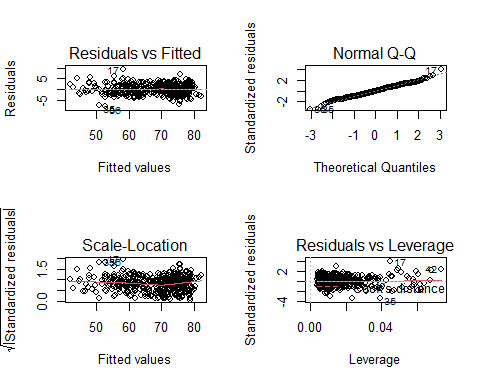
summary(life\_expectancy$life.expentancy)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 42.52 61.95 70.74 68.33 74.84 83.33

mult\_sig <- lm(life.expentancy ~ hiv + infant + gghe.d + water + thinnes + obesity, train)  
summary(mult\_sig)

##   
## Call:  
## lm(formula = life.expentancy ~ hiv + infant + gghe.d + water +   
## thinnes + obesity, data = train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -7.7985 -1.6860 -0.0424 1.6909 9.1731   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 68.350582 1.169256 58.456 < 2e-16 \*\*\*  
## hiv -0.554014 0.026005 -21.304 < 2e-16 \*\*\*  
## infant -0.215228 0.009795 -21.972 < 2e-16 \*\*\*  
## gghe.d 0.558369 0.096576 5.782 1.46e-08 \*\*\*  
## water 0.049841 0.011343 4.394 1.41e-05 \*\*\*  
## thinnes 0.129490 0.035583 3.639 0.000308 \*\*\*  
## obesity 0.258142 0.037888 6.813 3.37e-11 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.354 on 415 degrees of freedom  
## Multiple R-squared: 0.9318, Adjusted R-squared: 0.9308   
## F-statistic: 944.8 on 6 and 415 DF, p-value: < 2.2e-16

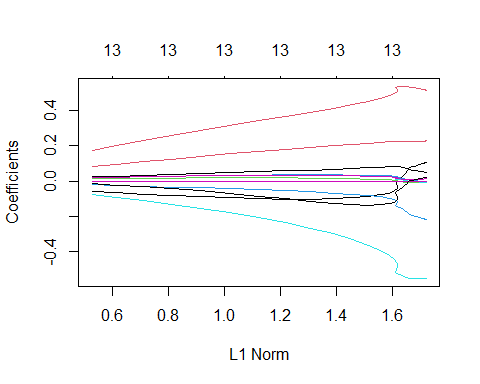
par(mfrow = c(2,2))  
plot(mult\_sig)



mult\_sig\_prediction <- predict(mult\_sig, test)  
RMSE(test$life.expentancy, mult\_sig\_prediction)

## [1] 2.302296

train\_matrix <- model.matrix(mult\_reg)  
  
lambdas <- 10^seq(2, -3, by = -.1) #create lambdas to be tested  
  
ridge\_reg <- glmnet(train\_matrix, train$life.expentancy, nlambda = 25, alpha = 0, family = "gaussian", lambda = lambdas)  
plot(ridge\_reg)



#now optimize the lambda value  
  
cv\_ridge <- cv.glmnet(train\_matrix, train$life.expentancy, alpha = 0, lambda = lambdas)  
optimal\_lambda <- cv\_ridge$lambda.min  
optimal\_lambda

## [1] 0.1258925

best\_ridge <- lmridge(life.expentancy ~., train, K = optimal\_lambda, scaling = "center")  
summary(best\_ridge)

##   
## Call:  
## lmridge.default(formula = life.expentancy ~ ., data = train,   
## K = optimal\_lambda, scaling = "center")  
##   
##   
## Coefficients: for Ridge parameter K= 0.1258925   
## Estimate Estimate (Sc) StdErr (Sc) t-value (Sc) Pr(>|t|)   
## Intercept 69.1964 69.1964 5.6833 12.1755 <2e-16 \*\*\*  
## thinnes 0.1056 0.1056 0.0392 2.6945 0.0073 \*\*   
## obesity 0.2279 0.2279 0.0442 5.1575 <2e-16 \*\*\*  
## hepatitis -0.0088 -0.0088 0.0126 -0.7047 0.4814   
## measles -0.0039 -0.0039 0.0228 -0.1730 0.8627   
## polio -0.0083 -0.0083 0.0396 -0.2105 0.8334   
## diphtheria 0.0133 0.0133 0.0414 0.3220 0.7476   
## water 0.0497 0.0497 0.0115 4.3225 <2e-16 \*\*\*  
## gghe.d 0.5143 0.5143 0.1060 4.8519 <2e-16 \*\*\*  
## pop 0.0000 0.0000 0.0000 1.2120 0.2262   
## infant -0.2174 -0.2174 0.0109 -19.8921 <2e-16 \*\*\*  
## hiv -0.5504 -0.5504 0.0290 -18.9778 <2e-16 \*\*\*  
## gni 0.0000 0.0000 0.0000 0.9388 0.3484   
## education 0.0227 0.0227 0.0838 0.2709 0.7866   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Ridge Summary  
## R2 adj-R2 DF ridge F AIC BIC   
## 0.93230 0.93030 12.99941 433.44680 738.16365 3341.73858   
## Ridge minimum MSE= 0.0268044 at K= 0.1258925   
## P-value for F-test ( 12.99941 , 409 ) = 8.777239e-230   
## -------------------------------------------------------------------

ridge\_prediction <- predict(best\_ridge, test)  
RMSE(test$life.expentancy, ridge\_prediction)

## [1] 2.301778

control <- trainControl(method = "repeatedcv", number = 10, repeats = 3)  
set.seed(2)  
mtry <- sqrt(ncol(train))  
tunegrid <- expand.grid(.mtry = mtry)  
rf\_default <- train(life.expentancy ~ .,   
 data = train,   
 method = "rf",   
 metric = "RMSE",   
 tuneGrid=tunegrid,  
 trControl = control)  
print(rf\_default)

## Random Forest   
##   
## 422 samples  
## 13 predictor  
##   
## No pre-processing  
## Resampling: Cross-Validated (10 fold, repeated 3 times)   
## Summary of sample sizes: 378, 382, 378, 381, 380, 378, ...   
## Resampling results:  
##   
## RMSE Rsquared MAE   
## 1.969315 0.9565674 1.465621  
##   
## Tuning parameter 'mtry' was held constant at a value of 3.741657

rf\_predict <- predict(rf\_default, newdata = test, type = "raw", norm.votes = T,   
 predict.all = FALSE, nodes = FALSE)  
RMSE(rf\_predict, test$life.expentancy)

## [1] 1.344838