Data Exploration and Analysis:
Multiple Regression with Life Expectancy Prediction

```
library(readr)
library(leaps)
library(corrplot)

library(tseries)
library(DescTools)
```

Loading and cleaning dataset

```
# Loading and cleaning data from source

col_names <- c('country', 'year', 'status', 'life_expectancy', 'adult_mortality',
               'infant_deaths', 'alcohol', 'percent_expend', 'hep_b', 'measles',
               'bmi', 'deaths_under5', 'polio', 'total_expend', 'diptheria', 'hiv_aids',
               'gdp', 'population', 'thin_1_19', 'thin_5_9', 'income_comp', 'schooling')

data <- read_csv('data/life_expectancy_raw.csv',
                 col_names = TRUE,
                 col_types = cols(population = col_double()),
                 trim_ws = TRUE)

colnames(data) <- col_names

head(data, 10) #view first 10 rows
```

```
## # A tibble: 10 x 22
##    country     year status   life_expectancy adult_mortality infant_deaths
##    <chr>      <int> <chr>              <dbl>           <int>         <int>
##  1 Afghanist~  2015 Develop~            65.0             263            62
##  2 Afghanist~  2014 Develop~            59.9             271            64
##  3 Afghanist~  2013 Develop~            59.9             268            66
##  4 Afghanist~  2012 Develop~            59.5             272            69
##  5 Afghanist~  2011 Develop~            59.2             275            71
##  6 Afghanist~  2010 Develop~            58.8             279            74
##  7 Afghanist~  2009 Develop~            58.6             281            77
##  8 Afghanist~  2008 Develop~            58.1             287            80
##  9 Afghanist~  2007 Develop~            57.5             295            82
## 10 Afghanist~  2006 Develop~            57.3             295            84
## # ... with 16 more variables: alcohol <dbl>, percent_expend <dbl>,
## #   hep_b <int>, measles <int>, bmi <dbl>, deaths_under5 <int>,
## #   polio <int>, total_expend <dbl>, diptheria <int>, hiv_aids <dbl>,
## #   gdp <dbl>, population <int>, thin_1_19 <dbl>, thin_5_9 <dbl>,
## #   income_comp <dbl>, schooling <dbl>
```

Data exploration and scoping

```
# Picking data from one year- try most recent year

data_2015 = data[data$year == 2015, ] # Most recent year available

missing_2015 <- colSums(is.na(data_2015))
print("Columns with more than 10% missing data for 2015:")
```

```
## [1] "Columns with more than 10% missing data for 2015:"
```

```
print(missing_2015[missing_2015 > 18])
```

```
##      alcohol total_expend          gdp   population
##          177          181           29           42
```

```
# Two predictors with data mostly incomplete- try the next most recent year

data_2014 = data[data$year == 2014, ] # Next most recent year
missing_2014 <- colSums(is.na(data_2014))
print("Columns with more than 10% missing data for 2014:")
```

```
## [1] "Columns with more than 10% missing data for 2014:"
```

```
print(missing_2014[missing_2014 > 18])
```

```
##        gdp population
##         28         42
```

```
# clean out 2014 data with complete columns to use as input for analysis

input <- data_2014[complete.cases(data_2014), ]


# convert country status to numerical dummy variable
input["status_code"] <- NA
input$status_code[input$status == 'Developed'] <- 1
input$status_code[input$status == 'Developing'] <- 0

# drop unused columns for regression model
input <- input[, !(colnames(input) %in% c('country', 'year', 'status'))]

head(input, 10) #view first 10 rows
```

```
## # A tibble: 10 x 20
##    life_expectancy adult_mortality infant_deaths alcohol percent_expend
##              <dbl>           <int>         <int>   <dbl>          <dbl>
##  1            59.9             271            64  0.0100           73.5
##  2            77.5               8             0  4.51            429.
##  3            75.4              11            21  0.0100           54.2
##  4            51.7             348            67  8.33             24.0
##  5            76.2             118             8  7.93            847.
##  6            74.6              12             1  3.91            296.
##  7            82.7               6             1  9.71          10769.
##  8            81.4              66             0 12.3            8350.
##  9            72.5             119             5  0.0100          306.
## 10            71.4             132            98  0.0100           10.4
## # ... with 15 more variables: hep_b <int>, measles <int>, bmi <dbl>,
## #   deaths_under5 <int>, polio <int>, total_expend <dbl>, diptheria <int>,
## #   hiv_aids <dbl>, gdp <dbl>, population <int>, thin_1_19 <dbl>,
## #   thin_5_9 <dbl>, income_comp <dbl>, schooling <dbl>, status_code <dbl>
```

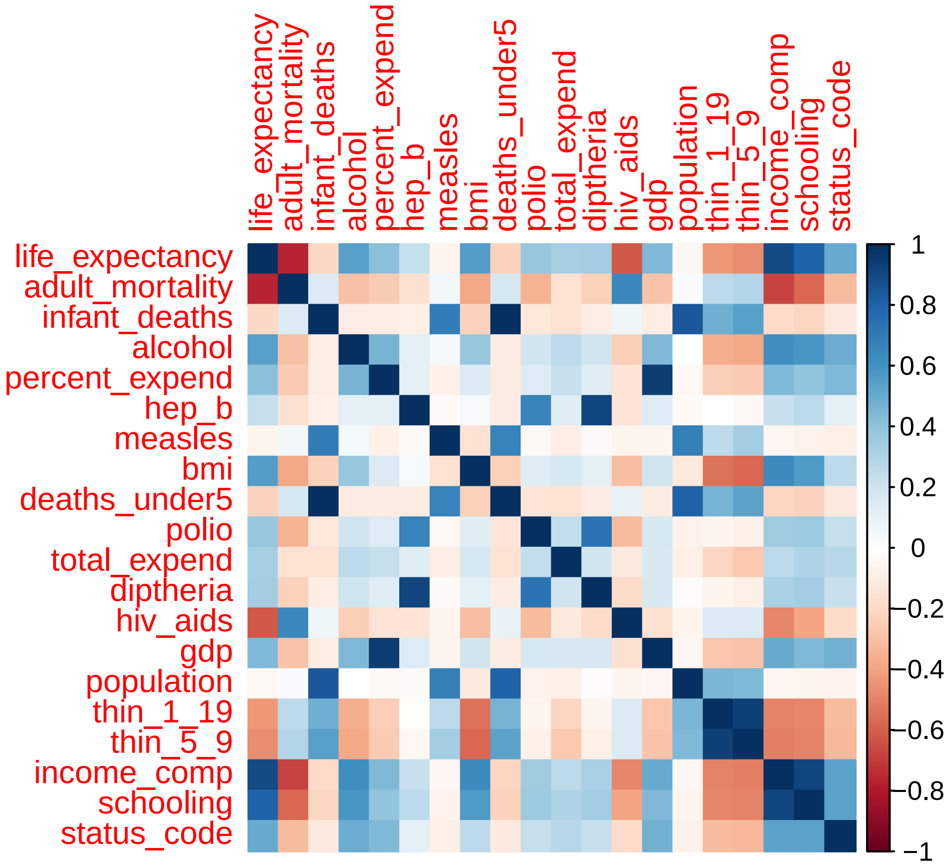
```
write.csv(input, file='data/life_expectancy_input.csv')
```

Check correlation of predictors

```
str(input)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    130 obs. of  20 variables:
##  $ life_expectancy: num  59.9 77.5 75.4 51.7 76.2 74.6 82.7 81.4 72.5 71.4 ...
##  $ adult_mortality: int  271 8 11 348 118 12 6 66 119 132 ...
##  $ infant_deaths  : int  64 0 21 67 8 1 1 0 5 98 ...
##  $ alcohol        : num  0.01 4.51 0.01 8.33 7.93 ...
##  $ percent_expend : num  73.5 428.7 54.2 24 847.4 ...
##  $ hep_b          : int  62 98 95 64 94 93 91 98 94 97 ...
##  $ measles        : int  492 0 0 11699 1 13 340 117 0 289 ...
##  $ bmi            : num  18.6 57.2 58.4 22.7 62.2 54.1 66.1 57.1 51.5 17.7 ...
##  $ deaths_under5  : int  86 1 24 101 9 1 1 0 6 121 ...
##  $ polio          : int  58 98 95 68 92 95 92 98 97 97 ...
##  $ total_expend   : num  8.18 5.88 7.21 3.31 4.79 ...
##  $ diptheria      : int  62 98 95 64 94 93 92 98 94 97 ...
##  $ hiv_aids       : num  0.1 0.1 0.1 2 0.1 0.1 0.1 0.1 0.1 0.1 ...
##  $ gdp            : num  613 4576 548 479 12245 ...
##  $ population     : int  327582 288914 39113313 2692466 42981515 29622 2346694 8541575 953579 1594527
##  $ thin_1_19      : num  17.5 1.2 6 8.5 1 2.1 0.6 1.8 2.8 18.1 ...
##  $ thin_5_9       : num  17.5 1.3 5.8 8.3 0.9 2.1 0.6 2 2.9 18.6 ...
##  $ income_comp    : num  0.476 0.761 0.741 0.527 0.825 0.739 0.936 0.892 0.752 0.57 ...
##  $ schooling      : num  10 14.2 14.4 11.4 17.3 12.7 20.4 15.9 12.2 10 ...
##  $ status_code    : num  0 0 0 0 0 0 1 1 0 0 ...
```

```
correlation <- cor(input)
corrplot(correlation, method = 'color')
```



Predictors with strong positive correlation with life_expectancy:

- income_comp
- schooling

Predictors with strong negative correlation with life_expectancy:

- adult_mortality
- hiv_aids

Stepwise selection using AIC

```
attach(input)

nullmodel <- lm(life_expectancy~1, data=input)
fullmodel <- lm(life_expectancy~., data=input)

step(nullmodel, data=input, scope=list(upper=fullmodel, lower=nullmodel, direction='both', k=2, test='F
```

```
##
## Call:
## lm(formula = life_expectancy ~ income_comp + adult_mortality +
##     hiv_aids + total_expend, data = input)
##
## Coefficients:
##     (Intercept)      income_comp  adult_mortality         hiv_aids
##        47.26321         36.59524         -0.01744         -0.81982
##    total_expend
##         0.36635
```

Reduced linear model after variable selection

After the stepwise selection, we choose to include the following four predictors:

- income_comp (income composition of resources index)
- adult_mortality (adult mortality probability)
- hiv_aids (death rate from HIV/AIDS)
- total_expend (government expenditure on health)

From this, we can build the regression model and perform analysis of variance:

```
model1 <- lm(life_expectancy ~ income_comp + adult_mortality + hiv_aids + total_expend)

anova(model1)
```

```
## Analysis of Variance Table
##
## Response: life_expectancy
##                  Df Sum Sq Mean Sq F value    Pr(>F)    
## income_comp       1 7651.5  7651.5 796.820 < 2.2e-16 ***
## adult_mortality   1  477.6   477.6  49.739 1.059e-10 ***
## hiv_aids          1  120.8   120.8  12.575  0.000551 ***
## total_expend      1  103.1   103.1  10.736  0.001360 ** 
## Residuals       125 1200.3     9.6                      
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(model1)
```

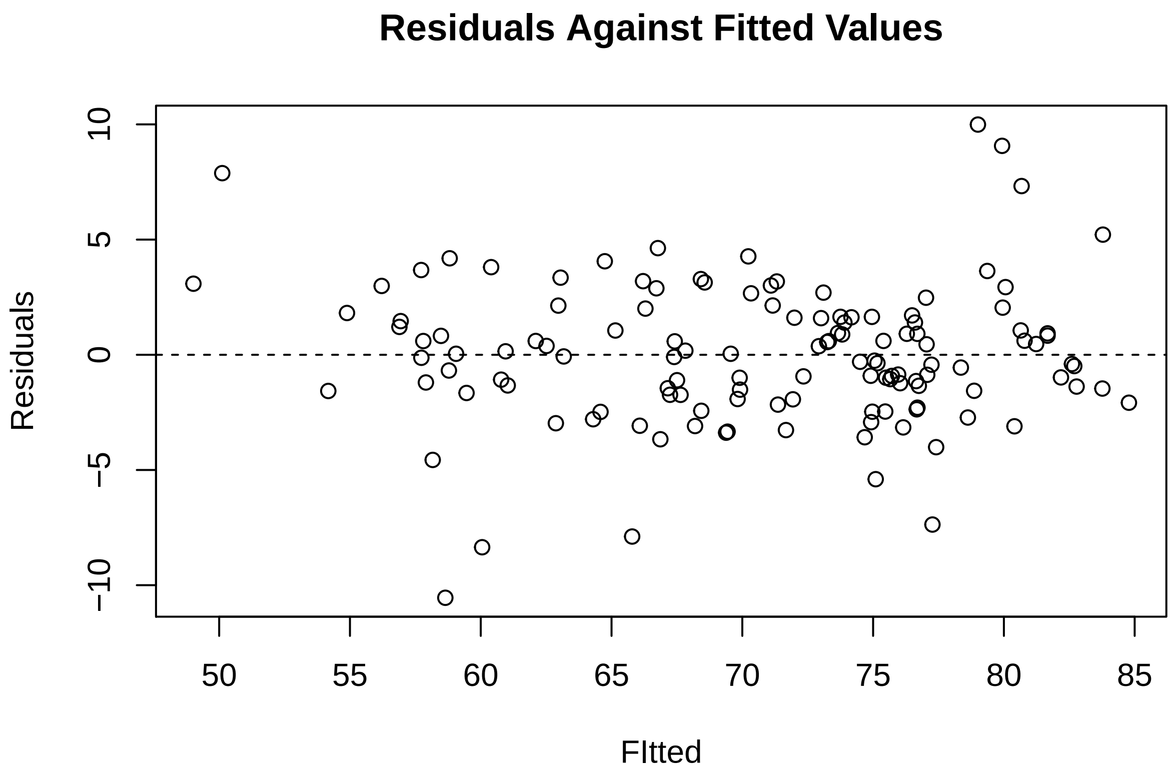
```
##
## Call:
## lm(formula = life_expectancy ~ income_comp + adult_mortality +
##     hiv_aids + total_expend)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -10.5461  -1.6357  -0.0831   1.6409   9.9919 
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)    
## (Intercept)     47.263214   2.070032  22.832  < 2e-16 ***
## income_comp     36.595239   2.501808  14.628  < 2e-16 ***
## adult_mortality -0.017439   0.003878  -4.497 1.56e-05 ***
## hiv_aids        -0.819817   0.230941  -3.550 0.000544 ***
## total_expend     0.366347   0.111809   3.277 0.001360 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.099 on 125 degrees of freedom
## Multiple R-squared:  0.8744, Adjusted R-squared:  0.8703 
## F-statistic: 217.5 on 4 and 125 DF,  p-value: < 2.2e-16
```

Testing linear model assumptions

Residual plot for constant variance of residuals:

```
plot(model1$fit, model1$res, xlab="FItted", ylab="Residuals", main="Residuals Against Fitted Values")

abline(h=0, lty=2)
```



Since there is no observable pattern within the residual plot, the assumption of constant variance is not violated.

Runs test and Durbin-Watson Test for independence of residuals:

```
res <- model1$res
runs.test(factor(sign(res)))
```

```
##
##  Runs Test
##
## data:  factor(sign(res))
## Standard Normal = 0.0027098, p-value = 0.9978
## alternative hypothesis: two.sided
```

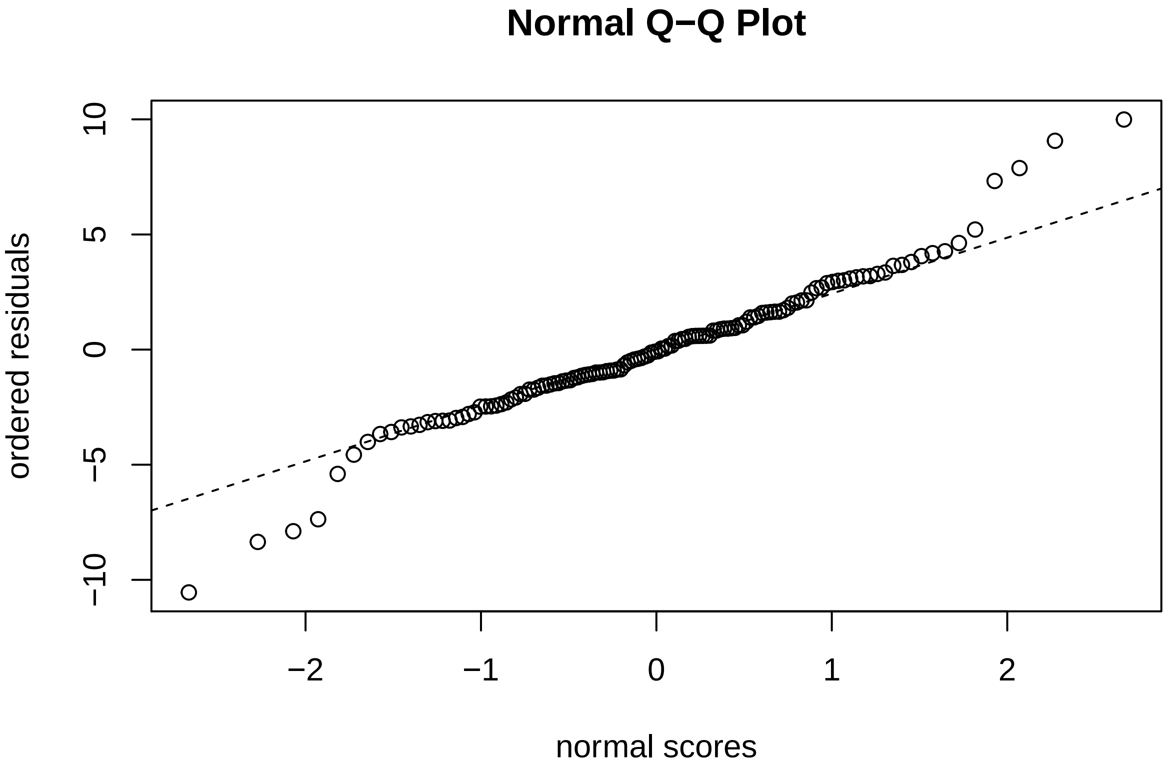
```
DurbinWatsonTest(model1, alternative="two.sided")
```

```
##
##  Durbin-Watson test
##
## data:  model1
## DW = 2.1762, p-value = 0.3096
## alternative hypothesis: true autocorrelation is not 0
```

Since both p-values are sufficiently large, there exists no significant evidence against the null hypothesis of both tests, which suggests that the residuals are not autocorrelated.

QQ plot for normal distribution of residuals:

```
qqnorm(res, xlab="normal scores", ylab="ordered residuals")
qqline(res, lty=2)
```



Since the normal probability plot is mostly close to the normal reference line, the assumption that residuals follow a normal distribution is not violated.