Life Expectancy Project

Sean ‘Cerulean’ Johnson

2020-12-15

Load Libraries

```
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.1.3
```

```
library(janitor)
```

```
## Warning: package 'janitor' was built under R version 4.1.3
```

```
library(readr)
```

```
## Warning: package 'readr' was built under R version 4.1.3
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.1.3
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.1.3
```

```
library(leaps)
```

```
## Warning: package 'leaps' was built under R version 4.1.3
```

Setting seed

```
seed = 1
set.seed(seed)
```

Load data into R

```
life_expec <- read_excel(here::here("data/life_expectancy.xlsx"))
```

Cleaning data

```
life_expec <-  clean_names(life_expec)
#head(life_expec)
```

Specific names to change

```
life_expec <- life_expec %>%
  rename(thin_10to19_years = thinness_1_19_years,
         thin_5to9_years = thinness_5_9_years,
         human_development_index = income_composition_of_resources
                          )
```

Removed the following variables as they do not add to analysis: infant_deaths, adult_mortality, and human_development_index.

```
life_expec <- life_expec %>% select(country,year,life_expectancy,population,alcohol,hiv_aids,thin_5to9_
```

Edited specific countries from undeveloped to developed as the data was inputed incorrectly

> This caused issues with data exploration, specifically when these countries were previously labeled asundeveloped caused outliers in boxplots.

```
#Canada
life_expec$status[497:512] <- "Developed"
#view(life_expec$status[497:512])

#Estonia
life_expec$status[865:880] <- "Developed"
#view(life_expec$status[865:880])

#Finland
life_expec$status[913:928] <- "Developed"
#view(life_expec$status[913:928])

#France
life_expec$status[929:944] <- "Developed"
#view(life_expec$status[929:944])

#Greece
life_expec$status[1025:1040] <- "Developed"
#view(life_expec$status[1025:1040])
```

See if there are any missing values

```
#total
sum(is.na(life_expec))
```

```
## [1] 1516
```

```
#variables with missing values
# 0 NA in sum(is.na(life_expec$country))
# 0 NA in sum(is.na(life_expec$year))
# 0 NA in sum(is.na(life_expec$life_expectancy))
# 624 NA in
sum(is.na(life_expec$population))
```

```
## [1] 624
```

```
# 30 NA in
sum(is.na(life_expec$alcohol))
```

```
## [1] 30
```

```
# 0 NA in sum(is.na(life_expec$hiv_aids))
# 32 NA in
sum(is.na(life_expec$thin_5to9_years))
```

```
## [1] 32
```

```
# 32 NA in
sum(is.na(life_expec$thin_10to19_years))
```

```
## [1] 32
```

```
# 78 NA in
sum(is.na(life_expec$hepatitis_b))
```

```
## [1] 78
```

```
# 0 NA in sum(is.na(life_expec$measles))
# 11 NA in
sum(is.na(life_expec$polio))
```

```
## [1] 11
```

```
# 11 NA in
sum(is.na(life_expec$diphtheria))
```

```
## [1] 11
```

```
# 32 NA in
sum(is.na(life_expec$bmi))
```

```
## [1] 32
```

```
# 0 NA in sum(is.na(life_expec$under_five_deaths))
# 65 NA in
sum(is.na(life_expec$total_expenditure))
```

```
## [1] 65
```

```
# 441 NA in
sum(is.na(life_expec$gdp))
```

```
## [1] 441
```

```
# 0 NA in sum(is.na(life_expec$percentage_expenditure))
# 160 NA in
sum(is.na(life_expec$schooling))
```

```
## [1] 160
```

```
# 0 NA in sum(is.na(life_expec$status))
```

> the largest variable with missing values is population followed by gdp.

Omitting all NA values from data to be able to run analysis

> Filling the na values with 0 does not make sense to me as this will throw the analysis by adding weigh to the point 0.
>
> Removing all rows that contain a NA. There are 1516 values missing of a total 55632 (19*2928) which is about 2.7% of total values.

```
le_adj <- na.omit(life_expec)
#dbl check
#sum(is.na(le_adj))
```

Checking structure of variables

```
summary(le_adj)
```

```
##    country               year      life_expectancy   population
##  Length:2256        Min.   :2000   Min.   :36.30   Min.   :3.400e+01
##  Class :character   1st Qu.:2004   1st Qu.:62.58   1st Qu.:1.930e+05
##  Mode  :character   Median :2008   Median :71.50   Median :1.351e+06
##                     Mean   :2008   Mean   :68.85   Mean   :1.276e+07
##                     3rd Qu.:2011   3rd Qu.:75.50   3rd Qu.:7.384e+06
##                      Max.   :2015   Max.   :89.00   Max.   :1.294e+09
##     alcohol          hiv_aids      thin_5to9_years  thin_10to19_years
##  Min.   : 0.000   Min.   : 0.100   Min.   : 0.100   Min.   : 0.100
##  1st Qu.: 0.680   1st Qu.: 0.100   1st Qu.: 1.500   1st Qu.: 1.500
##  Median : 4.030   Median : 0.100   Median : 3.100   Median : 2.950
##  Mean   : 4.657   Mean   : 2.056   Mean   : 4.931   Mean   : 4.870
##  3rd Qu.: 7.600   3rd Qu.: 1.100   3rd Qu.: 7.400   3rd Qu.: 7.325
##  Max.   :17.870   Max.   :50.600   Max.   :28.600   Max.   :27.700
##   hepatitis_b       measles             polio         diphtheria
##  Min.   : 0.00   Min.   :     0.0   Min.   : 3.00   Min.   : 2.00
##  1st Qu.:64.00   1st Qu.:     0.0   1st Qu.:76.00   1st Qu.:78.00
##  Median :87.50   Median :    15.0   Median :92.00   Median :92.00
##  Mean   :74.41   Mean   :  2556.3   Mean   :81.59   Mean   :81.82
##  3rd Qu.:95.00   3rd Qu.:   412.2   3rd Qu.:97.00   3rd Qu.:97.00
##  Max.   :99.00   Max.   :212183.0   Max.   :99.00   Max.   :99.00
##       bmi        under_five_deaths total_expenditure      gdp
##  Min.   : 1.40   Min.   :   0.00   Min.   : 0.000    Min.   :     1.68
##  1st Qu.:18.70   1st Qu.:   1.00   1st Qu.: 4.400    1st Qu.:   438.52
##  Median :41.80   Median :   4.00   Median : 5.900    Median :  1550.55
##  Mean   :37.37   Mean   :  46.96   Mean   : 6.024    Mean   :  6682.92
##  3rd Qu.:55.70   3rd Qu.:  29.00   3rd Qu.: 7.692    3rd Qu.:  5291.74
##  Max.   :77.60   Max.   :2500.00   Max.   :14.390    Max.   :119172.74
##  percentage_expenditure   schooling       status
##  Min.   :    0.00       Min.   : 0.0   Length:2256
##  1st Qu.:   21.88       1st Qu.:10.0   Class :character
##  Median :  100.43       Median :12.2   Mode  :character
##  Mean   :  843.92       Mean   :12.0
##  3rd Qu.:  509.10       3rd Qu.:14.4
##  Max.   :19479.91       Max.   :20.7
```

Changing categorical variable status into a numeric

```
le_adj$stat_num <- as.numeric(factor(le_adj$status)) -1
```

> Devevloping is equal to 1 and developed is equal 0. This will be usefull for exploratory data analysis. Also, if there is time, I can use with decision tree analysis like randomForest.

Split dataframe by categorical

```
#ordering data set by status
le_adj <- le_adj[order(le_adj$status),]

le_developed <-le_adj[1:496,]
le_developing <-le_adj[497:2256,]

#return dataframe to alphabetical list by country
le_adj <- le_adj[order(le_adj$country),]
```

> This was done due to the variability of undeveloped nations.

EDA

```
glimpse(le_adj)
summary(le_adj)
```

```
#non vaccines or gov't
le_adj %>%
    group_by(status) %>%
    summarize(count = n(),
              avg_lifexp = mean(life_expectancy),
              avg_pop = mean(population),
              avg_alcohol = mean(alcohol),
              avg_hiv = mean(hiv_aids),
              avg_thinLessThan10 = mean(thin_5to9_years),
              avg_thin10plus = mean(thin_10to19_years),
              avg_bmi = mean(bmi),
              avg_under5 = mean(under_five_deaths))
```

```
## # A tibble: 2 x 10
##   status    count avg_l~1 avg_pop avg_a~2 avg_hiv avg_t~3 avg_t~4 avg_bmi avg_u~5
##   <chr>     <int>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 Develop~    496    79.4  7.17e6    9.81     0.1    1.27    1.29    52.6   0.931
## 2 Develop~   1760    65.9  1.43e7    3.21    2.61    5.96    5.88    33.1  59.9
## # ... with abbreviated variable names 1: avg_lifexp, 2: avg_alcohol,
## #   3: avg_thinLessThan10, 4: avg_thin10plus, 5: avg_under5
```

```
#vaccines
le_adj %>%
    group_by(status) %>%
    summarize(count = n(),
              avg_hep = mean(hepatitis_b),
              avg_meas = mean(measles),
              avg_ploio= mean(polio),
              avg_dipth = mean(diphtheria))
```

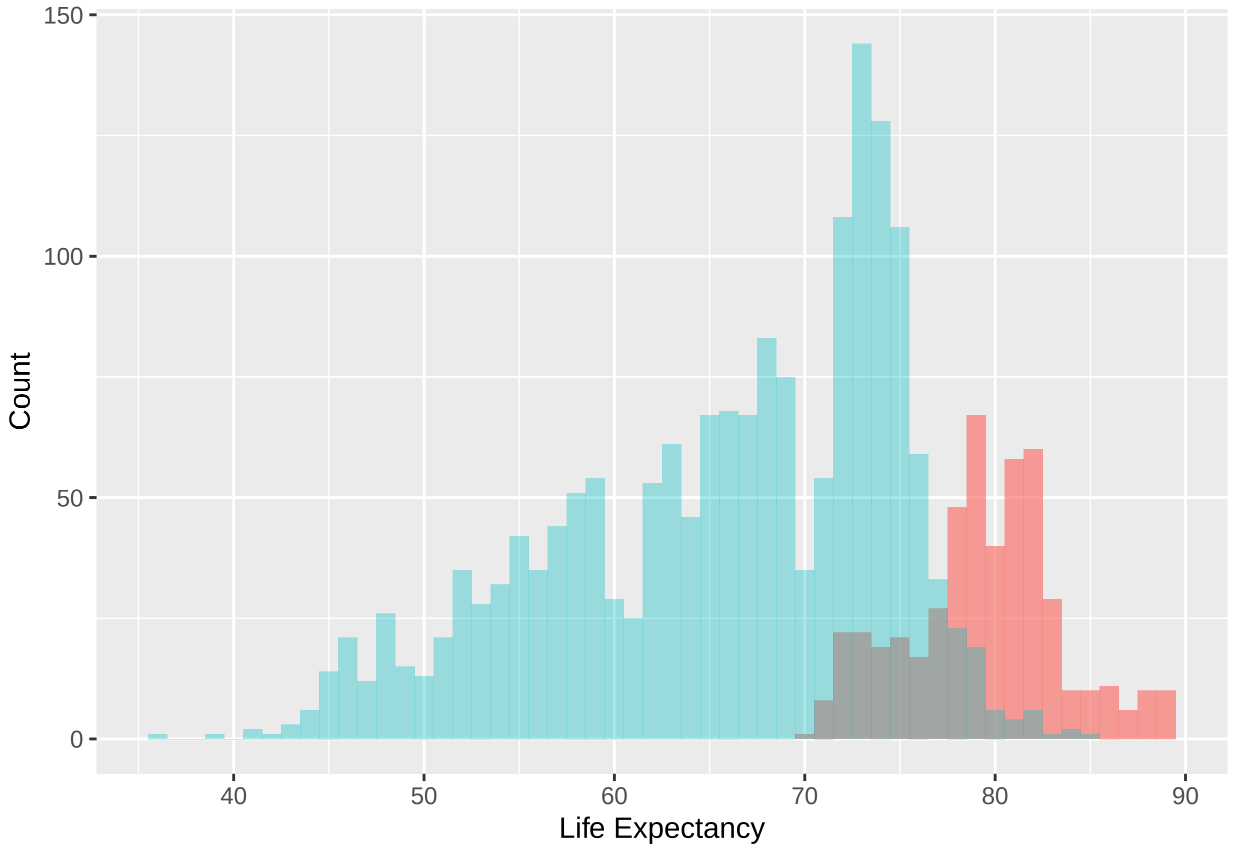
```
## # A tibble: 2 x 6
##   status     count avg_hep avg_meas avg_ploio avg_dipth
##   <chr>      <int>   <dbl>    <dbl>     <dbl>     <dbl>
## 1 Developed    496    70.6     572.      93.8      94.1
## 2 Developing  1760    75.5    3115.      78.1      78.3
```

```
#gov't input
le_adj %>%
    group_by(status) %>%
    summarize(count = n(),
              avg_totExp = mean(total_expenditure),
              avg_gdp = mean(gdp),
              avg_pctExp = mean(percentage_expenditure),
              avg_school = mean(schooling))
```

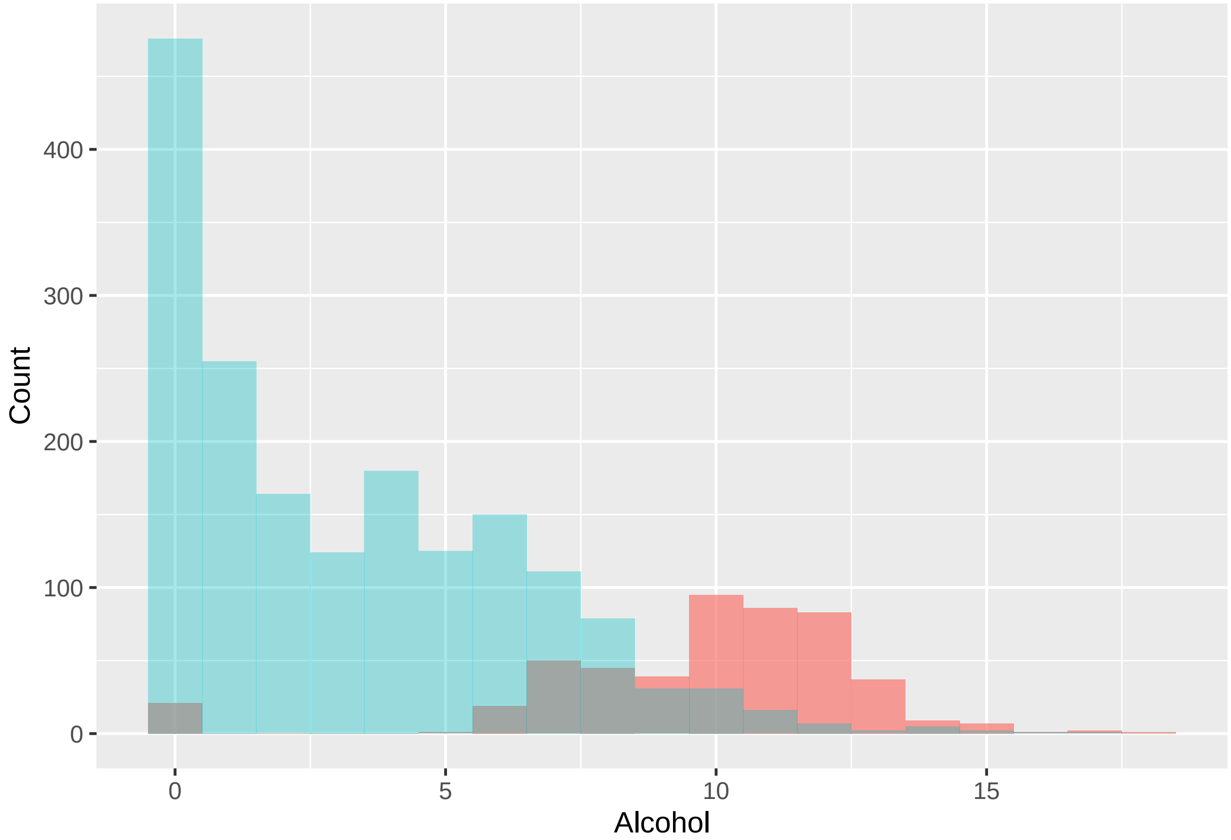
```
## # A tibble: 2 x 6
##   status     count avg_totExp avg_gdp avg_pctExp avg_school
##   <chr>      <int>      <dbl>   <dbl>      <dbl>      <dbl>
## 1 Developed    496       7.34  21801.      3072.       15.9
## 2 Developing  1760       5.65   2422.       216.       10.9
```

univariate eda

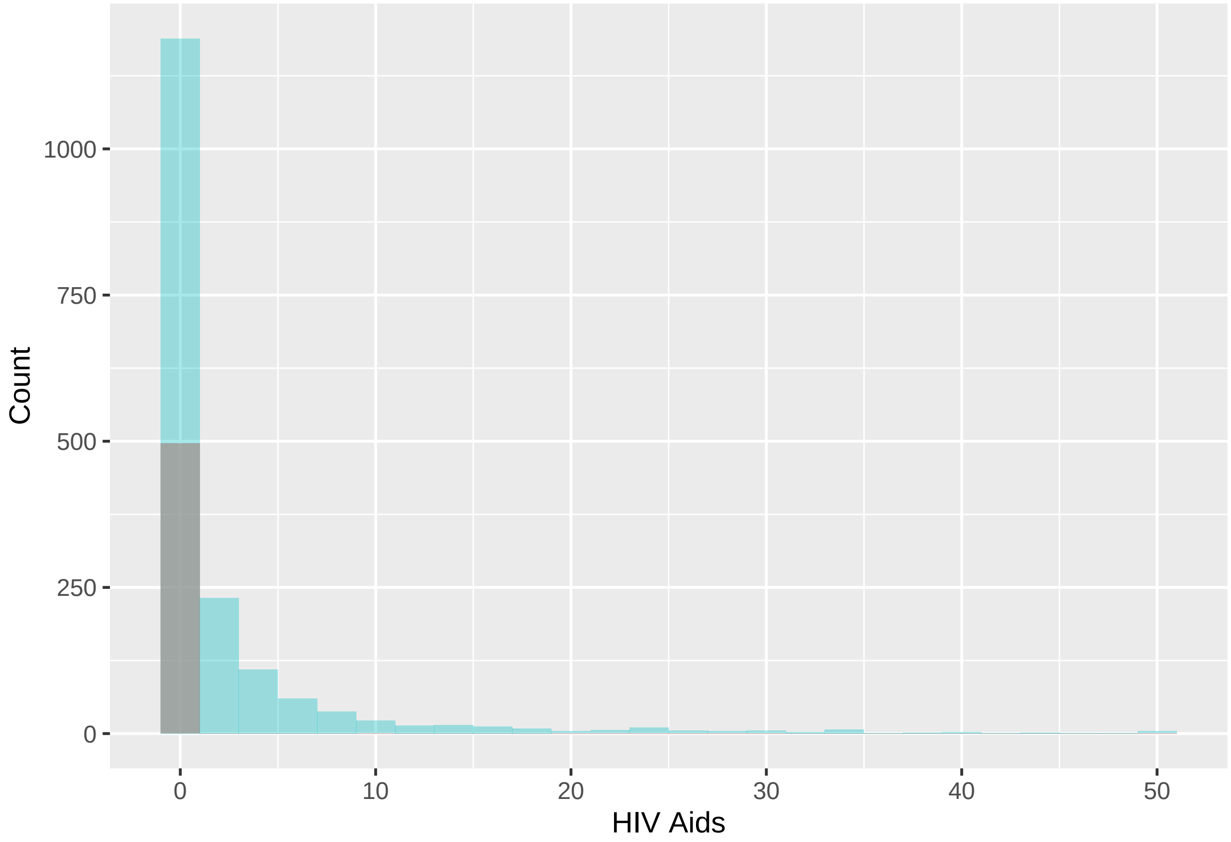
```
ggplot(data = le_adj, aes(x = life_expectancy))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 1, fill="#F8766D", alpha = .7)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 1, fill="#00BFC4", alpha = .35)+ labs
```



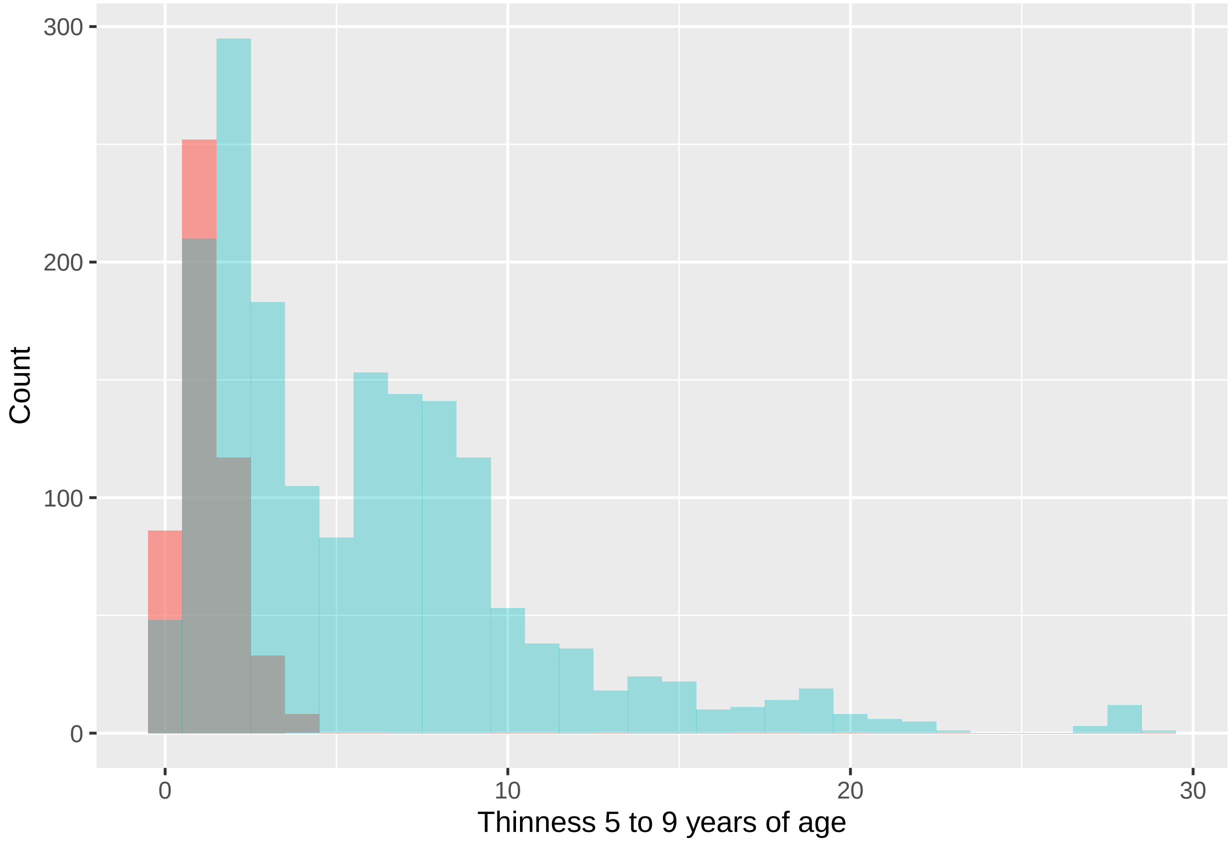
```
ggplot(data = le_adj, aes(x = alcohol))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 1, fill="#F8766D", alpha = .7)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 1, fill="#00BFC4", alpha = .35)+ labs
```



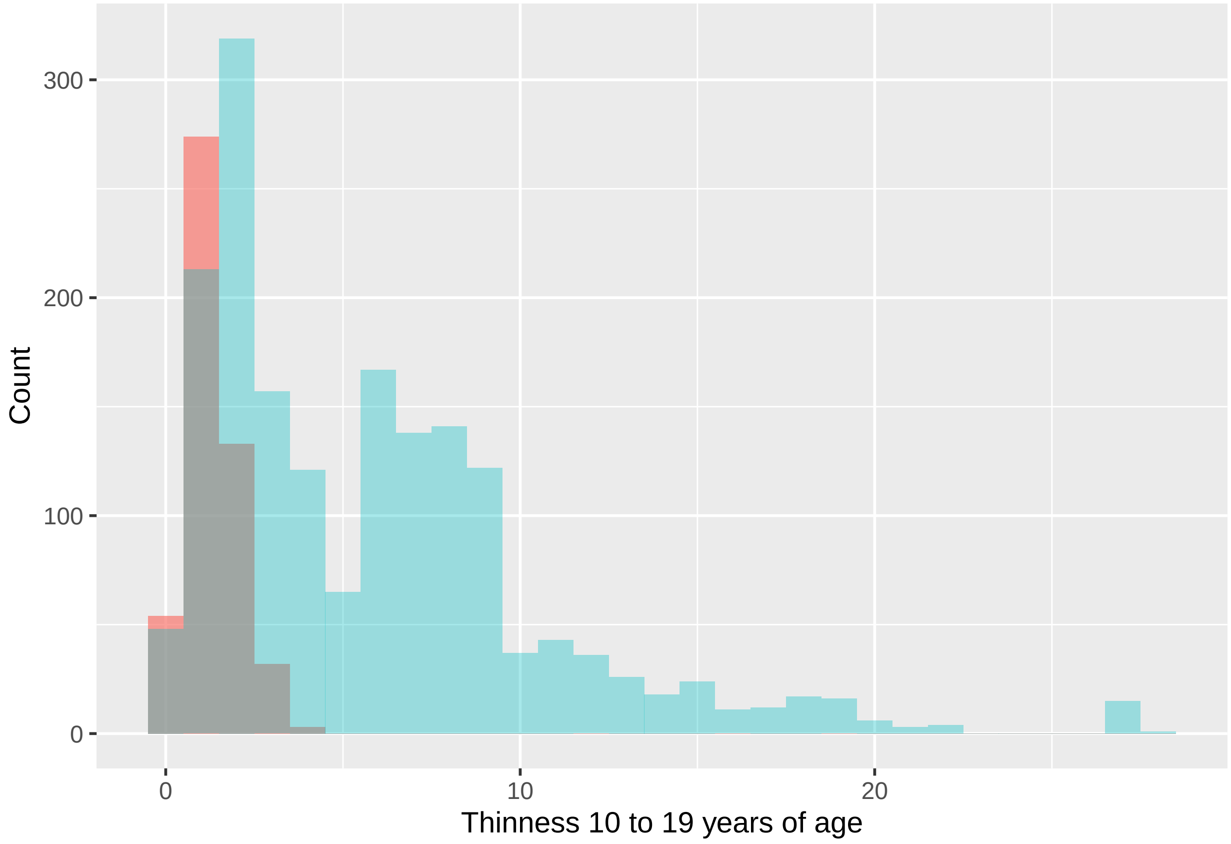
```
ggplot(data = le_adj, aes(x = hiv_aids))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 2, fill="#F8766D", alpha = .7)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 2, fill="#00BFC4", alpha = .35)+ labs
```



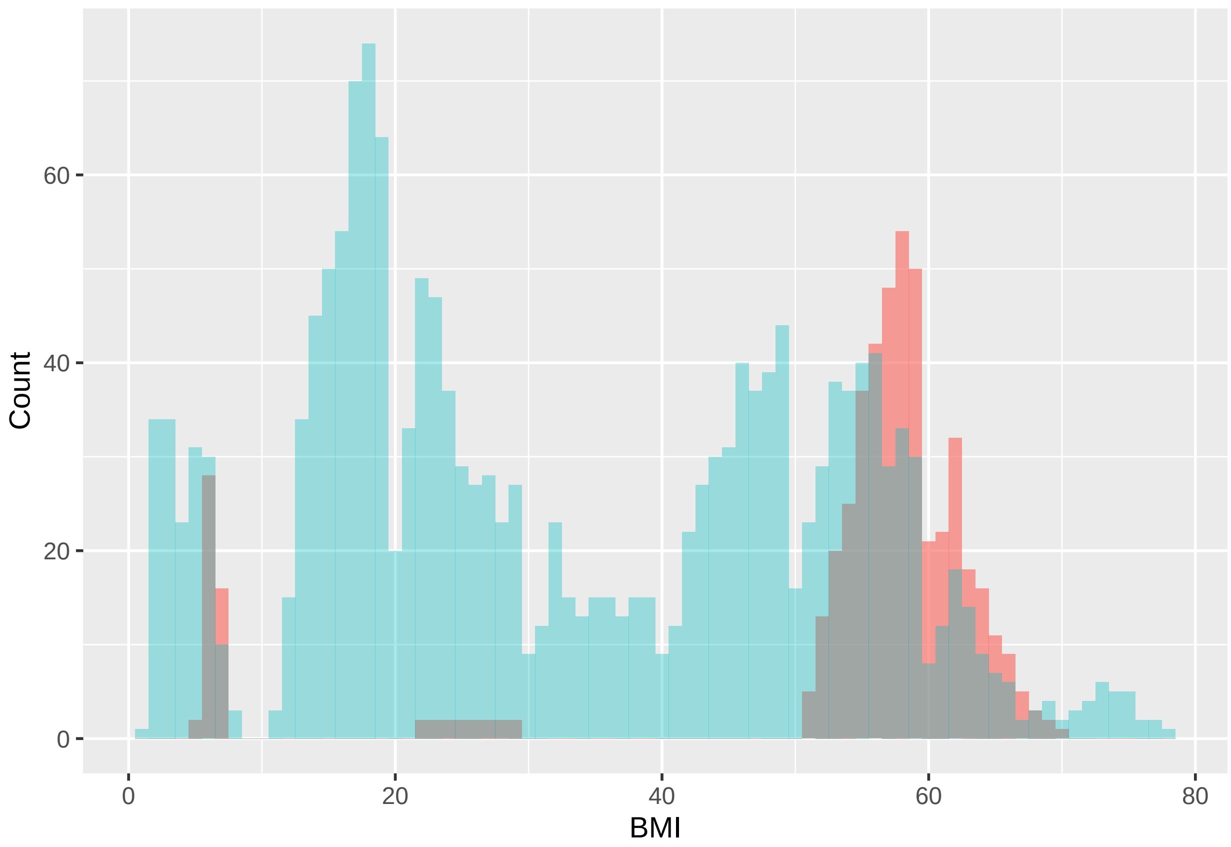
```
ggplot(data = le_adj, aes(x = thin_5to9_years))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 1, fill="#F8766D", alpha = .7)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 1, fill="#00BFC4", alpha = .35)+ labs
```



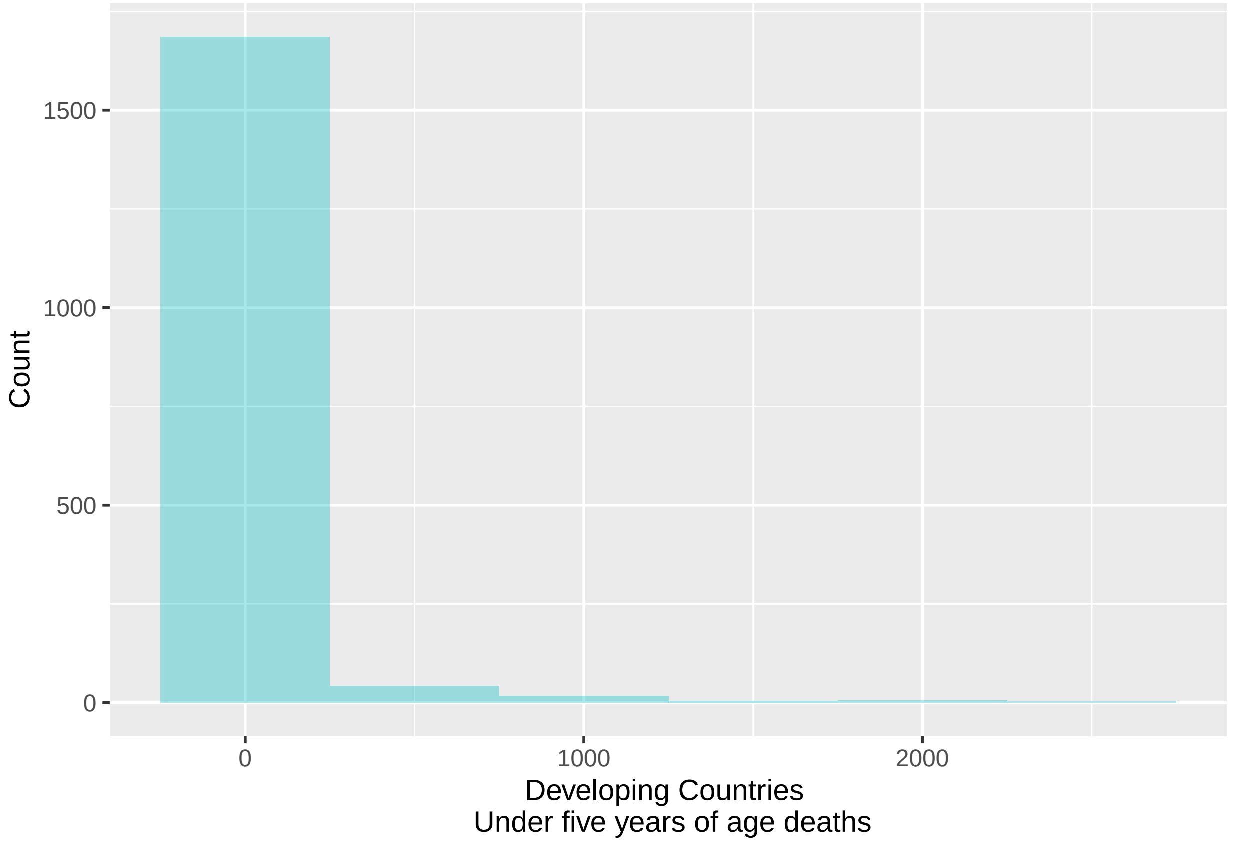
```
ggplot(data = le_adj, aes(x = thin_10to19_years))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 1, fill="#F8766D", alpha = .7)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 1, fill="#00BFC4", alpha = .35)+ labs
```



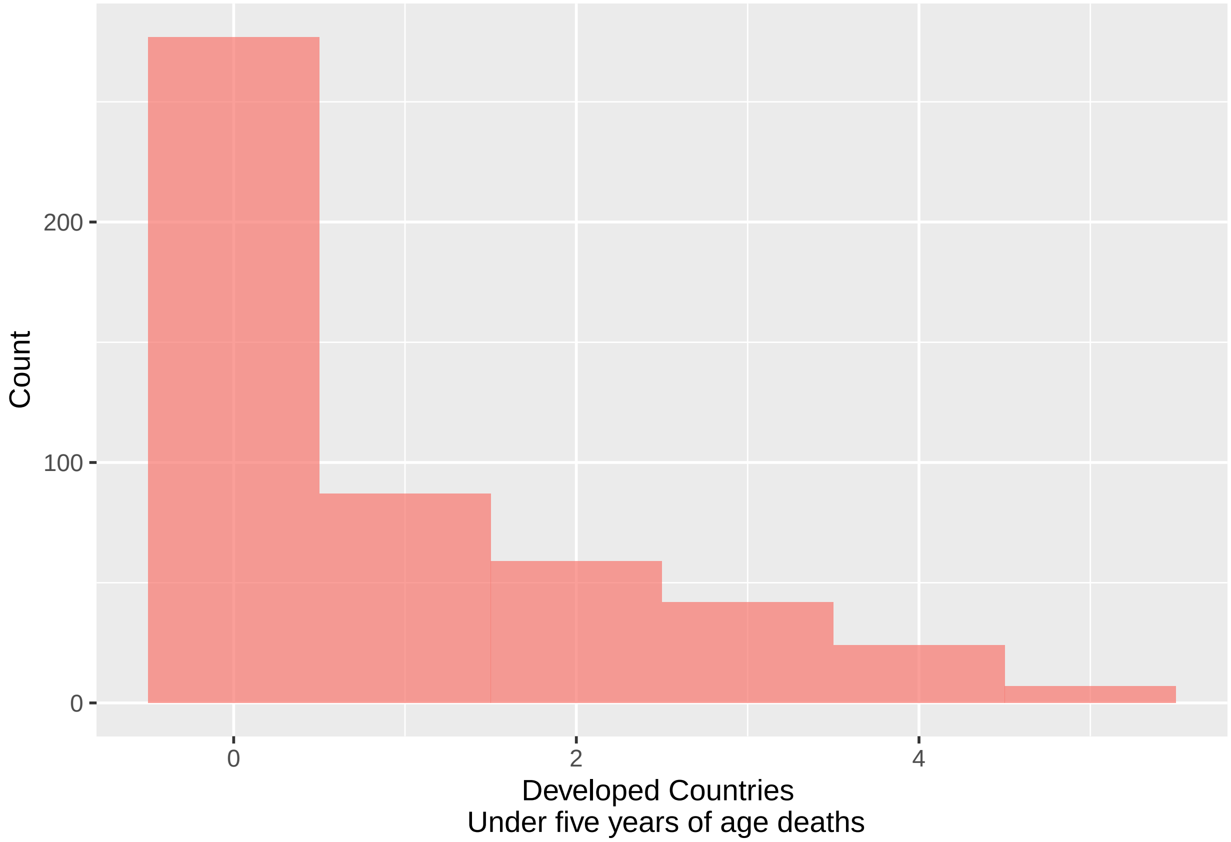
```
ggplot(data = le_adj, aes(x = bmi))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 1, fill="#F8766D", alpha = .7)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 1, fill="#00BFC4", alpha = .35)+ labs
```



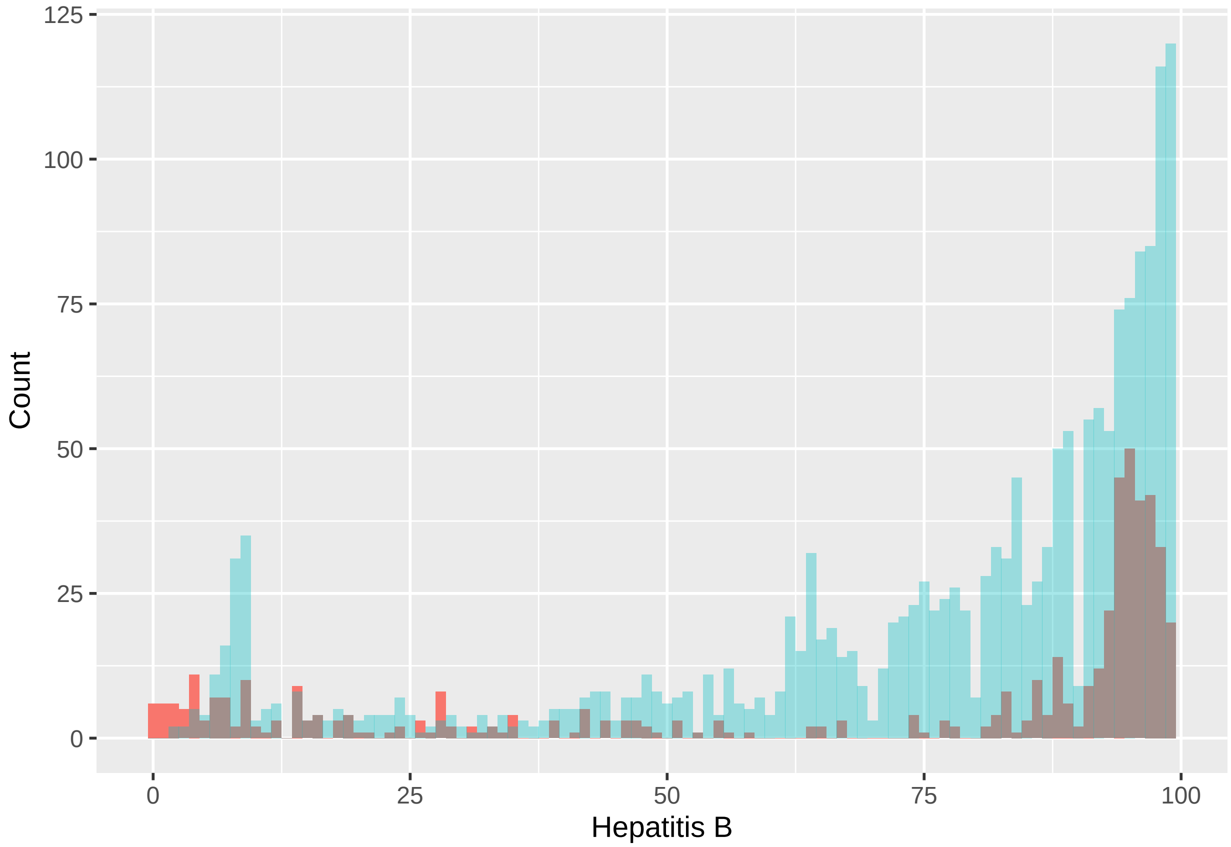
```
ggplot(data = le_adj, aes(x = under_five_deaths))+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 500, fill="#00BFC4", alpha = .35)+ lab
```



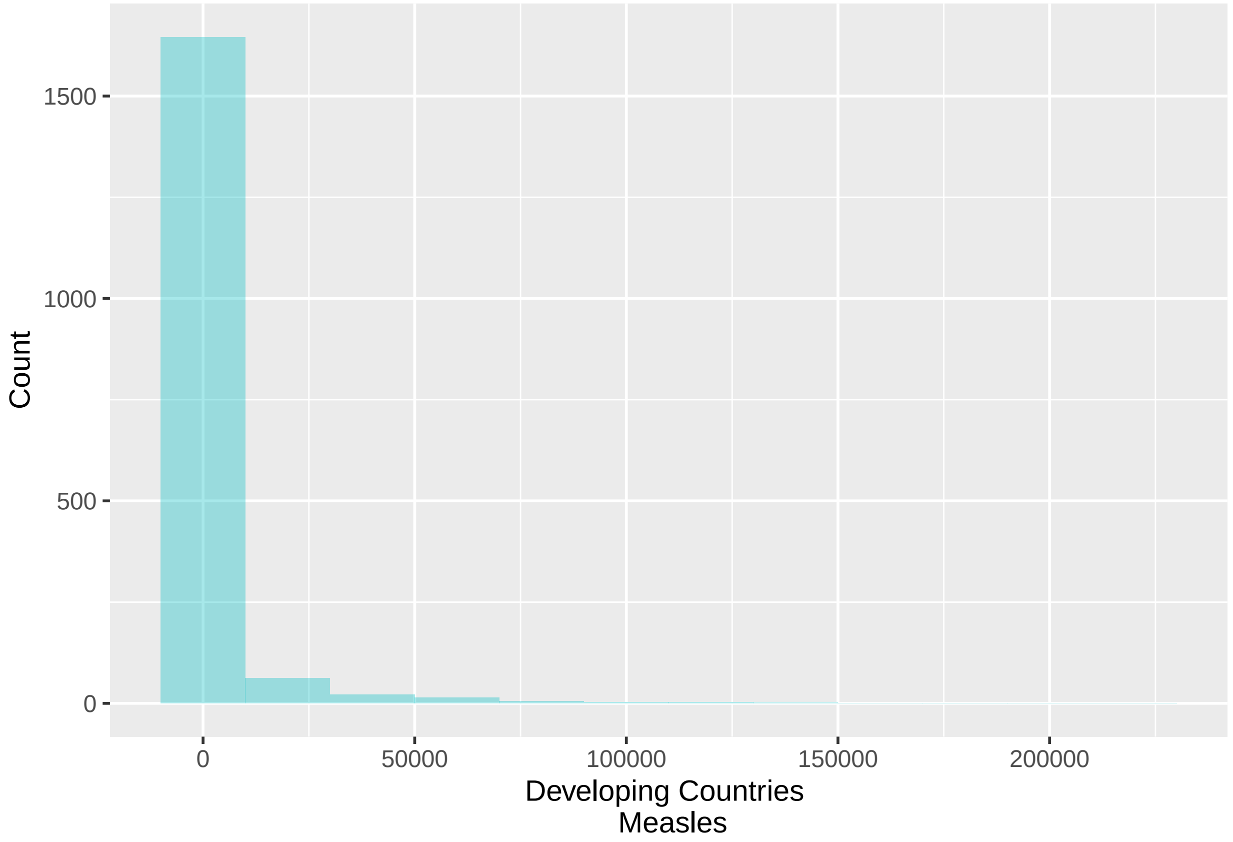
```
ggplot(data = le_adj, aes(x = under_five_deaths))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 1, fill="#F8766D", alpha = .7)+
   labs(x = "Developed Countries \n Under five years of age deaths", y = "Count")
```



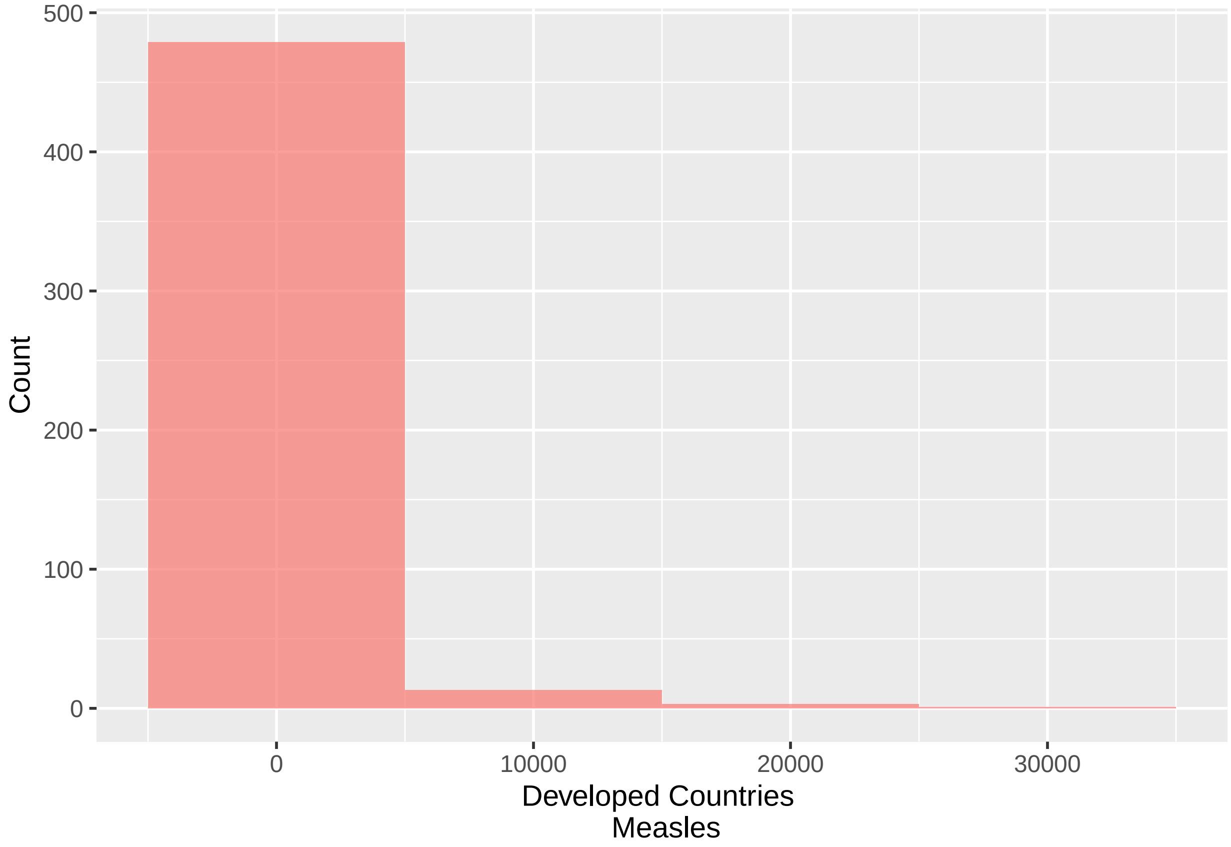
```
ggplot(data = le_adj, aes(x = hepatitis_b))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 1, fill="#F8766D", alpha = 1)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 1, fill="#00BFC4", alpha = .35)+ labs
```



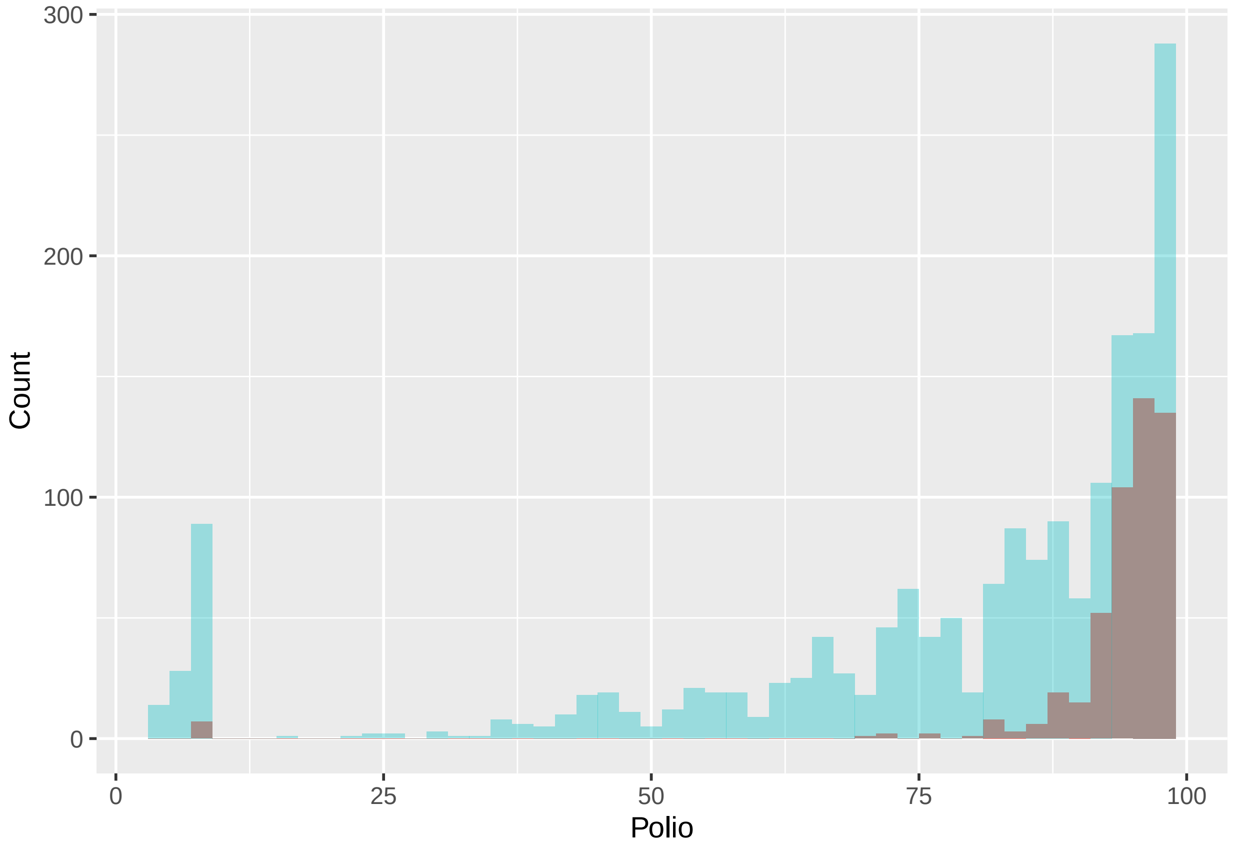
```
ggplot(data = le_adj, aes(x = measles))+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 20000, fill="#00BFC4", alpha = .35)+
```



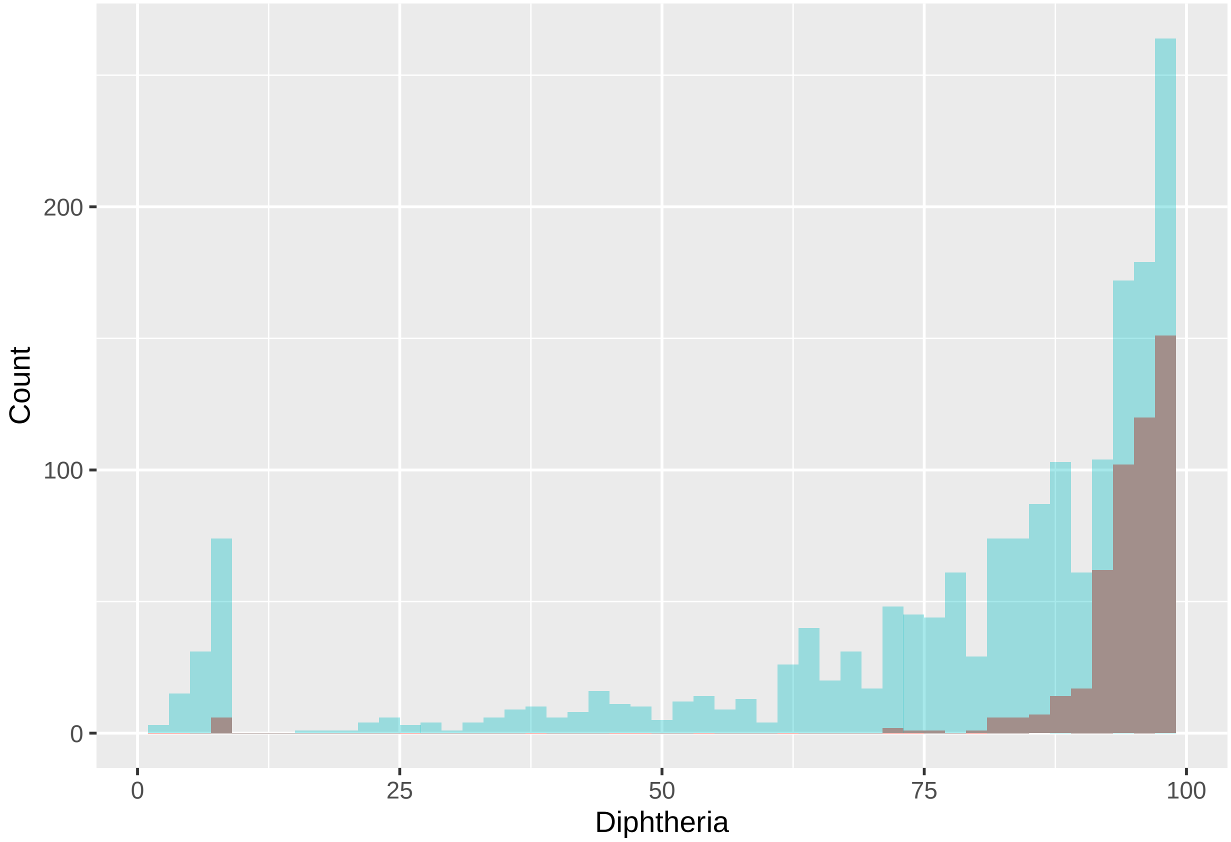
```
ggplot(data = le_adj, aes(x = measles))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 10000, fill="#F8766D", alpha = .7)+
  labs(x = "Developed Countries \n Measles", y = "Count")
```



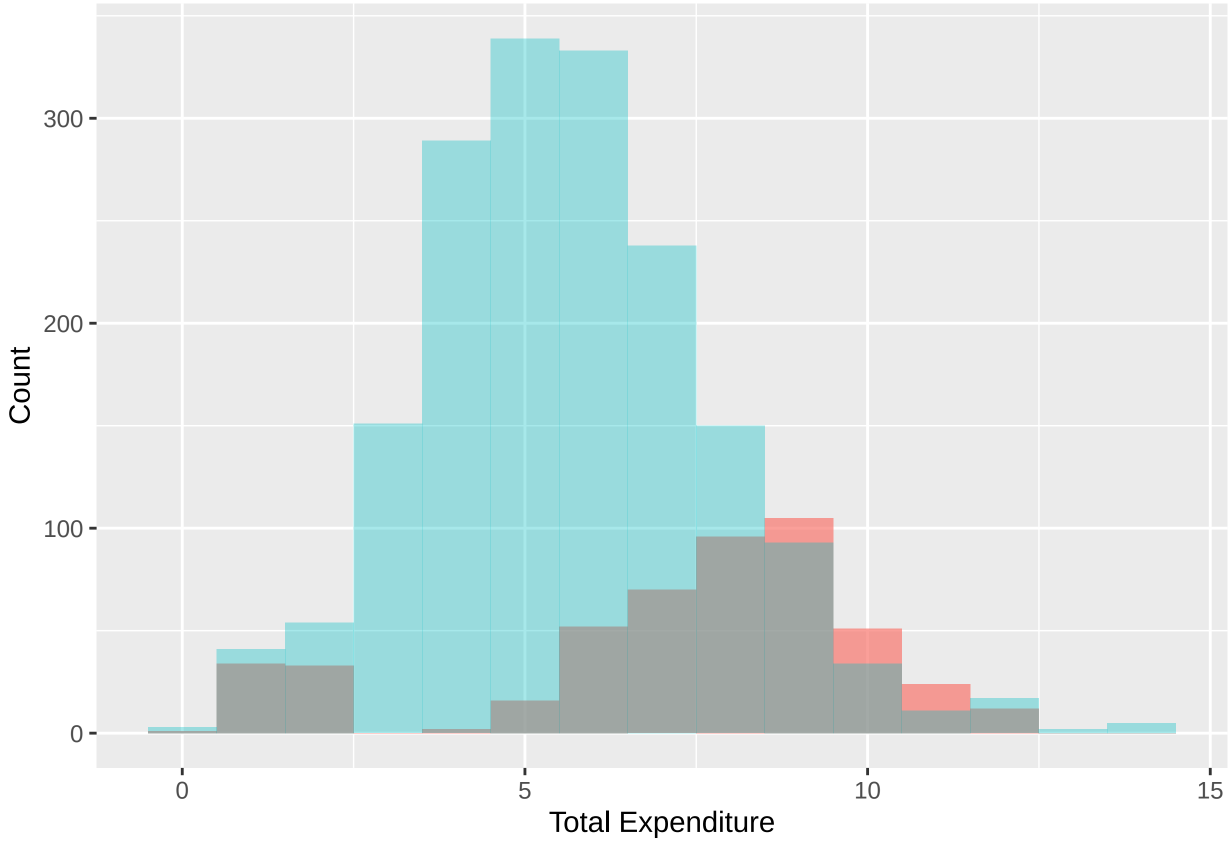
```
ggplot(data = le_adj, aes(x = polio))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 2, fill="#F8766D", alpha = 1)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 2, fill="#00BFC4", alpha = .35)+ labs
```



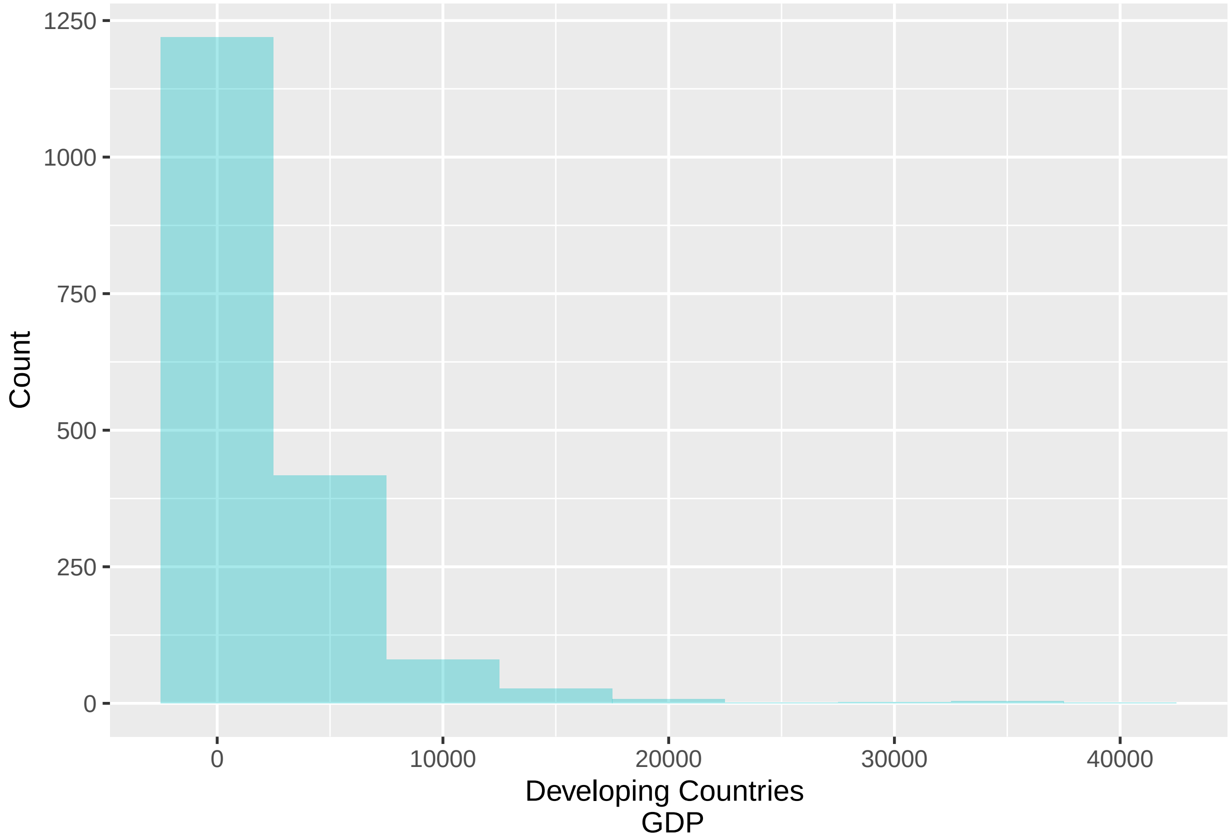
```
ggplot(data = le_adj, aes(x = diphtheria))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 2, fill="#F8766D", alpha = 1)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 2, fill="#00BFC4", alpha = .35)+ labs
```



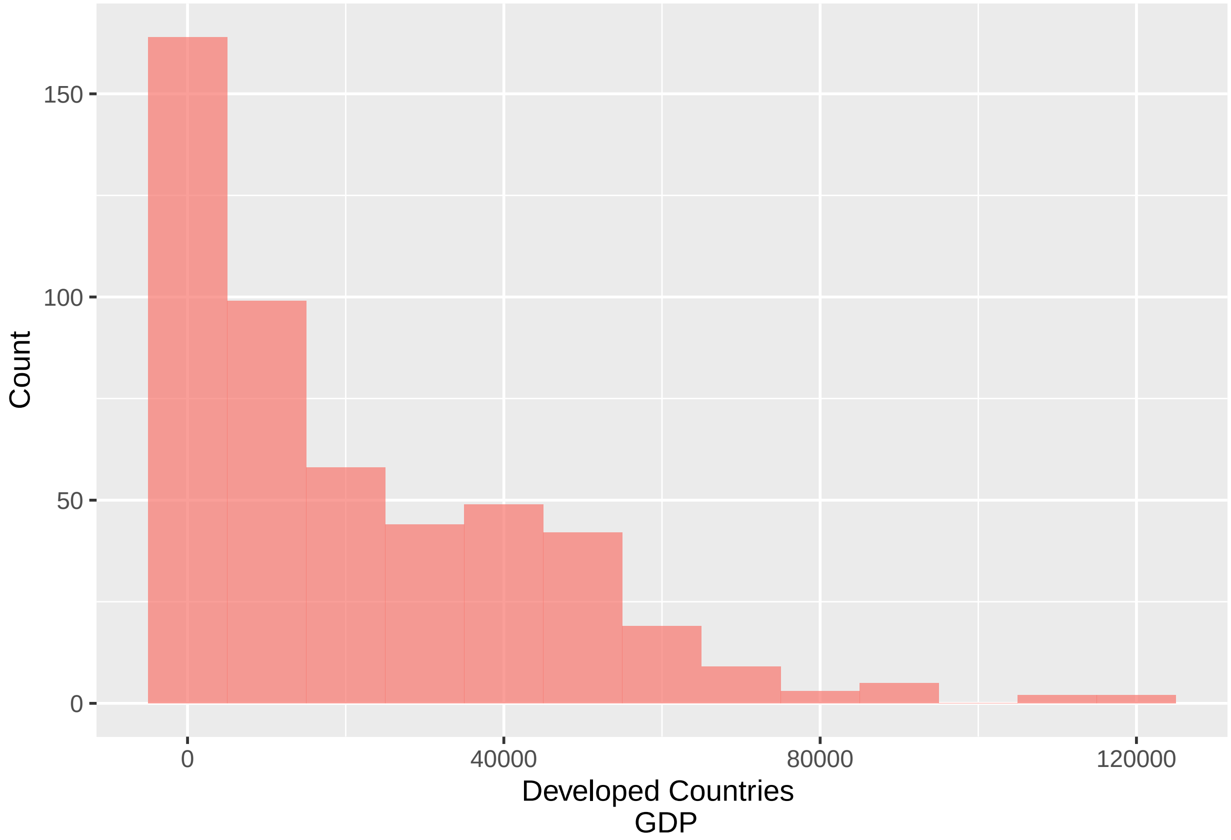
```
ggplot(data = le_adj, aes(x = total_expenditure))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 1, fill="#F8766D", alpha = .7)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 1, fill="#00BFC4", alpha = .35)+ labs
```



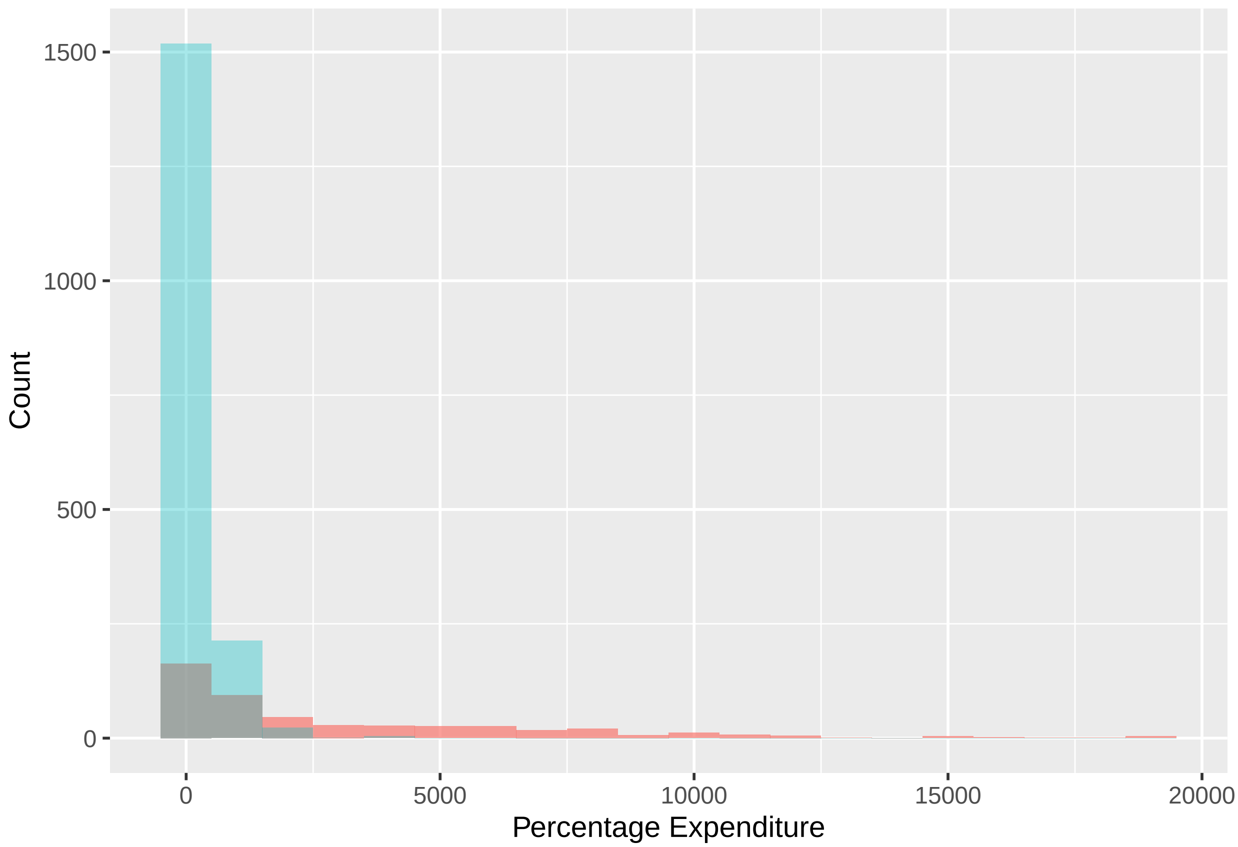
```
ggplot(data = le_adj, aes(x = gdp))+
  #geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 5000, fill="#F8766D", alpha = .7)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 5000, fill="#00BFC4", alpha = .35)+ la
```



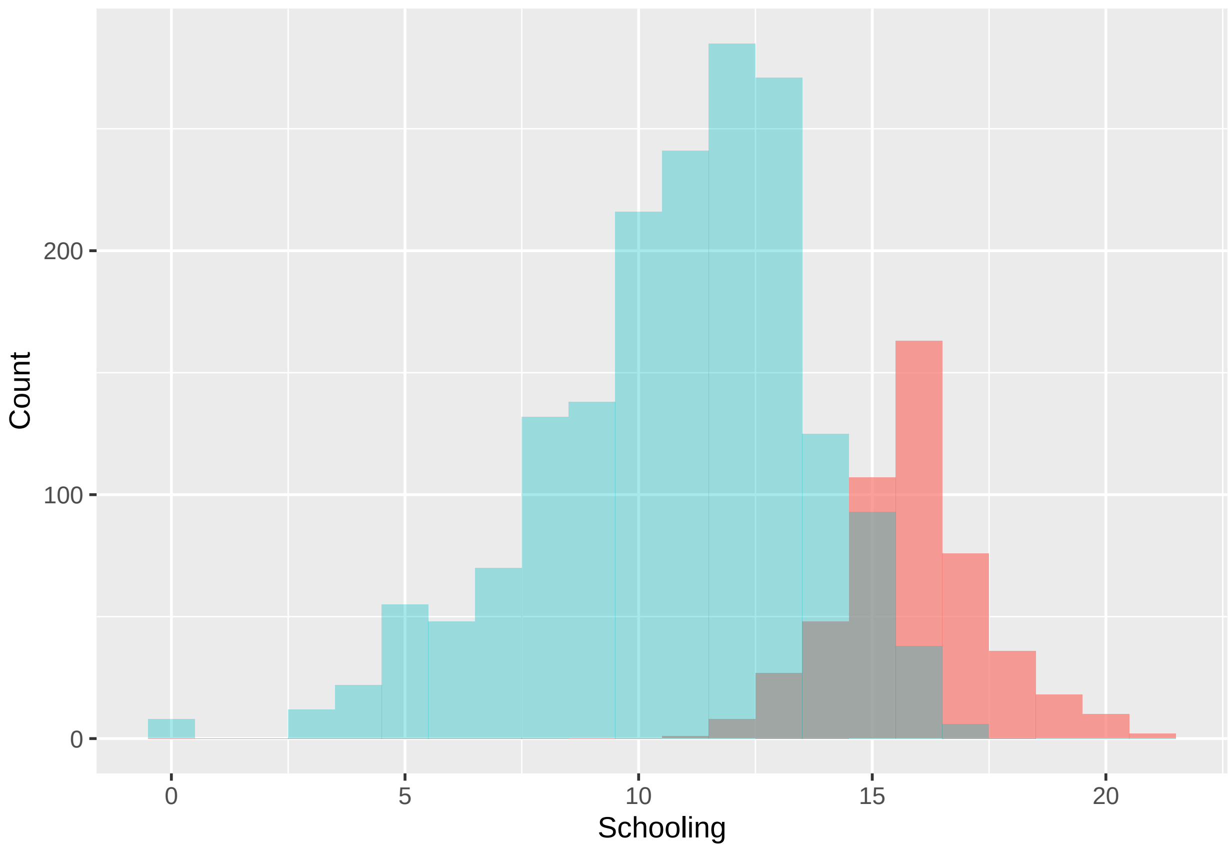
```
ggplot(data = le_adj, aes(x = gdp))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 10000, fill="#F8766D", alpha = .7)+
  #geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 5000, fill="#00BFC4", alpha = .35)+
  labs(x = "Developed Countries \n GDP", y = "Count")
```



```
ggplot(data = le_adj, aes(x = percentage_expenditure))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 1000, fill="#F8766D", alpha = .7)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 1000, fill="#00BFC4", alpha = .35)+ la
```



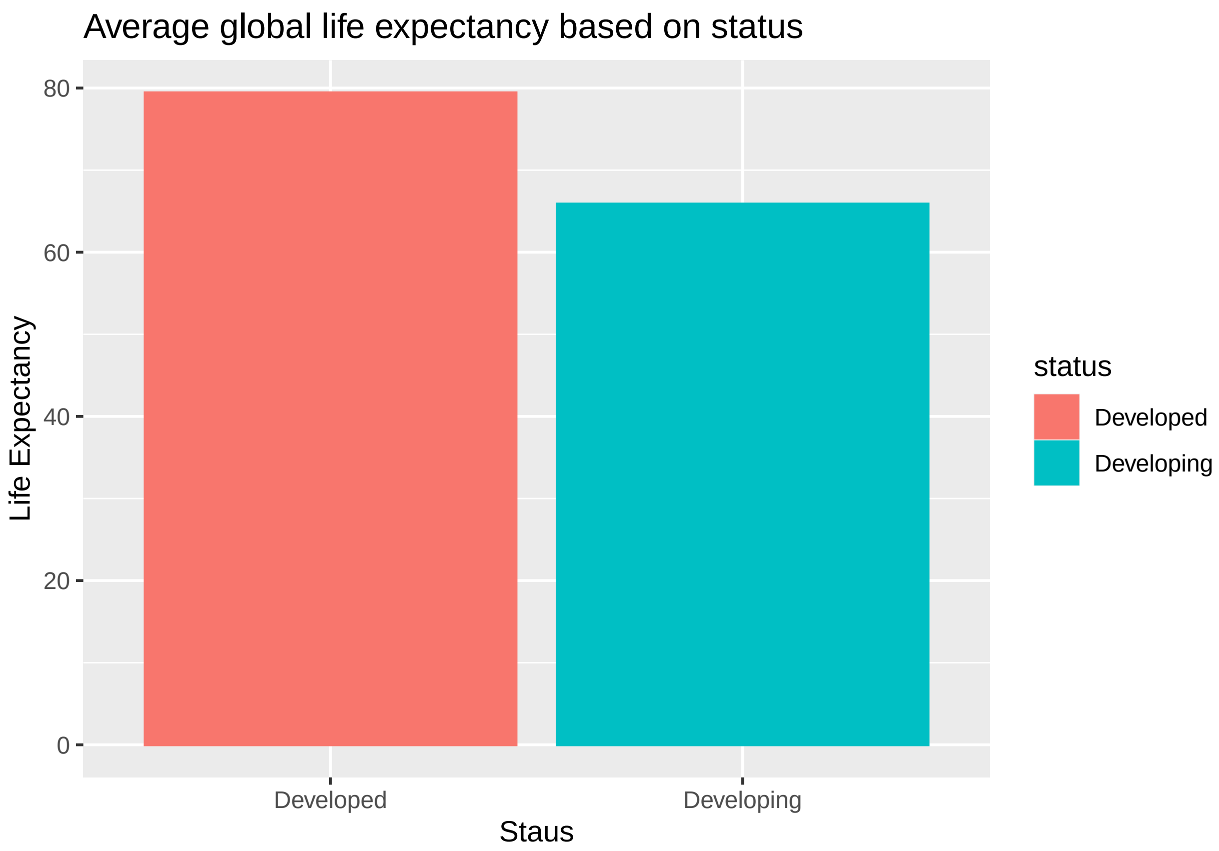
```
ggplot(data = le_adj, aes(x = schooling))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 1, fill="#F8766D", alpha = .7)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 1, fill="#00BFC4", alpha = .35)+ labs
```



Looking at difference between avg developed and avg developing life expectancy

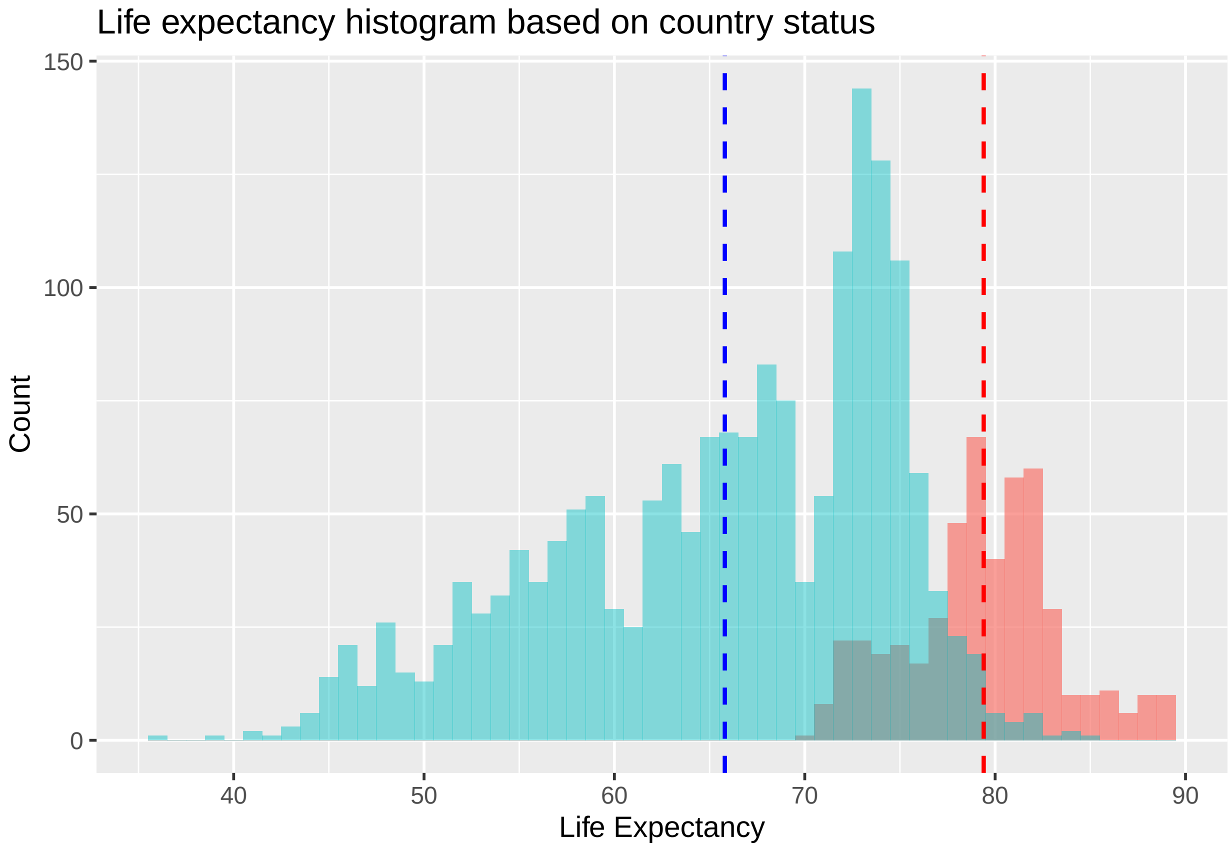
```
ggplot(le_adj, aes(x=factor(status), y=life_expectancy, color=status, fill=status)) +
  stat_summary(fun.y="mean", geom="bar")+
  labs(y = "Life Expectancy", x = "Staus", title="Average global life expectancy based on status")
```

```
## Warning: `fun.y` is deprecated. Use `fun` instead.
```



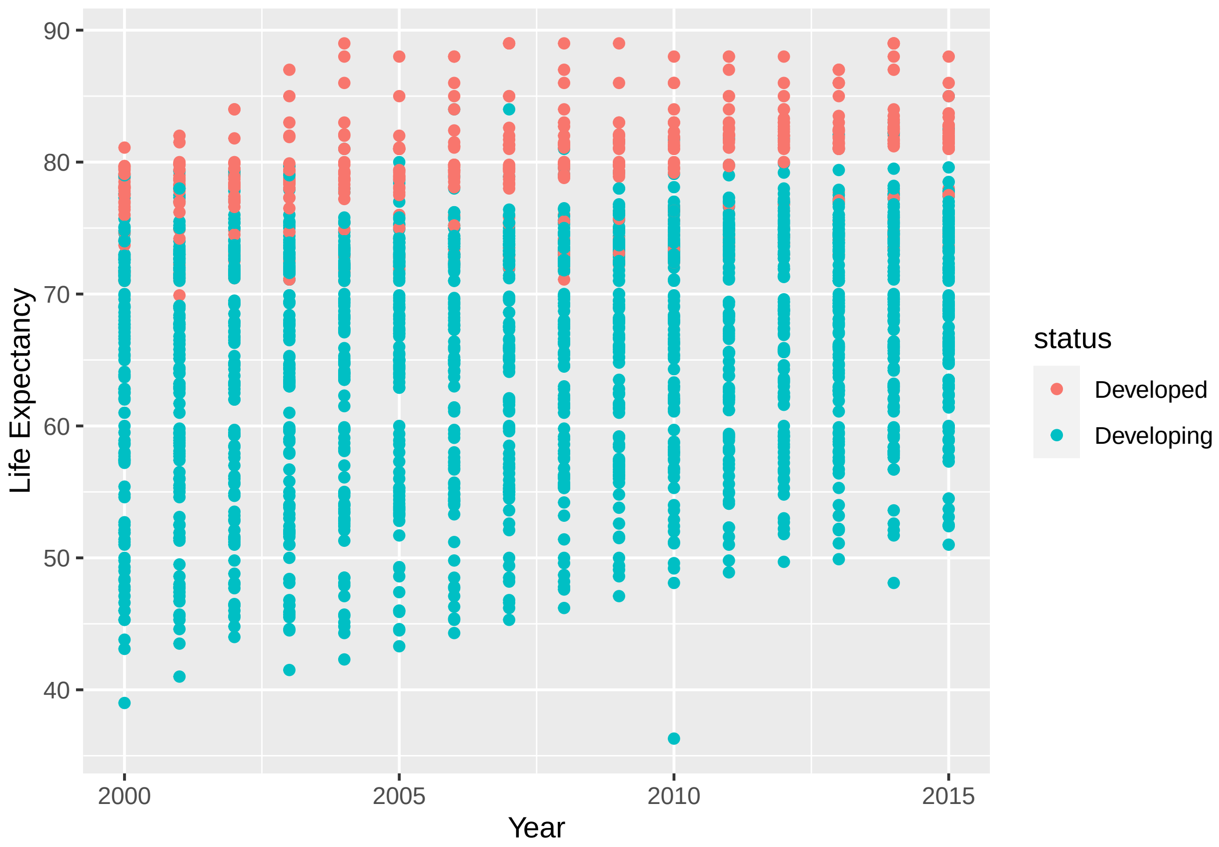
Further looking at differences between country status

```
ggplot(data = le_adj, aes(x = life_expectancy))+
  geom_histogram(data = subset(le_adj, stat_num == 0), binwidth = 1, fill="#F8766D", alpha = .7)+
  geom_histogram(data = subset(le_adj, stat_num == 1), binwidth = 1, fill="#00BFC4", alpha = .45)+
  geom_vline(aes(xintercept=79.4),color="red", linetype="dashed", size=.75) +
  geom_vline(aes(xintercept=65.8),color="blue", linetype="dashed", size=.75)+ labs(x = "Life Expectancy
```

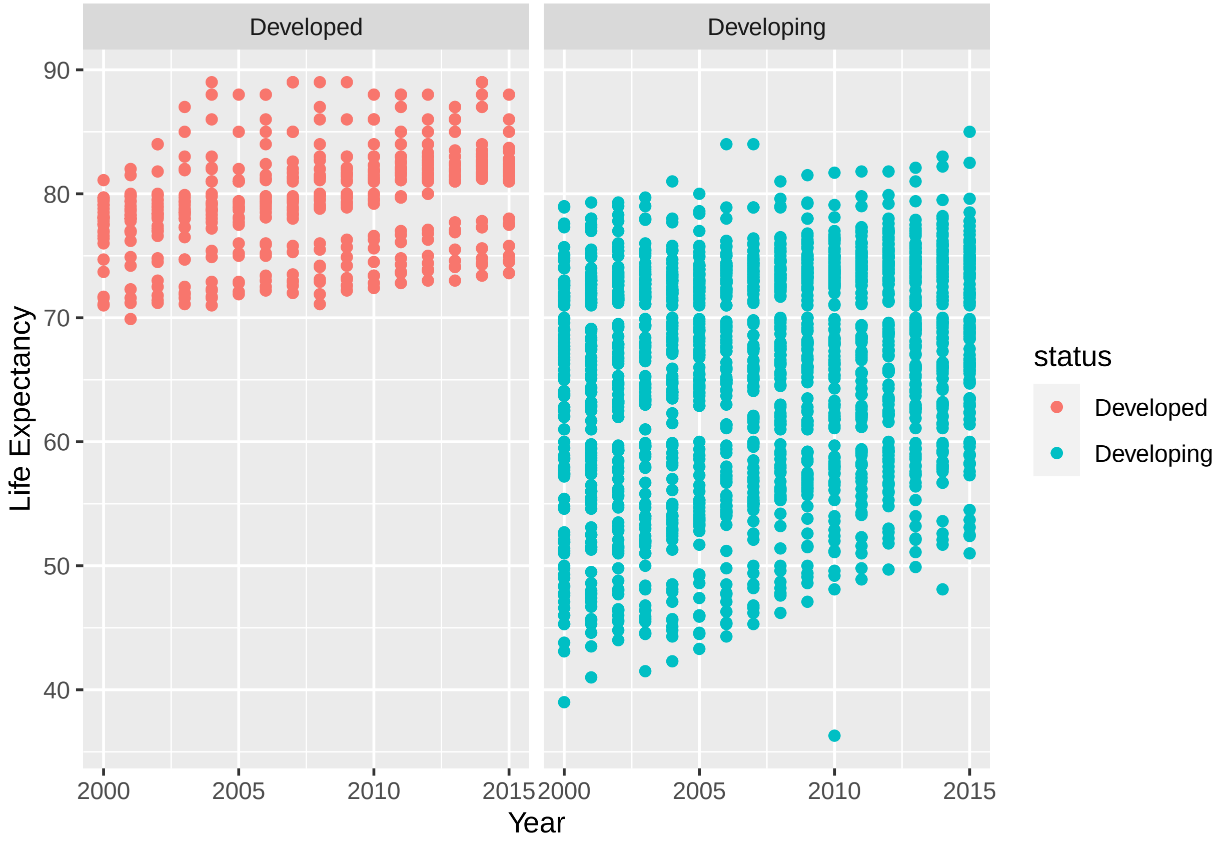


Life expectancy with year attribute

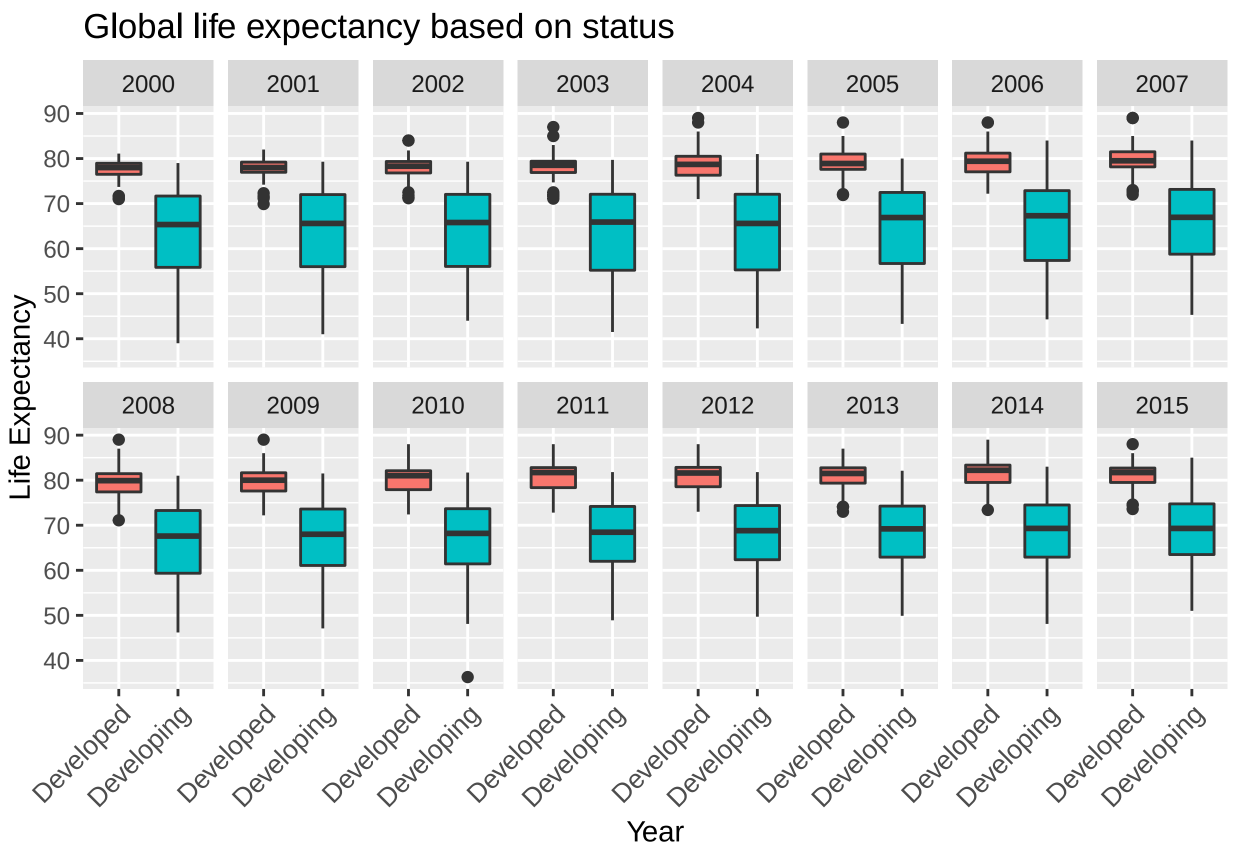
```
ggplot(data = le_adj, aes(x = year, y = life_expectancy, color= status))+
  geom_point() + labs(x = "Year", y = "Life Expectancy")
```



```
ggplot(data = le_adj, aes(x = year, y = life_expectancy, color=status))+
  geom_point() + labs(x = "Year", y = "Life Expectancy")+
  facet_wrap(~status)
```



```
ggplot(data = le_adj, aes(x = status, y=life_expectancy))+
  geom_boxplot(data = subset(le_adj, stat_num == 0), fill="#F8766D") +
  geom_boxplot(data = subset(le_adj, stat_num == 1), fill="#00BFC4") +
  labs(x = "Year", y = "Life Expectancy", title= "Global life expectancy based on status")+
  facet_wrap(~year,nrow = 2,  ncol = 8,)+
  theme(axis.text.x = element_text(angle = 45, vjust = 1,
    size = 10, hjust = 1))
```



```
le_new = le_adj[le_adj$year==2015,c(6,7,11,12,13,16,17,18,19)]
```

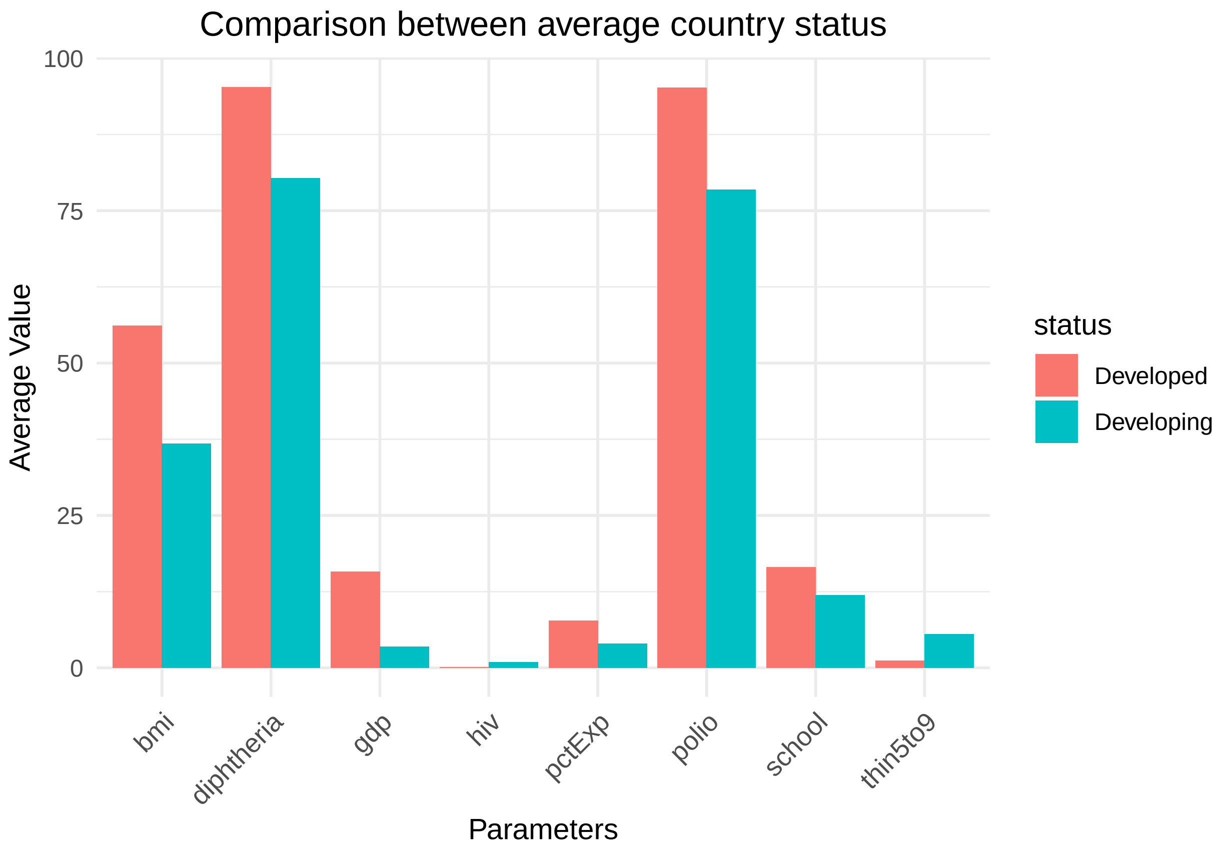
```
try = le_new %>%
    group_by(status) %>%
    summarize(
             hiv = mean(hiv_aids),
             thin5to9 = mean(thin_5to9_years),
             polio = mean(polio),
             diphtheria = mean(diphtheria),
             bmi = mean(bmi),
             gdp = mean(gdp)/1000,
             pctExp = mean(percentage_expenditure),
             school = mean(schooling))
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.1.3

## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v tibble  3.1.8     v stringr 1.4.1
## v tidyr   1.2.0     v forcats 0.5.2
## v purrr   0.3.4

## Warning: package 'tibble' was built under R version 4.1.3
## Warning: package 'tidyr' was built under R version 4.1.3

## Warning: package 'purrr' was built under R version 4.1.3

## Warning: package 'stringr' was built under R version 4.1.3

## Warning: package 'forcats' was built under R version 4.1.3

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

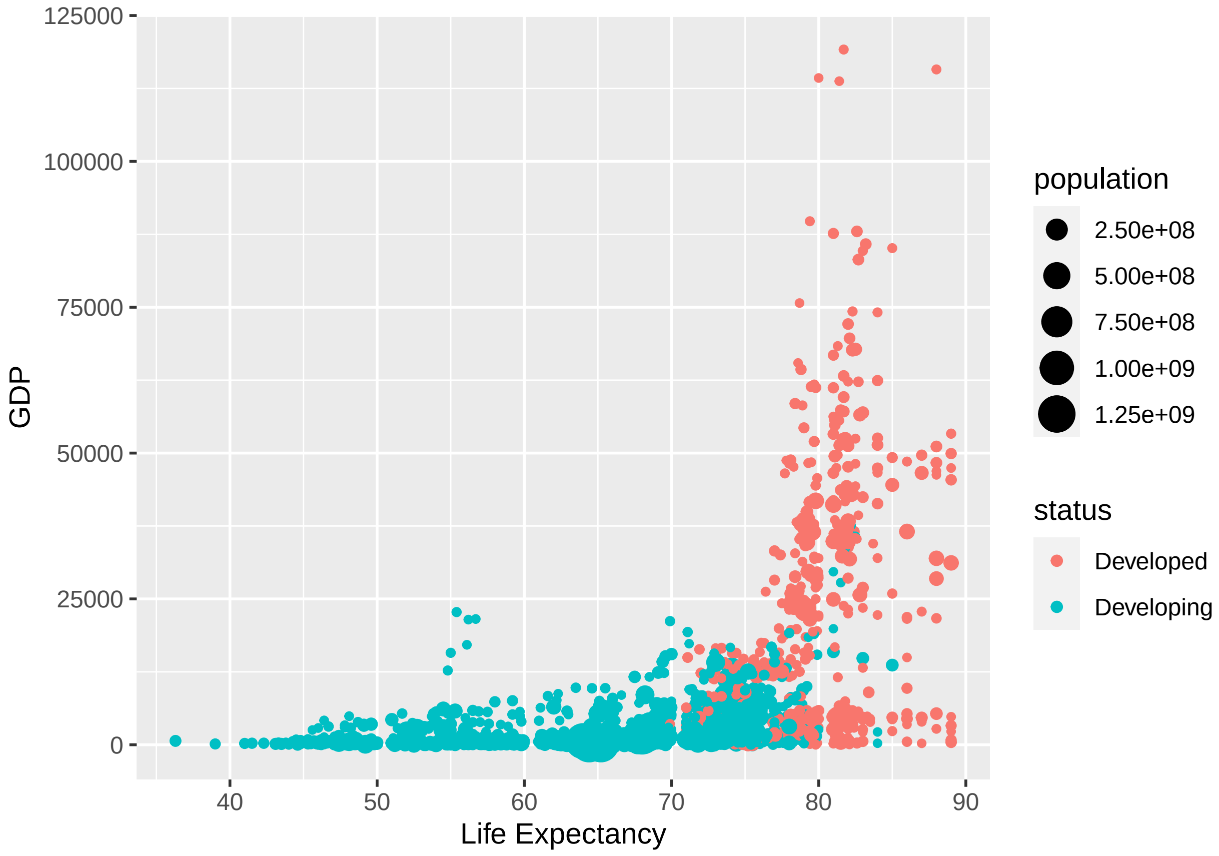
```
try2 <- gather(try, type, value, -status)
```

```
ggplot(try2, aes(type, value)) +
  geom_bar(aes(fill = status), stat = "identity", position = "dodge")+
  theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 10, hjust = 1))+
  ggtitle("Comparison between average country status")+
  labs(x = "Parameters", y = "Average Value")+
  theme_minimal()+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 10, hjust = 1))
```

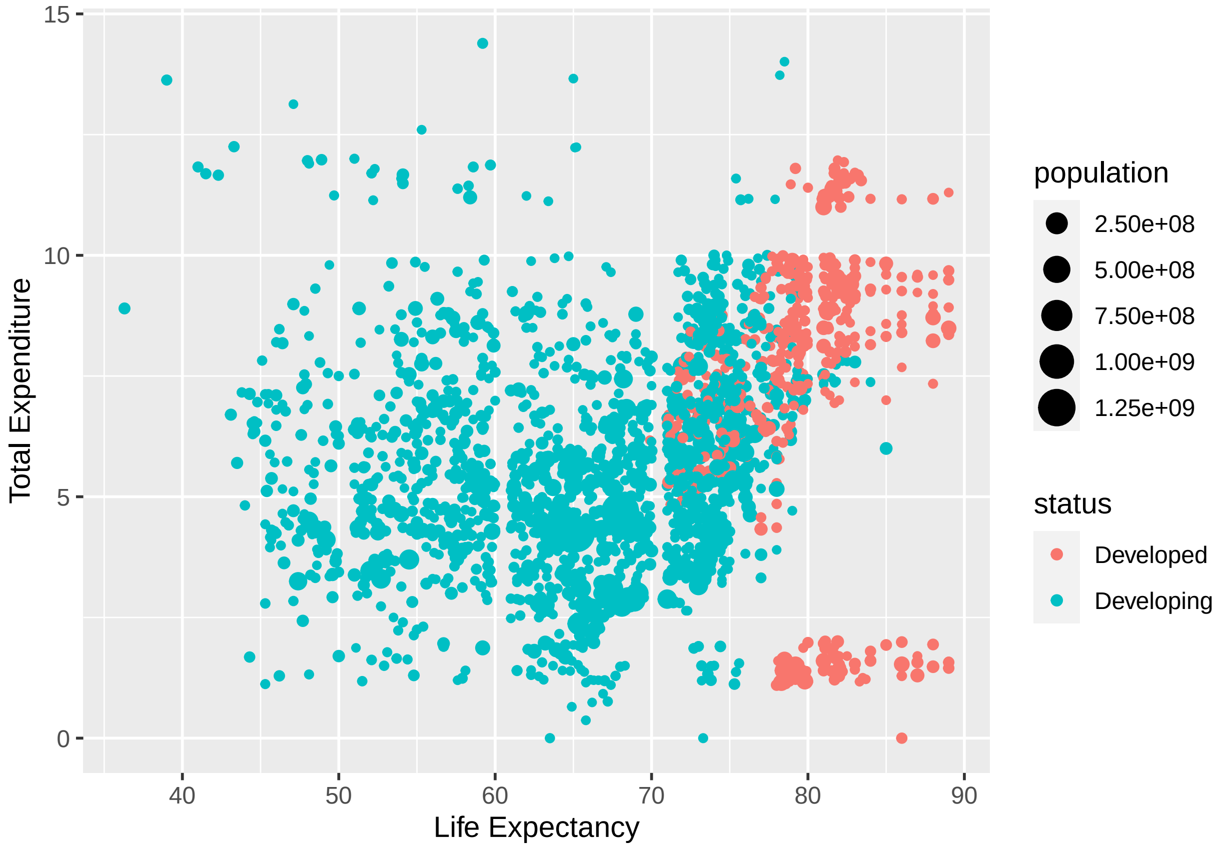


Looking at multi variable interactions

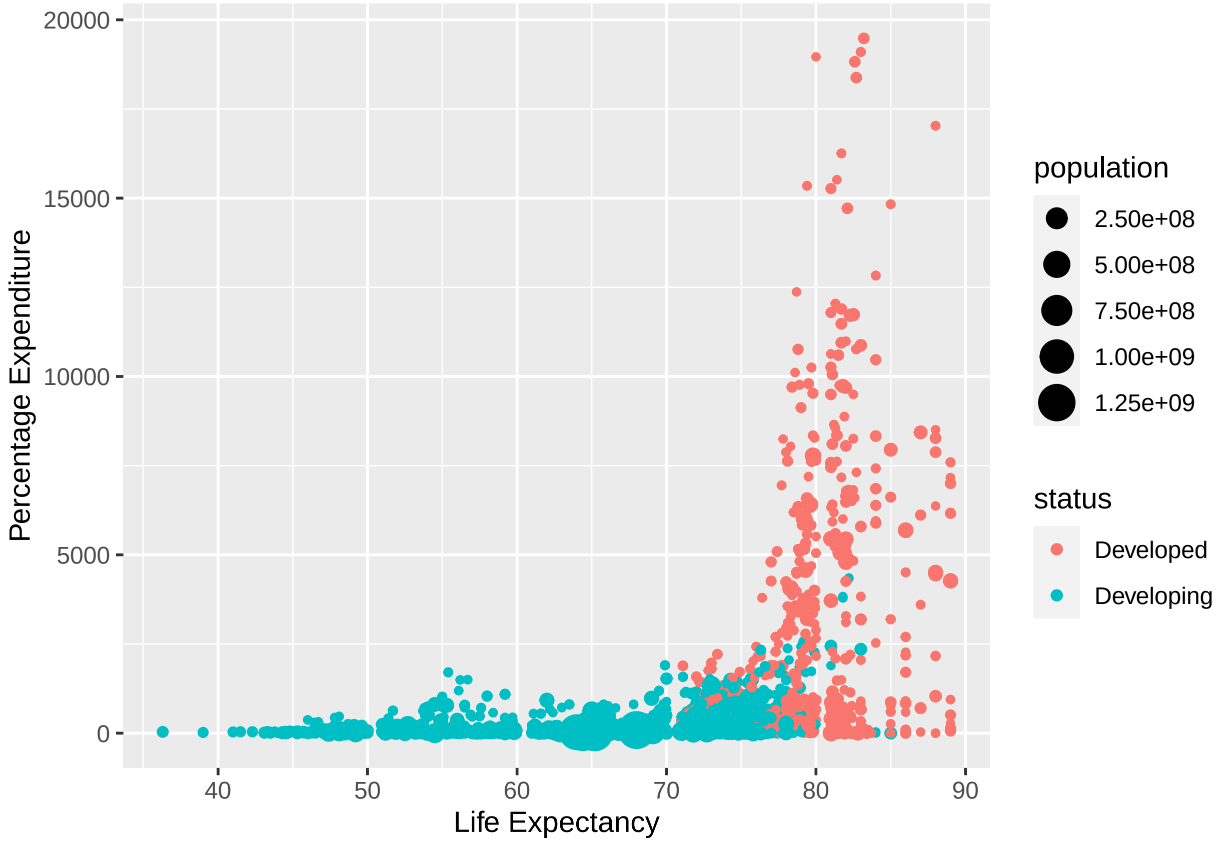
```
ggplot(data = le_adj, aes(x = life_expectancy, y = gdp, color= status, size=population))+
  geom_point() + labs(x = "Life Expectancy", y = "GDP")
```



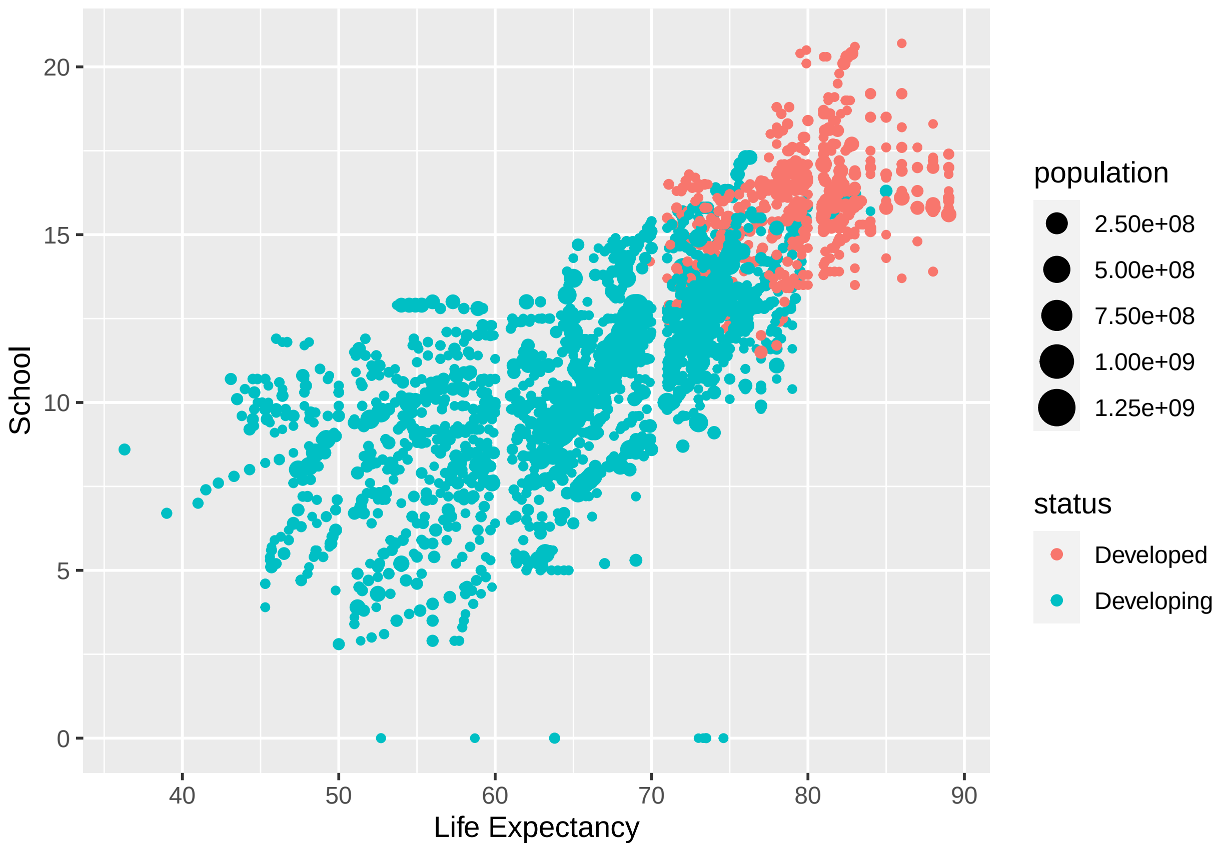
```
ggplot(data = le_adj, aes(x = life_expectancy, y = total_expenditure, color= status, size=population))+
  geom_point() + labs(x = "Life Expectancy", y = "Total Expenditure")
```



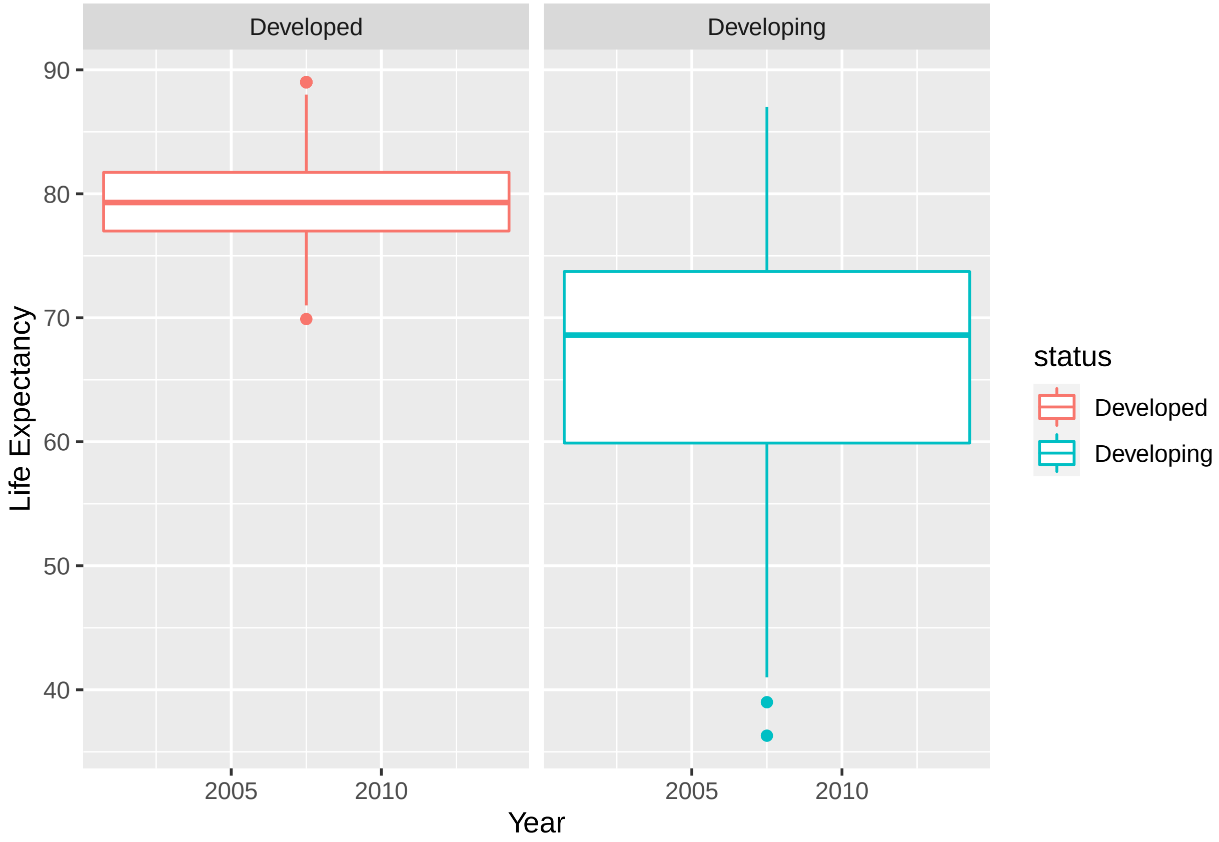
```
ggplot(data = le_adj, aes(x = life_expectancy, y = percentage_expenditure, color= status, size=populati
  geom_point() + labs(x = "Life Expectancy", y = "Percentage Expenditure")
```



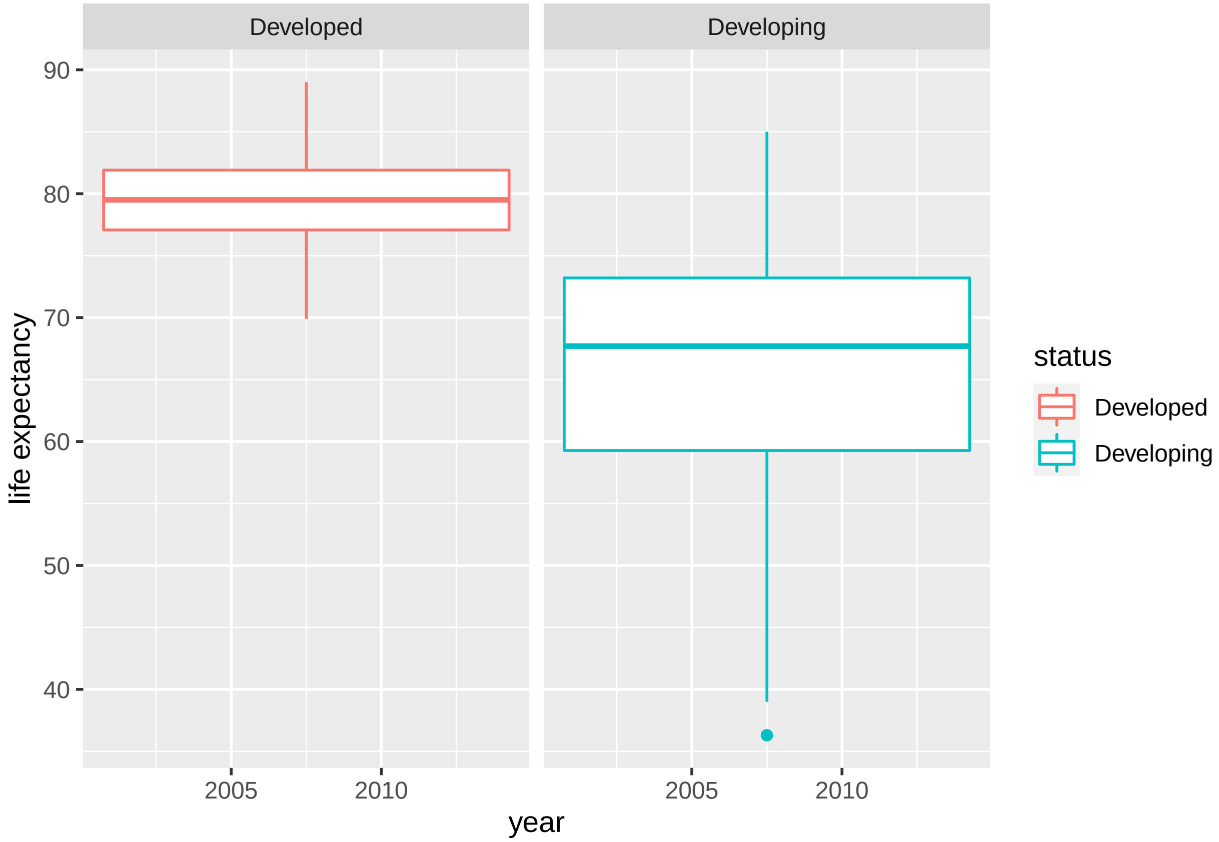
```
ggplot(data = le_adj, aes(x = life_expectancy, y = schooling, color= status, size=population))+
  geom_point() + labs(x = "Life Expectancy", y = "School")
```



```
ggplot(data = life_expec, aes(x = year, y = life_expectancy, color=status))+
  geom_boxplot() + labs(x = "Year", y = "Life Expectancy")+
  facet_wrap(~status)
```



```
ggplot(data = le_adj, aes(x = year, y = life_expectancy, color=status))+
  geom_boxplot() + labs(x = "year", y = "life expectancy")+
  facet_wrap(~status)
```

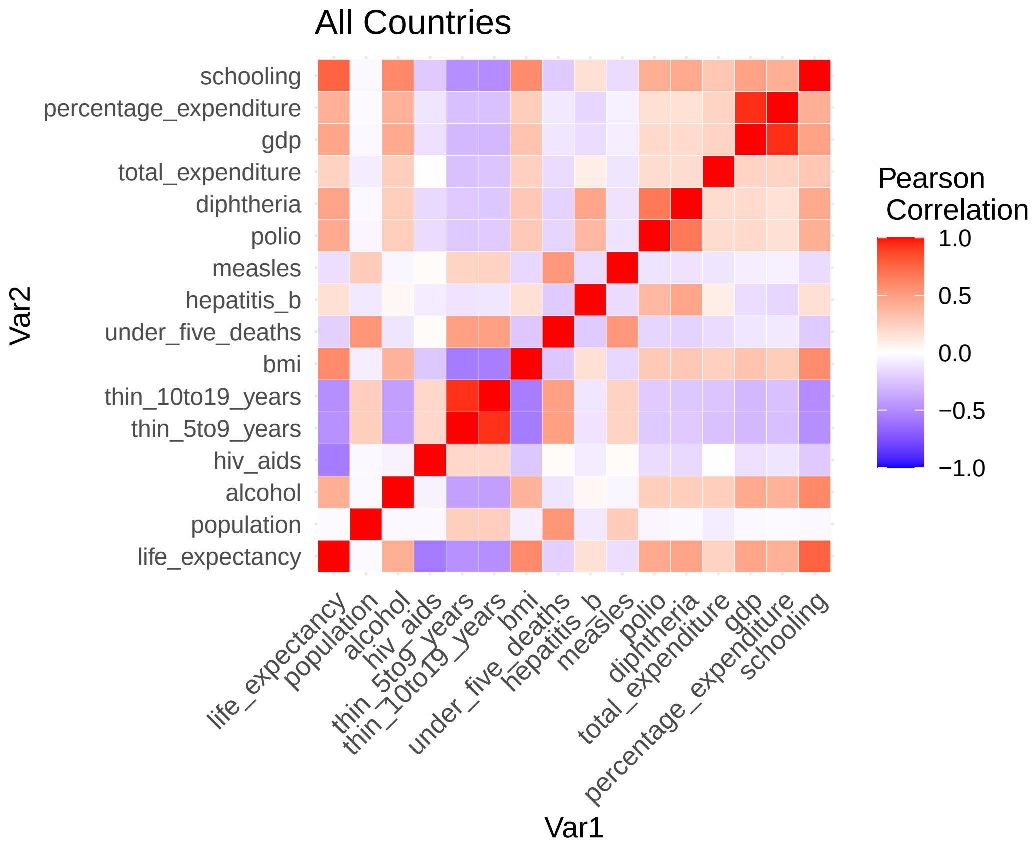


Correltation matrices

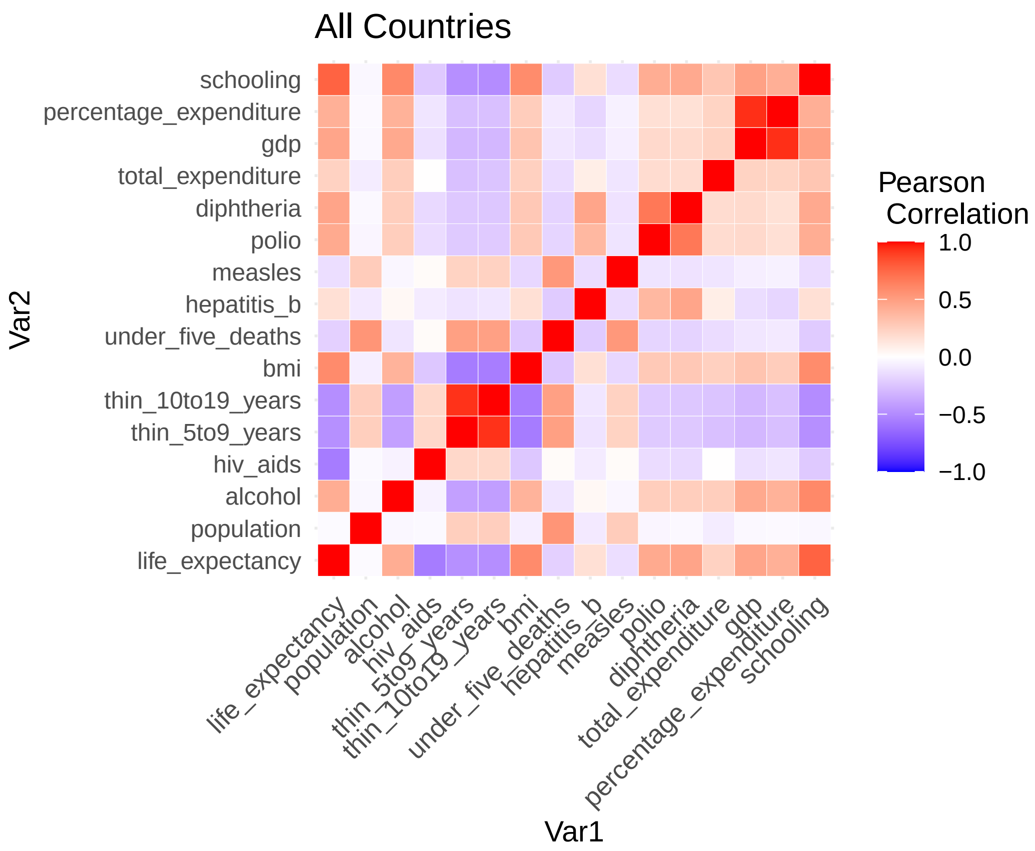
```
corplt <- le_adj %>%
    select(life_expectancy, population, alcohol, hiv_aids, thin_5to9_years,thin_10to19_years,bmi,under_
cormat <- cor(corplt)
melted <- reshape::melt(cormat)
```

```
## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by the
## caller; using TRUE

## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by the
## caller; using TRUE
```

```
melted<-melted%>%
  rename(Var1 = X1,
         Var2 = X2)

ggplot(data = melted, aes(x=Var1, y=Var2, fill=value)) +
  ggtitle("All Countries")+
  geom_tile(color = "white")+
 scale_fill_gradient2(low = "blue", high = "red", mid = "white",
   midpoint = 0, limit = c(-1,1), space = "Lab",
   name="Pearson \n Correlation") +
  theme_minimal()+
 theme(axis.text.x = element_text(angle = 45, vjust = 1,
    size = 10, hjust = 1))+
 coord_fixed()
```



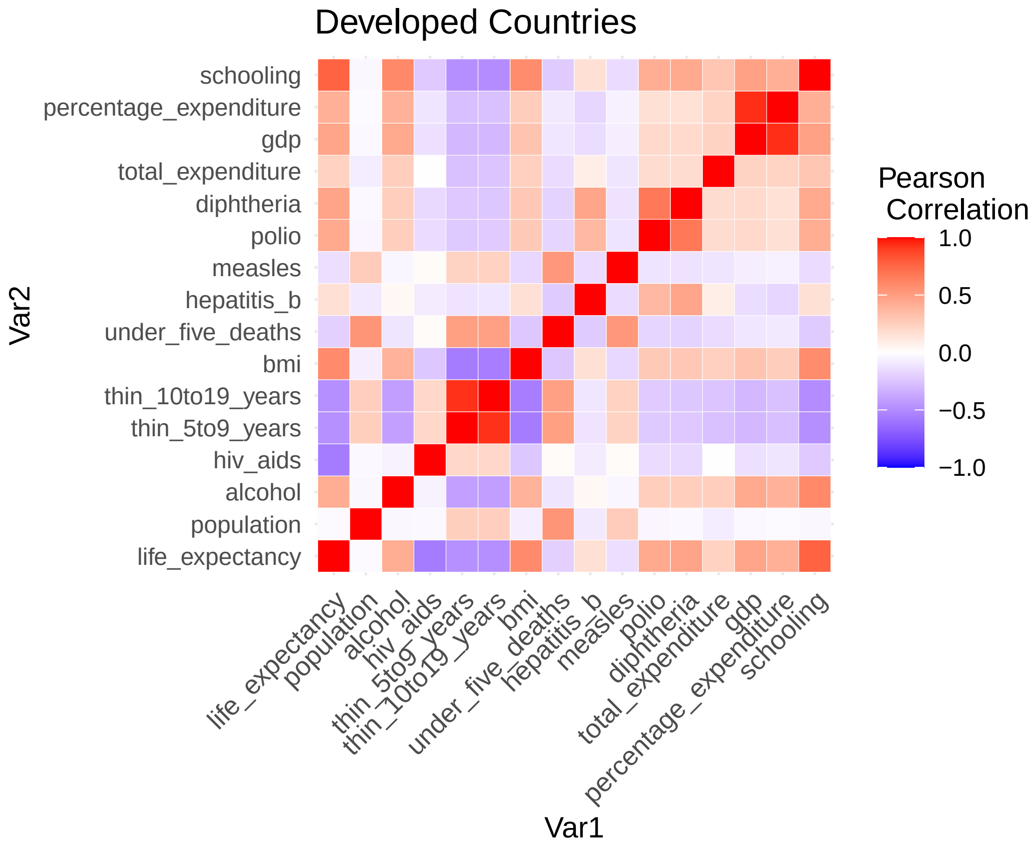
```
ggplot(data = melted, aes(x=Var1, y=Var2, fill=value)) +
  ggtitle("All Countries")+
  geom_tile(color = "white")+
 scale_fill_gradient2(low = "blue", high = "red", mid = "white",
   midpoint = 0, limit = c(-1,1), space = "Lab",
   name="Pearson \n Correlation") +
  theme_minimal()+
 theme(axis.text.x = element_text(angle = 45, vjust = 1,
    size = 10, hjust = 1))+
 coord_fixed()
```



```
corplot <- le_developed %>%
    select(life_expectancy, population, alcohol, hiv_aids, thin_5to9_years,thin_10to19_years,bmi,under_
cormat <- cor(corplt)
melted <- reshape::melt(cormat)
```

```
## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by the
## caller; using TRUE

## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by the
## caller; using TRUE
```

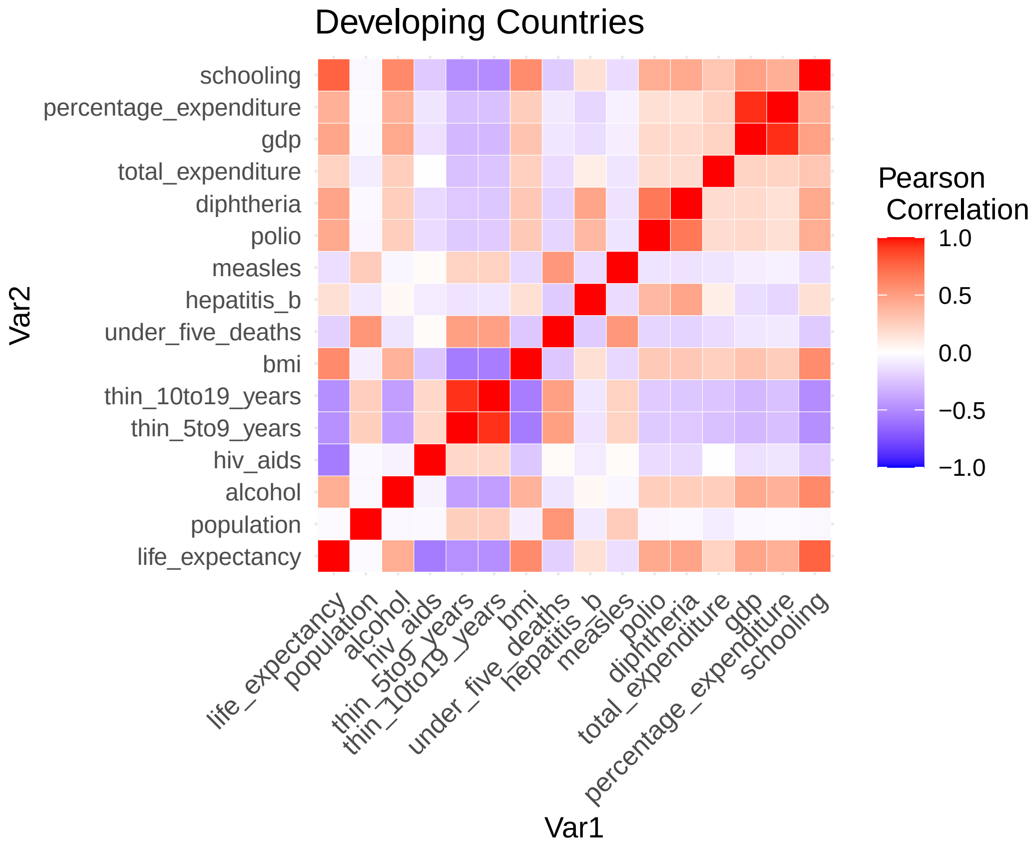
```
melted<-melted%>%
  rename(Var1 = X1,
         Var2 = X2)
ggplot(data = melted, aes(x=Var1, y=Var2, fill=value)) +
  ggtitle("Developed Countries")+
  geom_tile(color = "white")+
 scale_fill_gradient2(low = "blue", high = "red", mid = "white",
   midpoint = 0, limit = c(-1,1), space = "Lab",
   name="Pearson \n Correlation") +
  theme_minimal()+
 theme(axis.text.x = element_text(angle = 45, vjust = 1,
    size = 10, hjust = 1))+
 coord_fixed()
```



```
corplot <- le_developing %>%
    select(life_expectancy, population, alcohol, hiv_aids, thin_5to9_years,thin_10to19_years,bmi,under_
cormat <- cor(corplt)
melted <- reshape::melt(cormat)
```

```
## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by the
## caller; using TRUE

## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by the
## caller; using TRUE
```

```
melted<-melted%>%
  rename(Var1 = X1,
         Var2 = X2)

ggplot(data = melted, aes(x=Var1, y=Var2, fill=value)) +
  ggtitle("Developing Countries")+
  geom_tile(color = "white")+
 scale_fill_gradient2(low = "blue", high = "red", mid = "white",
   midpoint = 0, limit = c(-1,1), space = "Lab",
   name="Pearson \n Correlation") +
  theme_minimal()+
 theme(axis.text.x = element_text(angle = 45, vjust = 1,
    size = 10, hjust = 1))+
 coord_fixed()
```

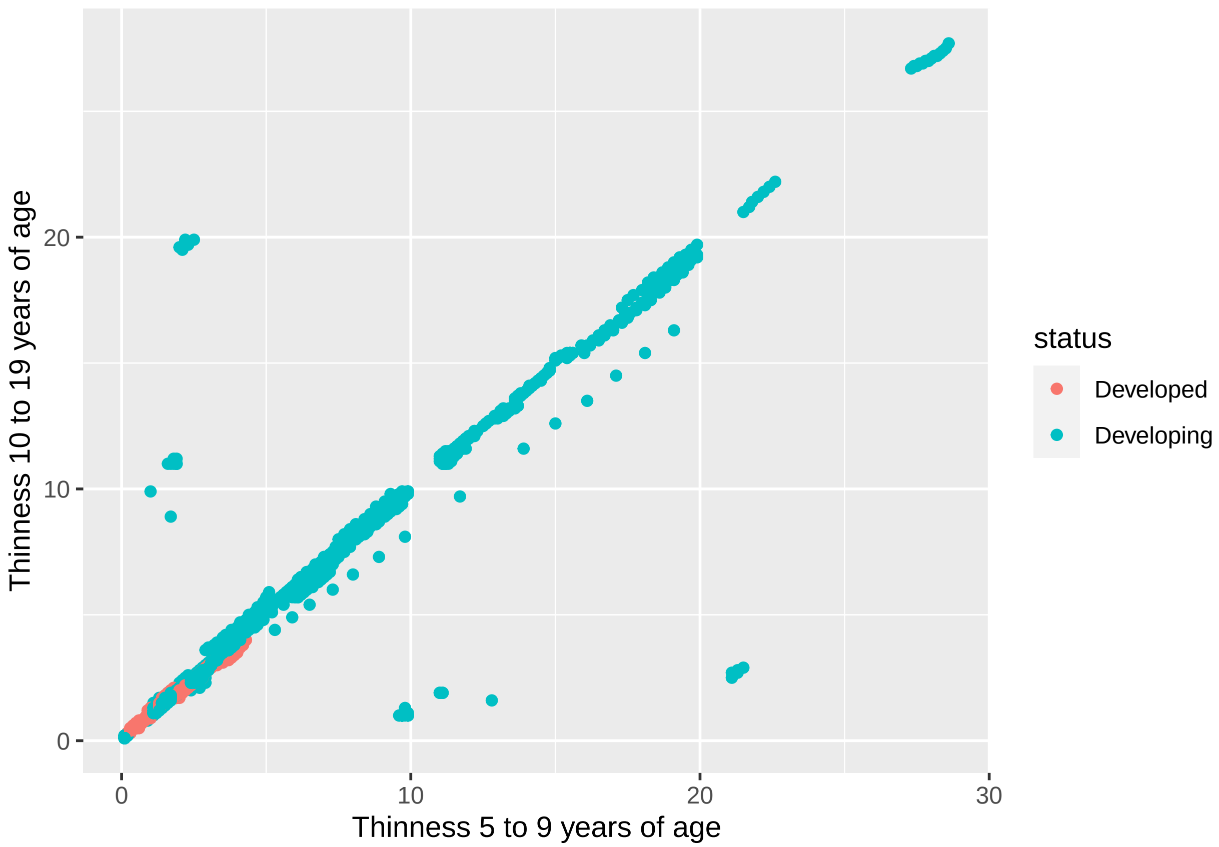


>All Countries::Life Expec High Corr: School, pct exp, gdp, dipth, polio, bmi, alcohol, tot exp, hepB Low Corr: hiv, thin5to9, thin10to19, under5,measles

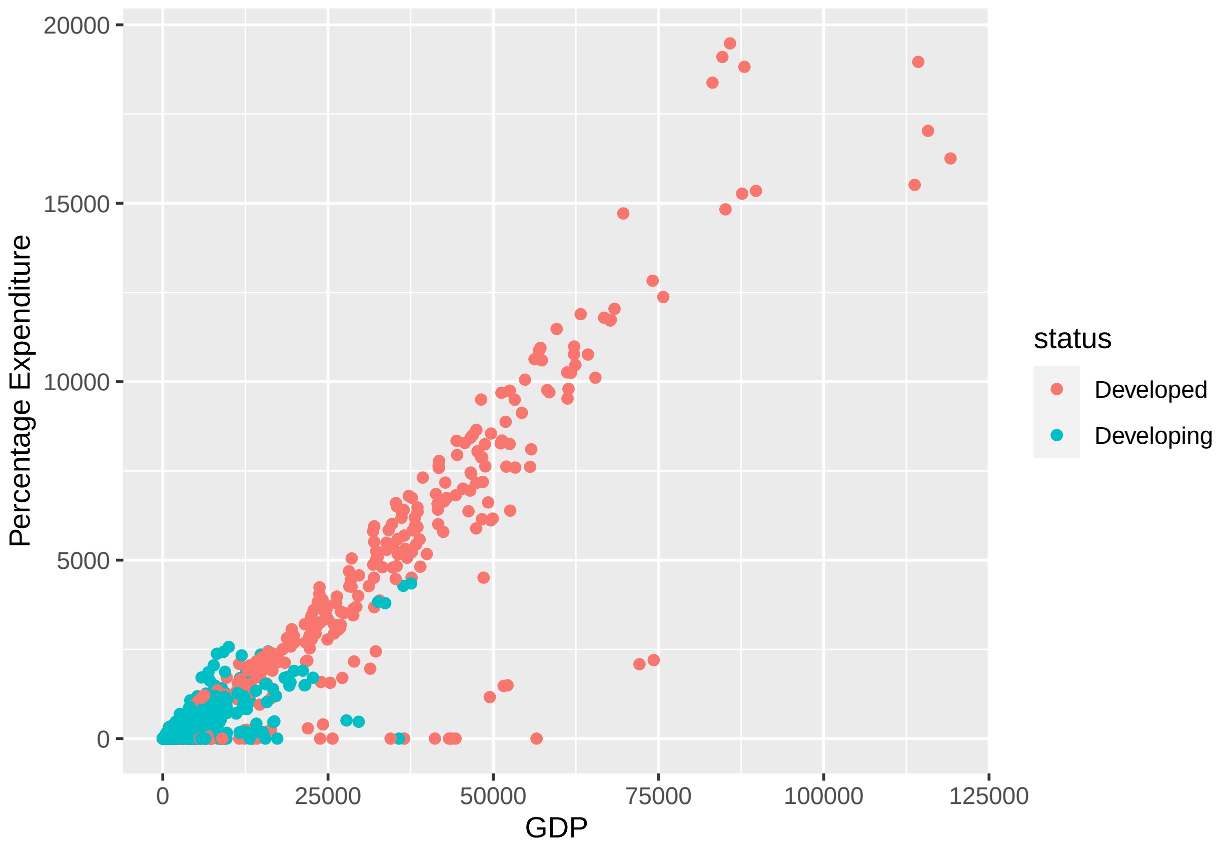
Not much differnet with Developed and Developing Countries

Plots of high correlations

```
ggplot(data = le_adj, aes(x = thin_5to9_years, y = thin_10to19_years, color=status))+
  geom_point() + labs(x = "Thinness 5 to 9 years of age", y = "Thinness 10 to 19 years of age")
```



```r
ggplot(data = le_adj, aes(x = gdp, y = percentage_expenditure, color=status))+
  geom_point() + labs(x = "GDP", y = "Percentage Expenditure")
```



> Will have to check for multicolinearity

> Use caret packages vif(). This calculates the variation inflation factors of all predictors in regression models, where high values are potentials to be dropped from the model.

removing indicator and categorical variables under different handles

```
le <- le_adj[,3:18]

le_ped <- le_developed[,3:18]

le_ping <- le_developing[,3:18]
```

##Finding the best method and variable selection through mean squared error (MSE)

```
set.seed(seed)
#all countries
train = le %>%
  sample_frac(0.7)
test = le %>%
  setdiff(train)
#developed countries
trained = le_ped %>%
  sample_frac(0.7)
tested = le_ped %>%
  setdiff(trained)
```

```
#developing countires
training = le_ping %>%
  sample_frac(0.7)
testing = le_ping %>%
  setdiff(training)
```

##Create a Baseline

```
#Mean only
base_MSE = mean((mean(train$life_expectancy)-test$life_expectancy)^2)
base_MSE
```

```
## [1] 98.22705
```

```
base_MSE1 = mean((mean(trained$life_expectancy)-tested$life_expectancy)^2)
base_MSE1
```

```
## [1] 16.3049
```

```
base_MSE2 = mean((mean(training$life_expectancy)-testing$life_expectancy)^2)
base_MSE2
```

```
## [1] 70.65962
```

##Ordinary Least Squares

```
lm = lm(life_expectancy ~., train)
summary(lm)
```

```
##
## Call:
## lm(formula = life_expectancy ~ ., data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.7763  -2.8147   0.0959   2.8157  19.2830
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)               4.529e+01  7.034e-01  64.383  < 2e-16 ***
## population               -1.326e-10  3.300e-09  -0.040   0.9680
## alcohol                  -1.583e-01  3.681e-02  -4.301 1.81e-05 ***
## hiv_aids                 -6.313e-01  2.097e-02 -30.111  < 2e-16 ***
## thin_5to9_years          -4.814e-02  6.973e-02  -0.690   0.4901
## thin_10to19_years        -3.763e-02  7.030e-02  -0.535   0.5925
## hepatitis_b              -9.831e-03  4.687e-03  -2.098   0.0361 *
## measles                   6.645e-07  1.105e-05   0.060   0.9521
## polio                     2.853e-02  6.566e-03   4.346 1.48e-05 ***
## diphtheria                3.934e-02  6.993e-03   5.625 2.20e-08 ***
## bmi                       6.368e-02  7.669e-03   8.303  < 2e-16 ***
## under_five_deaths       2.970e-04  9.657e-04   0.308   0.7585    
## total_expenditure       3.423e-02  5.118e-02   0.669   0.5037    
## gdp                     3.912e-05  2.540e-05   1.540   0.1236    
## percentage_expenditure  2.377e-04  1.527e-04   1.557   0.1198    
## schooling               1.519e+00  5.495e-02  27.651  < 2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
## 
## Residual standard error: 4.489 on 1563 degrees of freedom
## Multiple R-squared:  0.7895, Adjusted R-squared:  0.7875 
## F-statistic: 390.9 on 15 and 1563 DF,  p-value: < 2.2e-16
```

```
test = test %>%
  mutate(predictions = predict(lm, test))

slr_MSE_test = test %>%
  summarize(slr_MSE_test = mean((life_expectancy-predictions)^2))
slr_MSE_test
```

```
## # A tibble: 1 x 1
##   slr_MSE_test
##          <dbl>
## 1         19.7
```

```
lm1 = lm(life_expectancy ~., trained)
summary(lm1)
```

```
## 
## Call:
## lm(formula = life_expectancy ~ ., data = trained)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -4.7130 -1.9625 -0.4352  1.0678  9.4173 
## 
## Coefficients: (1 not defined because of singularities)
##                          Estimate Std. Error t value Pr(>|t|)    
## (Intercept)             7.731e+01  2.681e+00  28.836  < 2e-16 ***
## population             -4.491e-09  1.165e-08  -0.385 0.700194    
## alcohol                -2.832e-01  5.607e-02  -5.051 7.26e-07 ***
## hiv_aids                       NA         NA      NA       NA    
## thin_5to9_years        -4.584e-01  1.645e+00  -0.279 0.780642    
## thin_10to19_years      -2.636e+00  1.759e+00  -1.499 0.134926    
## hepatitis_b             5.012e-03  5.081e-03   0.986 0.324693    
## measles                 9.689e-06  5.681e-05   0.171 0.864670    
## polio                  -1.133e-03  2.219e-02  -0.051 0.959289    
## diphtheria              3.945e-02  2.047e-02   1.927 0.054805 .  
## bmi                    -1.514e-02  9.370e-03  -1.616 0.107057    
## under_five_deaths       3.557e-01  1.522e-01   2.337 0.020012 *  
## total_expenditure      -1.604e-01  5.939e-02  -2.701 0.007264 ** 
## gdp                     1.352e-05  1.676e-05   0.807 0.420478    
## percentage_expenditure  9.971e-05  9.746e-05   1.023 0.306999    
## schooling               3.755e-01  1.075e-01   3.491 0.000546 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.848 on 332 degrees of freedom
## Multiple R-squared:  0.5253, Adjusted R-squared:  0.5052
## F-statistic: 26.24 on 14 and 332 DF,  p-value: < 2.2e-16
```

```
tested = tested %>%
  mutate(predictions = predict(lm1, tested))
```

```
## Warning in predict.lm(lm1, tested): prediction from a rank-deficient fit may be
## misleading
```

```
slr_MSE_test1 = tested %>%
  summarize(slr_MSE_test1 = mean((life_expectancy-predictions)^2))
slr_MSE_test1
```

```
## # A tibble: 1 x 1
##   slr_MSE_test1
##           <dbl>
## 1          7.57
```

```
lm2 = lm(life_expectancy ~., training)
summary(lm2)
```

```
##
## Call:
## lm(formula = life_expectancy ~ ., data = training)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -25.5642  -2.8314   0.2078   3.1005  20.1826
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             4.582e+01  7.785e-01  58.853  < 2e-16 ***
## population              1.905e-09  2.477e-09   0.769 0.442145
## alcohol                -2.739e-01  4.867e-02  -5.628 2.26e-08 ***
## hiv_aids               -6.361e-01  2.156e-02 -29.499  < 2e-16 ***
## thin_5to9_years         9.182e-02  7.452e-02   1.232 0.218167
## thin_10to19_years      -1.055e-01  7.530e-02  -1.402 0.161319
## hepatitis_b             9.481e-03  6.800e-03   1.394 0.163501
## measles                 8.807e-06  1.048e-05   0.841 0.400775
## polio                   2.465e-02  6.943e-03   3.550 0.000400 ***
## diphtheria              3.479e-02  7.517e-03   4.628 4.08e-06 ***
## bmi                     9.056e-02  9.641e-03   9.393  < 2e-16 ***
## under_five_deaths      -4.743e-04  9.371e-04  -0.506 0.612841
## total_expenditure      -8.956e-02  6.675e-02  -1.342 0.179941
## gdp                     4.325e-05  6.023e-05   0.718 0.472795
## percentage_expenditure  2.170e-03  5.879e-04   3.691 0.000233 ***
## schooling               1.303e+00  6.775e-02  19.234  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.576 on 1216 degrees of freedom
## Multiple R-squared:  0.7448, Adjusted R-squared:  0.7416
## F-statistic: 236.6 on 15 and 1216 DF,  p-value: < 2.2e-16
```

```
testing = testing %>%
  mutate(predictions = predict(lm2, testing))

slr_MSE_test2 = testing %>%
  summarize(slr_MSE_test2 = mean((life_expectancy-predictions)^2))
slr_MSE_test2
```

```
## # A tibble: 1 x 1
##   slr_MSE_test2
##           <dbl>
## 1          18.8
```

Best Subsets

```
regfit_full = regsubsets(life_expectancy ~ ., data=train)
reg_summary = summary(regfit_full)
reg_summary
```

```
## Subset selection object
## Call: regsubsets.formula(life_expectancy ~ ., data = train)
## 15 Variables  (and intercept)
##                          Forced in Forced out
## population                   FALSE      FALSE
## alcohol                      FALSE      FALSE
## hiv_aids                     FALSE      FALSE
## thin_5to9_years              FALSE      FALSE
## thin_10to19_years            FALSE      FALSE
## hepatitis_b                  FALSE      FALSE
## measles                      FALSE      FALSE
## polio                        FALSE      FALSE
## diphtheria                   FALSE      FALSE
## bmi                          FALSE      FALSE
## under_five_deaths            FALSE      FALSE
## total_expenditure            FALSE      FALSE
## gdp                          FALSE      FALSE
## percentage_expenditure       FALSE      FALSE
## schooling                    FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: exhaustive
##          population alcohol hiv_aids thin_5to9_years thin_10to19_years
## 1  ( 1 ) " "        " "     " "      " "             " "
## 2  ( 1 ) " "        " "     "*"      " "             " "
## 3  ( 1 ) " "        " "     "*"      " "             " "
## 4  ( 1 ) " "        " "     "*"      " "             " "
## 5  ( 1 ) " "        " "     "*"      " "             " "
## 6  ( 1 ) " "        " "     "*"      " "             " "
## 7  ( 1 ) " "        "*"     "*"      " "             " "
## 8  ( 1 ) " "        "*"     "*"      "*"             " "
##          hepatitis_b measles polio diphtheria bmi under_five_deaths
## 1  ( 1 ) " "         " "     " "   " "        " " " "
## 2  ( 1 ) " "         " "     " "   " "        " " " "
## 3  ( 1 ) " "         " "     " "   " "        "*" " "
## 4  ( 1 ) " "         " "     " "   "*"        "*" " "
## 5  ( 1 ) " "         " "     " "   "*"        "*" " "
## 6  ( 1 ) " "         " "     "*"   "*"        "*" " "
## 7  ( 1 ) " "         " "     "*"   "*"        "*" " "
## 8  ( 1 ) " "         " "     "*"   "*"        "*" " "
##          total_expenditure gdp percentage_expenditure schooling
## 1  ( 1 ) " "               " " " "                    "*"
## 2  ( 1 ) " "               " " " "                    "*"
## 3  ( 1 ) " "               " " " "                    "*"
## 4  ( 1 ) " "               " " " "                    "*"
## 5  ( 1 ) " "               " " "*"                    "*"
## 6  ( 1 ) " "               " " "*"                    "*"
## 7  ( 1 ) " "               " " "*"                    "*"
## 8  ( 1 ) " "               " " "*"                    "*"
```

```
names(reg_summary)
```

```
## [1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat" "obj"
```

```
reg_summary$rsq
```

```
## [1] 0.5980878 0.7494978 0.7626662 0.7744410 0.7833097 0.7857222 0.7876044
## [8] 0.7884353
```

```
regfit_full1 = regsubsets(life_expectancy ~ ., data=trained)
```

```
## Warning in leaps.setup(x, y, wt = wt, nbest = nbest, nvmax = nvmax, force.in =
## force.in, : 1 linear dependencies found
```

```
## Reordering variables and trying again:
```

```
reg_summary1 = summary(regfit_full1)
reg_summary1
```

```
## Subset selection object
## Call: regsubsets.formula(life_expectancy ~ ., data = trained)
## 15 Variables  (and intercept)
##                         Forced in Forced out
## population                  FALSE      FALSE
## alcohol                     FALSE      FALSE
## thin_5to9_years             FALSE      FALSE
## thin_10to19_years           FALSE      FALSE
## hepatitis_b                 FALSE      FALSE
## measles                     FALSE      FALSE
## polio                     FALSE      FALSE
## diphtheria                FALSE      FALSE
## bmi                       FALSE      FALSE
## under_five_deaths         FALSE      FALSE
## total_expenditure         FALSE      FALSE
## gdp                       FALSE      FALSE
## percentage_expenditure    FALSE      FALSE
## schooling                 FALSE      FALSE
## hiv_aids                  FALSE      FALSE
## 1 subsets of each size up to 9
## Selection Algorithm: exhaustive
##          population alcohol hiv_aids thin_5to9_years thin_10to19_years
## 1  ( 1 ) " "        " "     " "      "*"             " "              
## 2  ( 1 ) " "        "*"     " "      "*"             " "              
## 3  ( 1 ) " "        "*"     " "      " "             "*"              
## 4  ( 1 ) " "        "*"     " "      " "             "*"              
## 5  ( 1 ) " "        "*"     " "      " "             "*"              
## 6  ( 1 ) " "        "*"     " "      " "             "*"              
## 7  ( 1 ) " "        "*"     " "      " "             "*"              
## 8  ( 1 ) " "        "*"     " "      " "             "*"              
## 9  ( 1 ) " "        "*"     " "      " "             "*"              
##          hepatitis_b measles polio diphtheria bmi under_five_deaths
## 1  ( 1 ) " "         " "     " "   " "        " " " "              
## 2  ( 1 ) " "         " "     " "   " "        " " " "              
## 3  ( 1 ) " "         " "     " "   " "        " " " "              
## 4  ( 1 ) " "         " "     " "   " "        " " "*"              
## 5  ( 1 ) " "         " "     " "   " "        " " "*"              
## 6  ( 1 ) " "         " "     " "   "*"        " " "*"              
## 7  ( 1 ) " "         " "     " "   "*"        " " "*"              
## 8  ( 1 ) " "         " "     " "   "*"        "*" "*"              
## 9  ( 1 ) "*"         " "     " "   "*"        "*" "*"              
##          total_expenditure gdp percentage_expenditure schooling
## 1  ( 1 ) " "               " " " "                    " "      
## 2  ( 1 ) " "               " " " "                    " "      
## 3  ( 1 ) " "               "*" " "                    " "      
## 4  ( 1 ) " "               "*" " "                    " "      
## 5  ( 1 ) " "               "*" " "                    "*"      
## 6  ( 1 ) " "               "*" " "                    "*"      
## 7  ( 1 ) "*"               " " "*"                    "*"      
## 8  ( 1 ) "*"               " " "*"                    "*"      
## 9  ( 1 ) "*"               " " "*"                    "*"      
```

```
names(reg_summary1)
```

```
## [1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat" "obj"
```

```
reg_summary1$rsq
```

```
## [1] 0.4213812 0.4562360 0.4760525 0.4902368 0.5024694 0.5096154 0.5189832
## [8] 0.5227800 0.5239252
```

```
regfit_full2 = regsubsets(life_expectancy ~ ., data=training)
reg_summary2 = summary(regfit_full2)
reg_summary2
```

```
## Subset selection object
## Call: regsubsets.formula(life_expectancy ~ ., data = training)
## 15 Variables  (and intercept)
##                        Forced in Forced out
## population                 FALSE      FALSE
## alcohol                    FALSE      FALSE
## hiv_aids                   FALSE      FALSE
## thin_5to9_years            FALSE      FALSE
## thin_10to19_years          FALSE      FALSE
## hepatitis_b                FALSE      FALSE
## measles                    FALSE      FALSE
## polio                      FALSE      FALSE
## diphtheria                 FALSE      FALSE
## bmi                        FALSE      FALSE
## under_five_deaths          FALSE      FALSE
## total_expenditure          FALSE      FALSE
## gdp                        FALSE      FALSE
## percentage_expenditure     FALSE      FALSE
## schooling                  FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: exhaustive
##          population alcohol hiv_aids thin_5to9_years thin_10to19_years
## 1  ( 1 ) " "        " "     " "      " "             " "              
## 2  ( 1 ) " "        " "     "*"      " "             " "              
## 3  ( 1 ) " "        " "     "*"      " "             " "              
## 4  ( 1 ) " "        " "     "*"      " "             " "              
## 5  ( 1 ) " "        " "     "*"      " "             " "              
## 6  ( 1 ) " "        "*"     "*"      " "             " "              
## 7  ( 1 ) " "        "*"     "*"      " "             " "              
## 8  ( 1 ) " "        "*"     "*"      " "             " "              
##          hepatitis_b measles polio diphtheria bmi under_five_deaths
## 1  ( 1 ) " "         " "     " "   " "        " " " "              
## 2  ( 1 ) " "         " "     " "   " "        " " " "              
## 3  ( 1 ) " "         " "     " "   " "        "*" " "              
## 4  ( 1 ) " "         " "     " "   "*"        "*" " "              
## 5  ( 1 ) " "         " "     " "   "*"        "*" " "              
## 6  ( 1 ) " "         " "     " "   "*"        "*" " "              
## 7  ( 1 ) " "         " "     "*"   "*"        "*" " "              
## 8  ( 1 ) " "         " "     "*"   "*"        "*" " "              
##          total_expenditure gdp percentage_expenditure schooling
## 1  ( 1 ) " "               " " " "                    "*"      
## 2  ( 1 ) " "               " " " "                    "*"      
## 3  ( 1 ) " "               " " " "                    "*"      
## 4  ( 1 ) " "               " " " "                    "*"      
## 5  ( 1 ) " "               " " "*"                    "*"      
## 6  ( 1 ) " "               " " "*"                    "*"      
## 7  ( 1 ) " "               " " "*"                    "*"      
## 8  ( 1 ) "*"               " " "*"                    "*"      
```

```
names(reg_summary2)
```

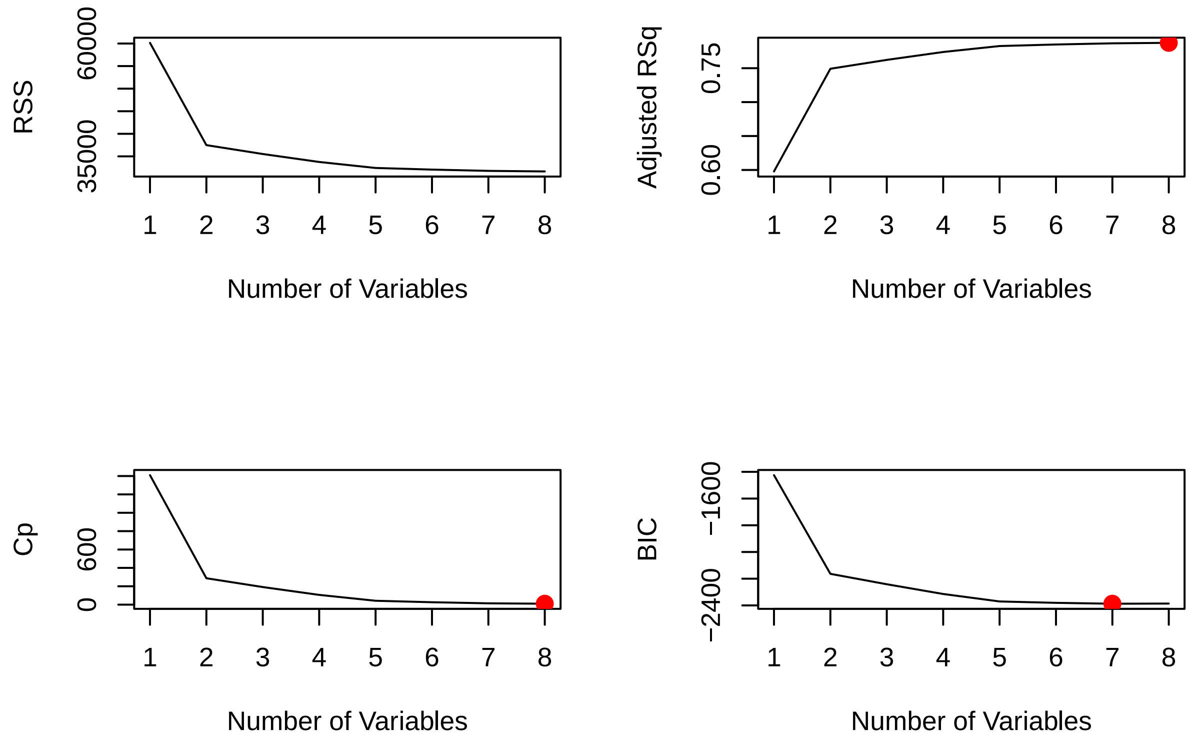
```
## [1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat" "obj"
```

```
reg_summary2$rsq
```

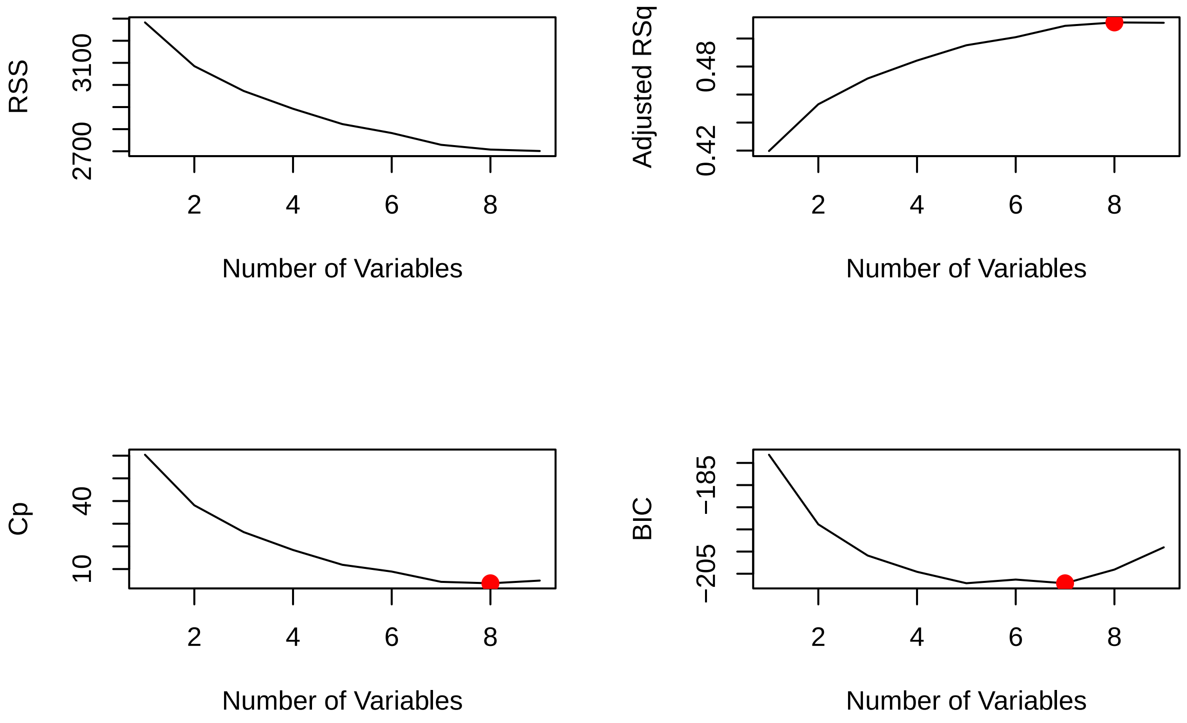
```
## [1] 0.4299386 0.6814867 0.7072204 0.7272066 0.7333712 0.7403756 0.7432554
## [8] 0.7437098
```

developed countries data set has issues with hiv_aids

```
par(mfrow = c(2,2))
plot(reg_summary$rss, xlab = "Number of Variables", ylab = "RSS", type = "l")

plot(reg_summary$adjr2, xlab = "Number of Variables", ylab = "Adjusted RSq", type = "l")
adj_r2_max = which.max(reg_summary$adjr2)
points(adj_r2_max, reg_summary$adjr2[adj_r2_max], col ="red", cex = 2, pch = 20)

plot(reg_summary$cp, xlab = "Number of Variables", ylab = "Cp", type = "l")
cp_min = which.min(reg_summary$cp)
points(cp_min, reg_summary$cp[cp_min], col = "red", cex = 2, pch = 20)

plot(reg_summary$bic, xlab = "Number of Variables", ylab = "BIC", type = "l")
bic_min = which.min(reg_summary$bic)
points(bic_min, reg_summary$bic[bic_min], col = "red", cex = 2, pch = 20)
```

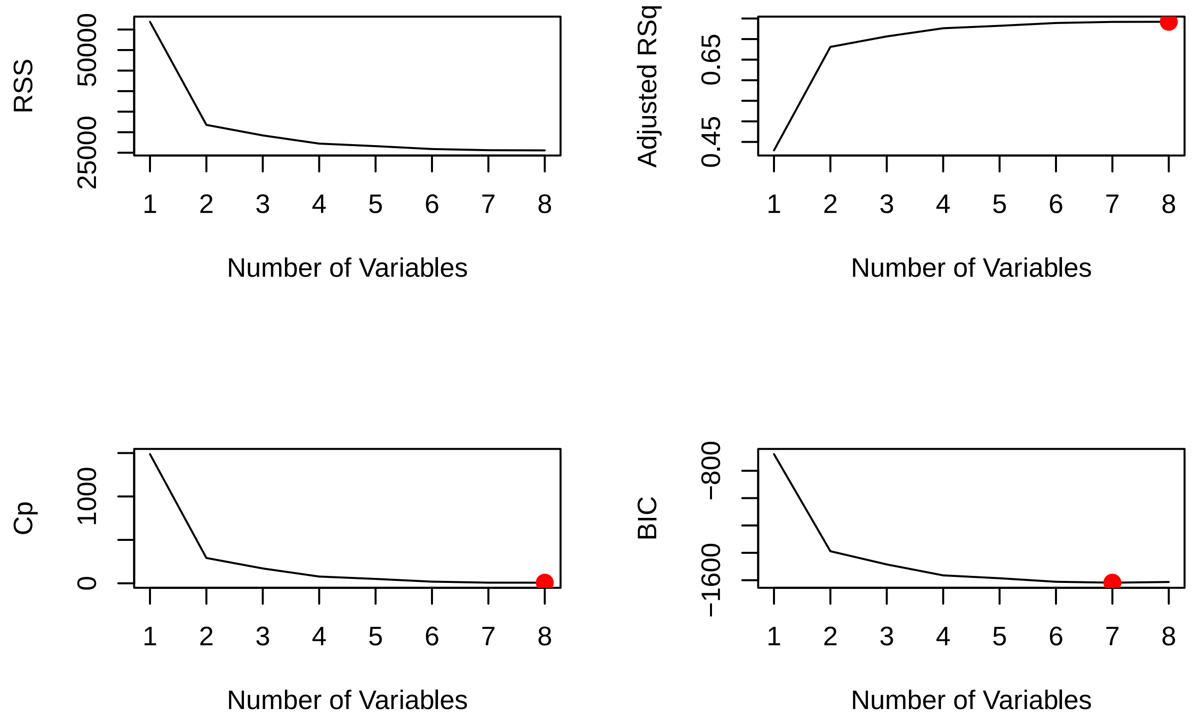


```r
par(mfrow = c(2,2))
plot(reg_summary1$rss, xlab = "Number of Variables", ylab = "RSS", type = "l")

plot(reg_summary1$adjr2, xlab = "Number of Variables", ylab = "Adjusted RSq", type = "l")
adj_r2_max = which.max(reg_summary1$adjr2)
points(adj_r2_max, reg_summary1$adjr2[adj_r2_max], col ="red", cex = 2, pch = 20)

plot(reg_summary1$cp, xlab = "Number of Variables", ylab = "Cp", type = "l")
cp_min = which.min(reg_summary1$cp)
points(cp_min, reg_summary1$cp[cp_min], col = "red", cex = 2, pch = 20)

plot(reg_summary1$bic, xlab = "Number of Variables", ylab = "BIC", type = "l")
bic_min = which.min(reg_summary1$bic)
points(bic_min, reg_summary1$bic[bic_min], col = "red", cex = 2, pch = 20)
```



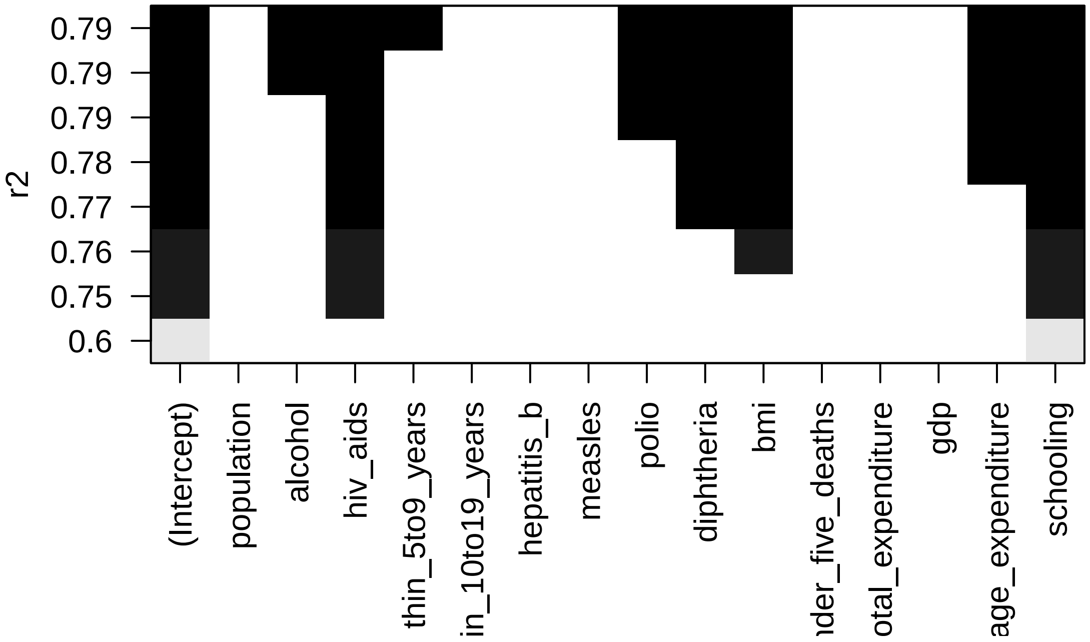
```r
par(mfrow = c(2,2))
plot(reg_summary2$rss, xlab = "Number of Variables", ylab = "RSS", type = "l")

plot(reg_summary2$adjr2, xlab = "Number of Variables", ylab = "Adjusted RSq", type = "l")
adj_r2_max = which.max(reg_summary2$adjr2)
points(adj_r2_max, reg_summary2$adjr2[adj_r2_max], col ="red", cex = 2, pch = 20)

plot(reg_summary2$cp, xlab = "Number of Variables", ylab = "Cp", type = "l")
cp_min = which.min(reg_summary2$cp)
points(cp_min, reg_summary2$cp[cp_min], col = "red", cex = 2, pch = 20)
```

```
plot(reg_summary2$bic, xlab = "Number of Variables", ylab = "BIC", type = "l")
bic_min = which.min(reg_summary2$bic)
points(bic_min, reg_summary2$bic[bic_min], col = "red", cex = 2, pch = 20)
```

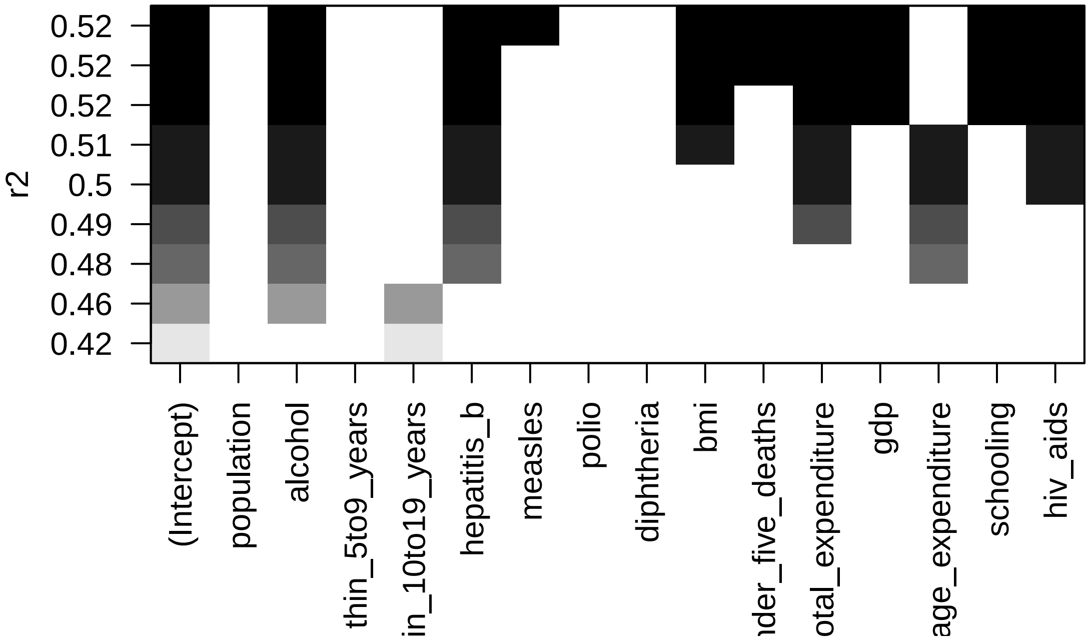


r squared

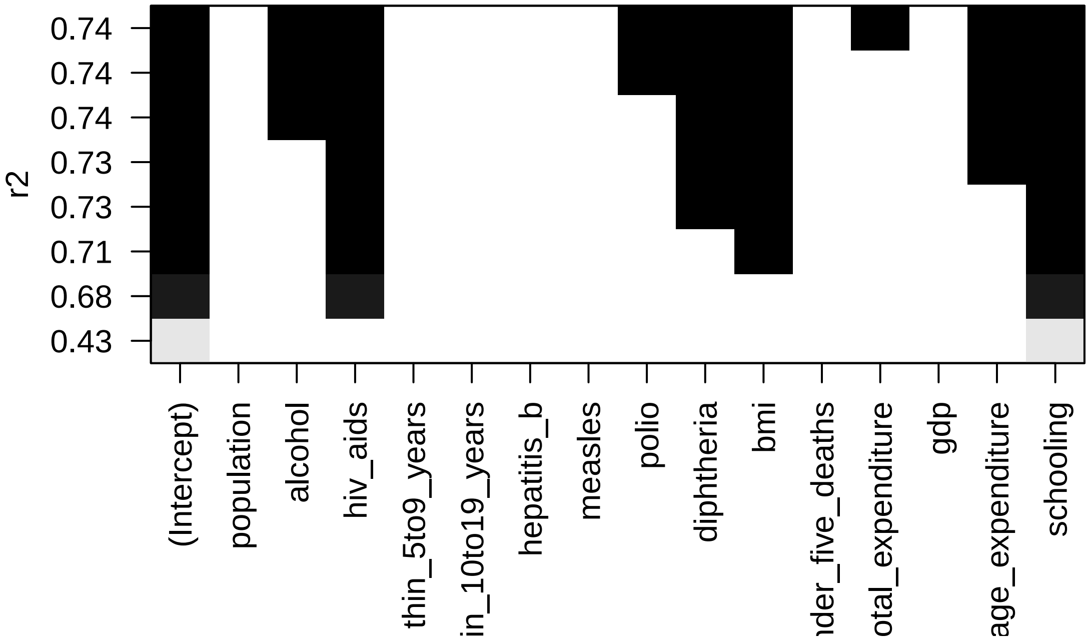
```
plot(regfit_full, scale="r2")
```



```
plot(regfit_full1, scale="r2")
```

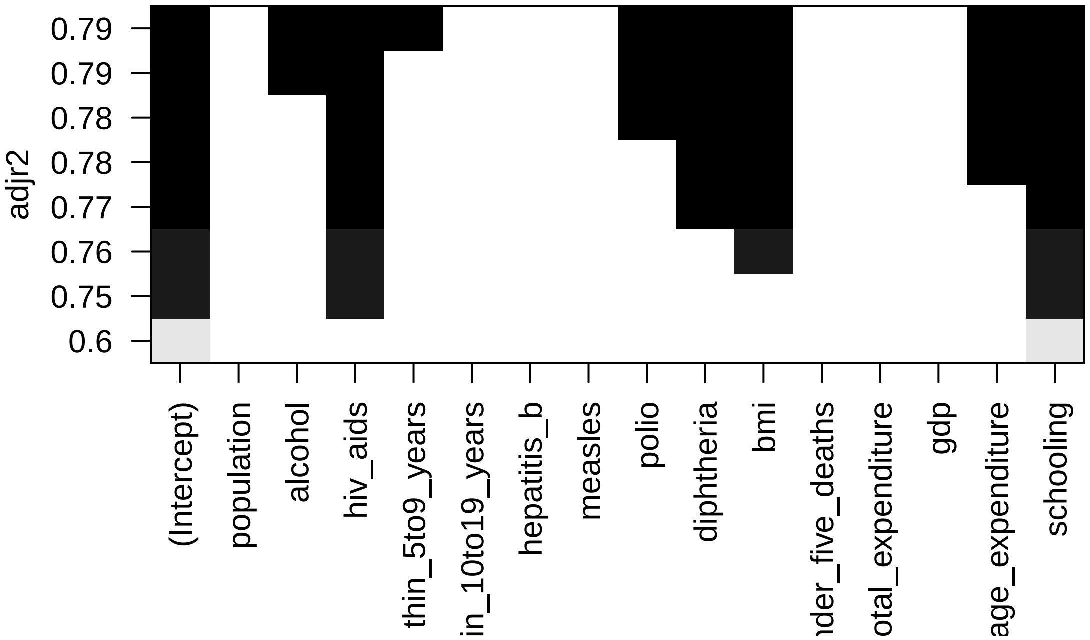


```r
plot(regfit_full2, scale="r2")
```



adj rsq

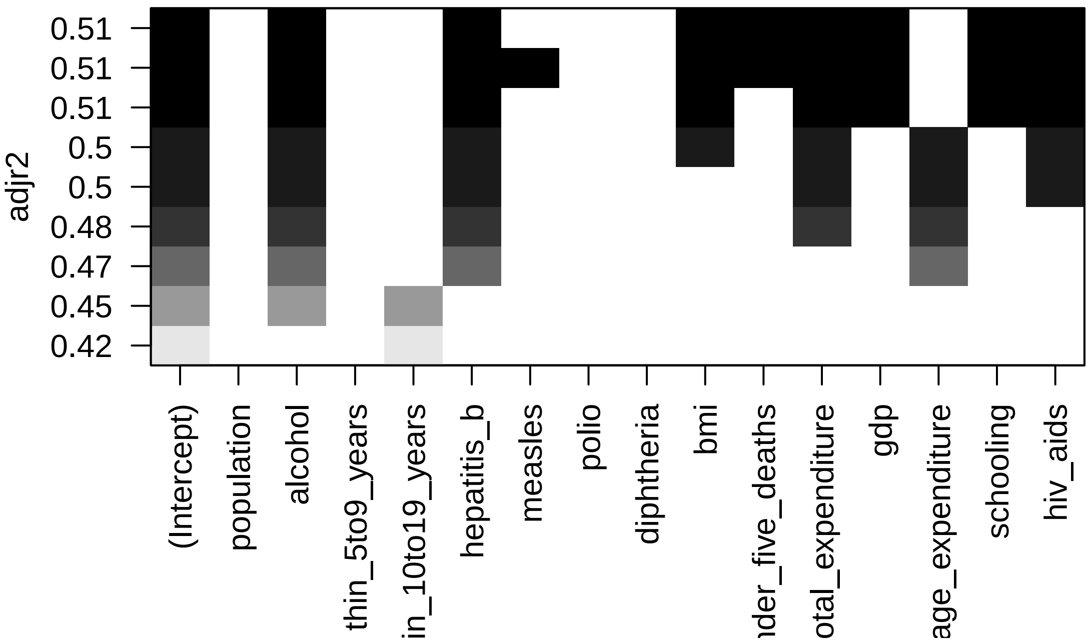
```
plot(regfit_full, scale="adjr2")
```



```
coef(regfit_full, 8)
```

```
##          (Intercept)             alcohol             hiv_aids
##        45.0278766390       -0.1473975161       -0.6312808710
##      thin_5to9_years               polio           diphtheria
##       -0.0761081980        0.0273868866         0.0343873185
##                  bmi percentage_expenditure           schooling
##         0.0632215140        0.0004926727         1.5371228552
```

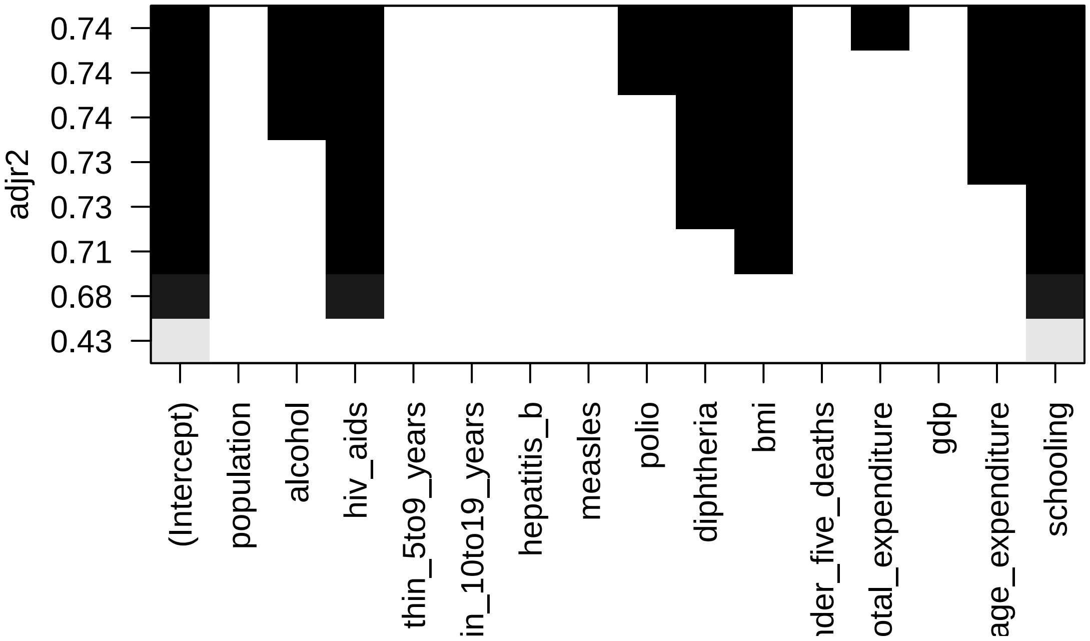
```
plot(regfit_full1, scale="adjr2")
```



```
coef(regfit_full1, 8)
```

```
##       (Intercept)           alcohol       hepatitis_b               bmi
##      7.043277e+01     -2.819576e-01     -1.107946e-02     -1.859388e-03
## under_five_deaths total_expenditure               gdp         schooling
##      3.236449e-01     -8.040935e-03      5.118661e-05      7.098157e-01
##          hiv_aids
##      0.000000e+00
```

```
plot(regfit_full2, scale="adjr2")
```



```
coef(regfit_full2, 8)
```

```
##             (Intercept)                 alcohol                hiv_aids 
##            45.789434309            -0.268768579            -0.638470640 
##                   polio              diphtheria                     bmi 
##             0.025883322             0.039694213             0.091405055 
##       total_expenditure percentage_expenditure               schooling 
##            -0.096561324             0.002474391             1.327160646
```

```
lm = lm(life_expectancy ~ alcohol + hiv_aids + thin_5to9_years + polio + diphtheria + bmi + percentage_
summary(lm)
```

```
## 
## Call:
## lm(formula = life_expectancy ~ alcohol + hiv_aids + thin_5to9_years + 
##     polio + diphtheria + bmi + percentage_expenditure + schooling, 
##     data = train)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -26.4836  -2.8253   0.1107   2.8523  19.5578 
## 
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            45.0278766  0.6531132  68.943  < 2e-16 ***
## alcohol                -0.1473975  0.0363213  -4.058 5.19e-05 ***
## hiv_aids               -0.6312809  0.0207406 -30.437  < 2e-16 ***
## thin_5to9_years        -0.0761082  0.0306505  -2.483   0.0131 *
## polio                   0.0273869  0.0065232   4.198 2.84e-05 ***
## diphtheria              0.0343873  0.0066050   5.206 2.18e-07 ***
## bmi                     0.0632215  0.0076152   8.302  < 2e-16 ***
## percentage_expenditure  0.0004927  0.0000574   8.583  < 2e-16 ***
## schooling               1.5371229  0.0539167  28.509  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.491 on 1570 degrees of freedom
## Multiple R-squared:  0.7884, Adjusted R-squared:  0.7874
## F-statistic: 731.4 on 8 and 1570 DF,  p-value: < 2.2e-16
```

```
test = test %>%
  mutate(predictions = predict(lm, test))

adj_MSE_test = test %>%
  summarize(adj_MSE_test = mean((life_expectancy-predictions)^2))
adj_MSE_test
```

```
## # A tibble: 1 x 1
##   adj_MSE_test
##          <dbl>
## 1         19.9
```

```
lm1 = lm(life_expectancy ~ alcohol + hepatitis_b + under_five_deaths + total_expenditure + gdp + bmi +
summary(lm1)
```

```
##
## Call:
## lm(formula = life_expectancy ~ alcohol + hepatitis_b + under_five_deaths +
##     total_expenditure + gdp + bmi + schooling + hiv_aids, data = trained)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.6947 -2.2779 -0.0853  1.7298 10.7562
##
## Coefficients: (1 not defined because of singularities)
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.043e+01  2.243e+00  31.404  < 2e-16 ***
## alcohol           -2.820e-01  6.720e-02  -4.196 3.48e-05 ***
## hepatitis_b       -1.108e-02  5.671e-03  -1.954   0.0516 .
## under_five_deaths  3.236e-01  1.500e-01   2.158   0.0316 *
## total_expenditure -8.041e-03  6.975e-02  -0.115   0.9083
## gdp                5.119e-05  8.815e-06   5.807 1.47e-08 ***
## bmi               -1.859e-03  1.107e-02  -0.168   0.8667
## schooling          7.098e-01  1.204e-01   5.896 9.00e-09 ***
## hiv_aids                  NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 3.463 on 339 degrees of freedom
## Multiple R-squared:  0.2835, Adjusted R-squared:  0.2688
## F-statistic: 19.17 on 7 and 339 DF,  p-value: < 2.2e-16
```

```
tested = tested %>%
  mutate(predictions = predict(lm1, tested))
```

```
## Warning in predict.lm(lm1, tested): prediction from a rank-deficient fit may be
## misleading
```

```
adj_MSE_test1 = tested %>%
  summarize(adj_MSE_test1 = mean((life_expectancy-predictions)^2))
adj_MSE_test1
```

```
## # A tibble: 1 x 1
##   adj_MSE_test1
##           <dbl>
## 1          11.9
```

```
lm2 = lm(life_expectancy ~ alcohol + hiv_aids + total_expenditure + polio + diphtheria + bmi + percentag
summary(lm2)
```

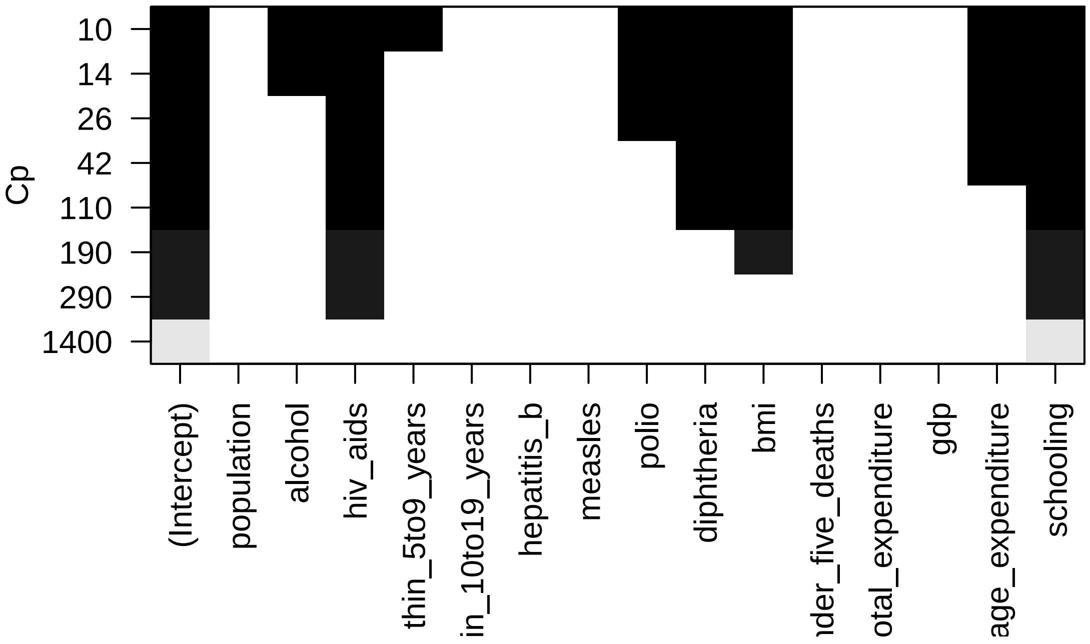
```
##
## Call:
## lm(formula = life_expectancy ~ alcohol + hiv_aids + total_expenditure +
##     polio + diphtheria + bmi + percentage_expenditure + schooling,
##     data = training)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -25.7404  -2.8381   0.2187   3.0102  20.2290
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            45.7894343  0.6753529  67.801  < 2e-16 ***
## alcohol                -0.2687686  0.0473362  -5.678 1.70e-08 ***
## hiv_aids               -0.6384706  0.0213132 -29.957  < 2e-16 ***
## total_expenditure      -0.0965613  0.0655734  -1.473 0.141125
## polio                   0.0258833  0.0068510   3.778 0.000166 ***
## diphtheria              0.0396942  0.0067626   5.870 5.62e-09 ***
## bmi                     0.0914051  0.0085619  10.676  < 2e-16 ***
## percentage_expenditure  0.0024744  0.0003976   6.223 6.68e-10 ***
## schooling               1.3271606  0.0657820  20.175  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.573 on 1223 degrees of freedom
## Multiple R-squared:  0.7437, Adjusted R-squared:  0.742
## F-statistic: 443.6 on 8 and 1223 DF,  p-value: < 2.2e-16
```

```
testing = testing %>%
  mutate(predictions = predict(lm2, testing))

adj_MSE_test2 = testing %>%
  summarize(adj_MSE_test2 = mean((life_expectancy-predictions)^2))
adj_MSE_test2
```

```
## # A tibble: 1 x 1
##   adj_MSE_test2
##           <dbl>
## 1          18.6
```

marrows cp

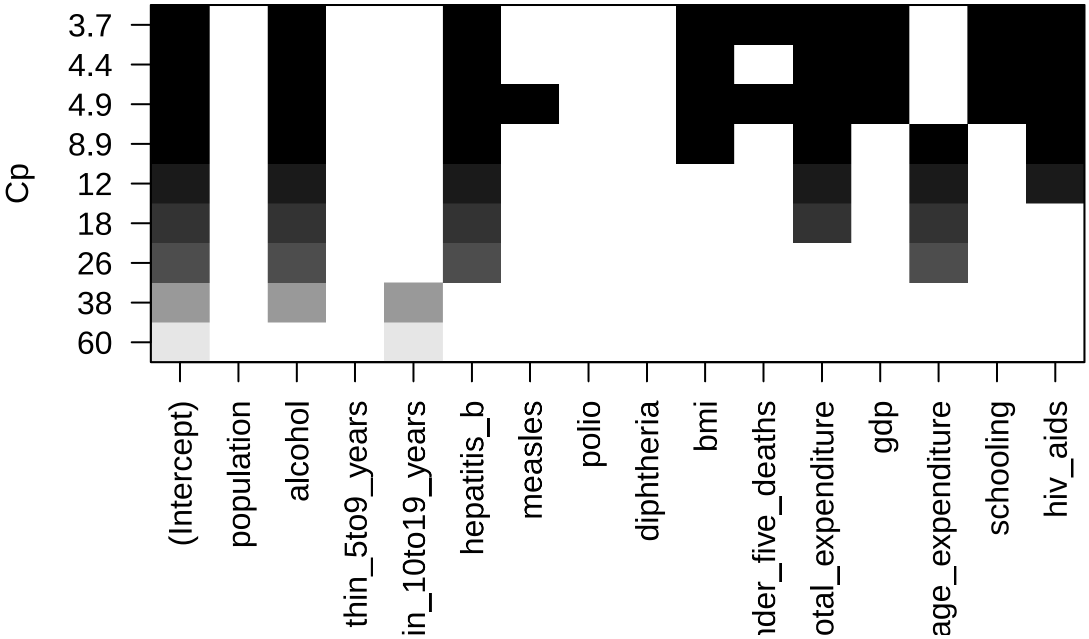
```
plot(regfit_full, scale="Cp")
```



```
coef(regfit_full, 8)
```

```
##           (Intercept)               alcohol              hiv_aids
##         45.0278766390         -0.1473975161         -0.6312808710
##       thin_5to9_years                 polio            diphtheria
##         -0.0761081980          0.0273868866          0.0343873185
##                   bmi percentage_expenditure             schooling
##          0.0632215140          0.0004926727          1.5371228552
```

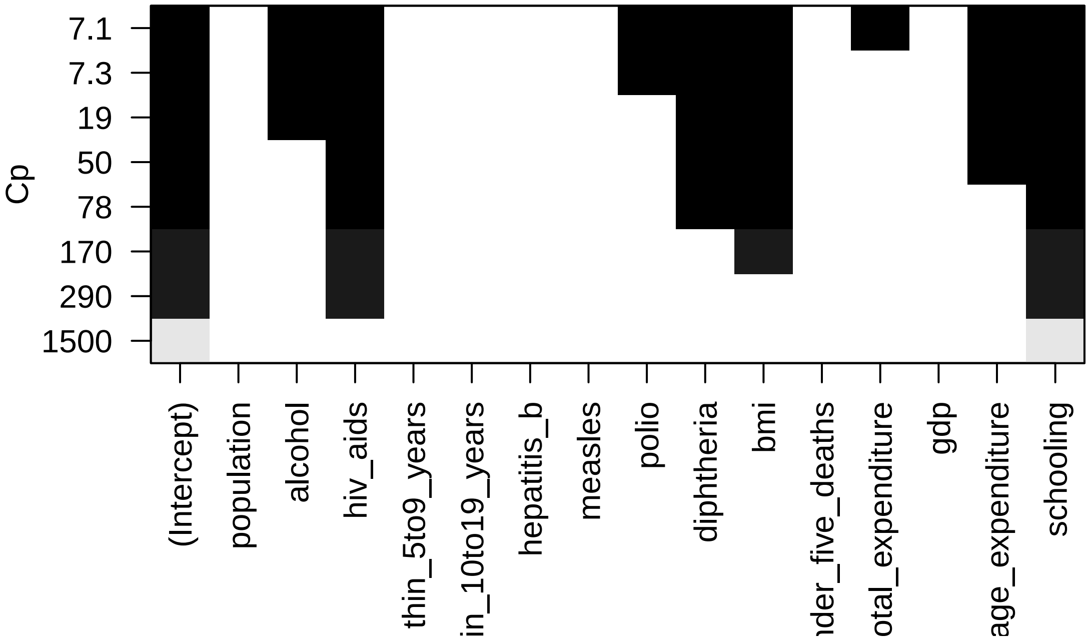
```
plot(regfit_full1, scale="Cp")
```



```
coef(regfit_full1, 8)
```

```
##       (Intercept)           alcohol       hepatitis_b               bmi 
##      7.043277e+01     -2.819576e-01     -1.107946e-02     -1.859388e-03 
## under_five_deaths total_expenditure               gdp         schooling 
##      3.236449e-01     -8.040935e-03      5.118661e-05      7.098157e-01 
##          hiv_aids 
##      0.000000e+00
```

```
plot(regfit_full2, scale="Cp")
```



```r
coef(regfit_full2, 8)
```

```
##           (Intercept)               alcohol              hiv_aids 
##          45.789434309          -0.268768579          -0.638470640 
##                 polio            diphtheria                   bmi 
##           0.025883322           0.039694213           0.091405055 
##     total_expenditure percentage_expenditure             schooling 
##          -0.096561324           0.002474391           1.327160646
```

```r
lm = lm(life_expectancy ~ alcohol + hiv_aids + thin_5to9_years + polio + diphtheria + bmi + percentage_e
summary(lm)
```

```
## 
## Call:
## lm(formula = life_expectancy ~ alcohol + hiv_aids + thin_5to9_years + 
##     polio + diphtheria + bmi + percentage_expenditure + schooling, 
##     data = train)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -26.4836  -2.8253   0.1107   2.8523  19.5578 
## 
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            45.0278766  0.6531132  68.943  < 2e-16 ***
## alcohol                -0.1473975  0.0363213  -4.058 5.19e-05 ***
## hiv_aids               -0.6312809  0.0207406 -30.437  < 2e-16 ***
## thin_5to9_years        -0.0761082  0.0306505  -2.483   0.0131 *
## polio                   0.0273869  0.0065232   4.198 2.84e-05 ***
## diphtheria              0.0343873  0.0066050   5.206 2.18e-07 ***
## bmi                     0.0632215  0.0076152   8.302  < 2e-16 ***
## percentage_expenditure  0.0004927  0.0000574   8.583  < 2e-16 ***
## schooling               1.5371229  0.0539167  28.509  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.491 on 1570 degrees of freedom
## Multiple R-squared:  0.7884, Adjusted R-squared:  0.7874
## F-statistic: 731.4 on 8 and 1570 DF,  p-value: < 2.2e-16
```

```
test = test %>%
  mutate(predictions = predict(lm, test))

Cp_MSE_test = test %>%
  summarize(Cp_MSE_test = mean((life_expectancy-predictions)^2))
Cp_MSE_test
```

```
## # A tibble: 1 x 1
##   Cp_MSE_test
##         <dbl>
## 1        19.9
```

```
lm1 = lm(life_expectancy ~ alcohol + hepatitis_b + under_five_deaths + total_expenditure + gdp + bmi + 
summary(lm1)
```

```
##
## Call:
## lm(formula = life_expectancy ~ alcohol + hepatitis_b + under_five_deaths +
##     total_expenditure + gdp + bmi + schooling + hiv_aids, data = trained)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.6947 -2.2779 -0.0853  1.7298 10.7562
##
## Coefficients: (1 not defined because of singularities)
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.043e+01  2.243e+00  31.404  < 2e-16 ***
## alcohol           -2.820e-01  6.720e-02  -4.196 3.48e-05 ***
## hepatitis_b       -1.108e-02  5.671e-03  -1.954   0.0516 .
## under_five_deaths  3.236e-01  1.500e-01   2.158   0.0316 *
## total_expenditure -8.041e-03  6.975e-02  -0.115   0.9083
## gdp                5.119e-05  8.815e-06   5.807 1.47e-08 ***
## bmi               -1.859e-03  1.107e-02  -0.168   0.8667
## schooling          7.098e-01  1.204e-01   5.896 9.00e-09 ***
## hiv_aids                  NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 3.463 on 339 degrees of freedom
## Multiple R-squared:  0.2835, Adjusted R-squared:  0.2688
## F-statistic: 19.17 on 7 and 339 DF,  p-value: < 2.2e-16
```

```
tested = tested %>%
  mutate(predictions = predict(lm1, tested))
```

```
## Warning in predict.lm(lm1, tested): prediction from a rank-deficient fit may be
## misleading
```

```
Cp_MSE_test1 = tested %>%
  summarize(Cp_MSE_test1 = mean((life_expectancy-predictions)^2))
Cp_MSE_test1
```

```
## # A tibble: 1 x 1
##   Cp_MSE_test1
##          <dbl>
## 1         11.9
```

```
lm2 = lm(life_expectancy ~ alcohol + hiv_aids + total_expenditure + polio + diphtheria + bmi + percentag
summary(lm2)
```

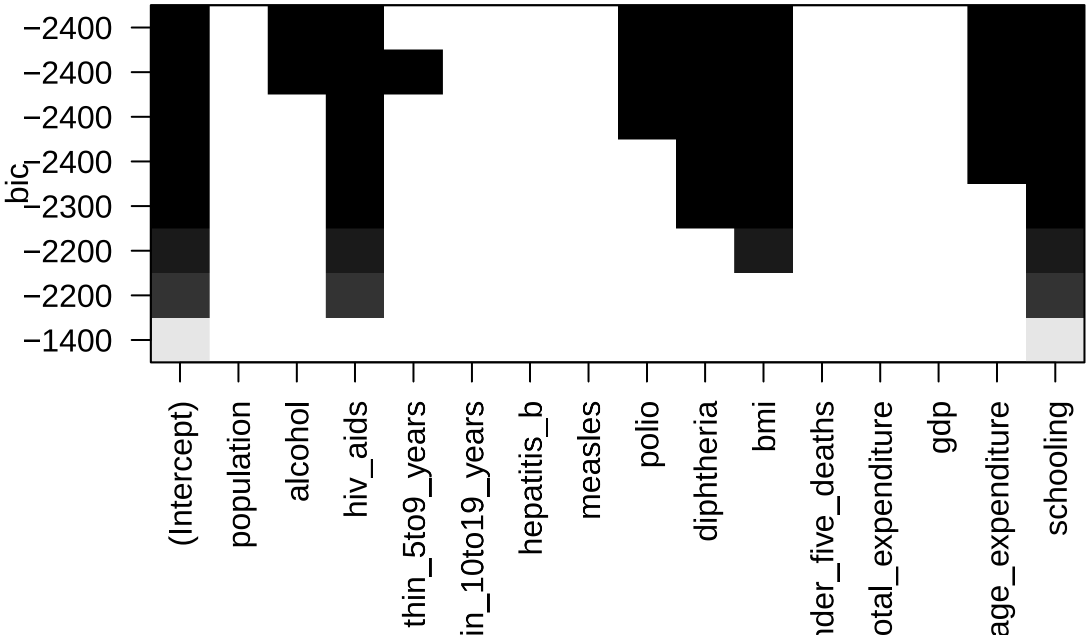
```
##
## Call:
## lm(formula = life_expectancy ~ alcohol + hiv_aids + total_expenditure +
##     polio + diphtheria + bmi + percentage_expenditure + schooling,
##     data = training)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -25.7404  -2.8381   0.2187   3.0102  20.2290
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            45.7894343  0.6753529  67.801  < 2e-16 ***
## alcohol                -0.2687686  0.0473362  -5.678 1.70e-08 ***
## hiv_aids               -0.6384706  0.0213132 -29.957  < 2e-16 ***
## total_expenditure      -0.0965613  0.0655734  -1.473 0.141125
## polio                   0.0258833  0.0068510   3.778 0.000166 ***
## diphtheria              0.0396942  0.0067626   5.870 5.62e-09 ***
## bmi                     0.0914051  0.0085619  10.676  < 2e-16 ***
## percentage_expenditure  0.0024744  0.0003976   6.223 6.68e-10 ***
## schooling               1.3271606  0.0657820  20.175  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.573 on 1223 degrees of freedom
## Multiple R-squared:  0.7437, Adjusted R-squared:  0.742
## F-statistic: 443.6 on 8 and 1223 DF,  p-value: < 2.2e-16
```

```
testing = testing %>%
  mutate(predictions = predict(lm2, testing))

Cp_MSE_test2 = testing %>%
  summarize(Cp_MSE_test2 = mean((life_expectancy-predictions)^2))
Cp_MSE_test2
```

```
## # A tibble: 1 x 1
##   Cp_MSE_test2
##          <dbl>
## 1         18.6
```

bic

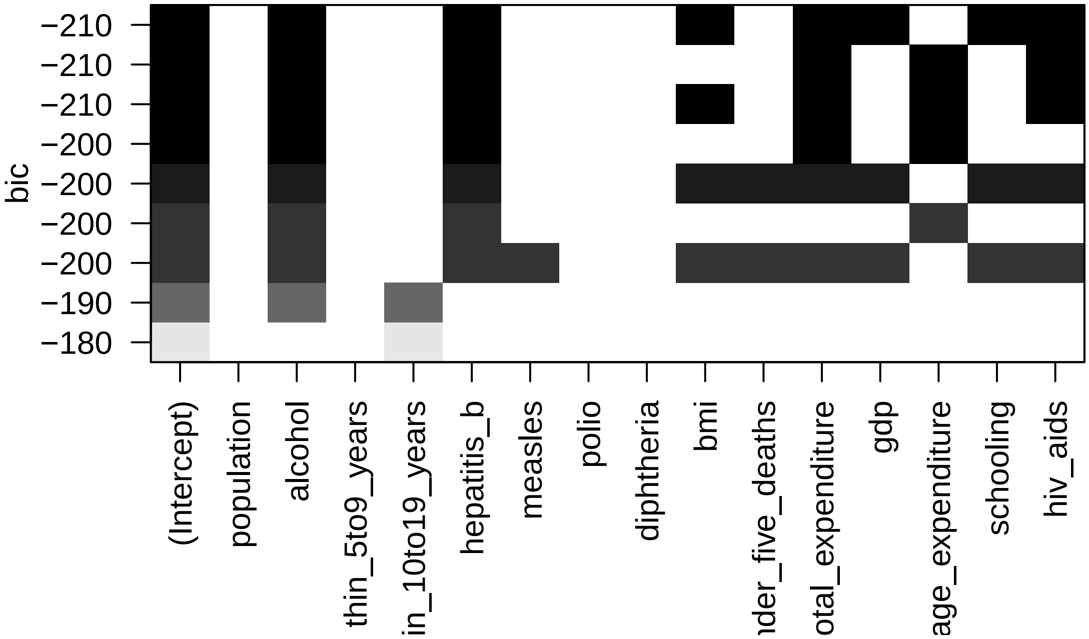
```
plot(regfit_full, scale="bic")
```



```
coef(regfit_full, 7)
```

```
##           (Intercept)               alcohol              hiv_aids
##         44.1665682016         -0.1343105472         -0.6353755038
##                 polio            diphtheria                   bmi
##          0.0276687197          0.0342678640          0.0702706739
## percentage_expenditure             schooling
##          0.0005015473          1.5500986521
```

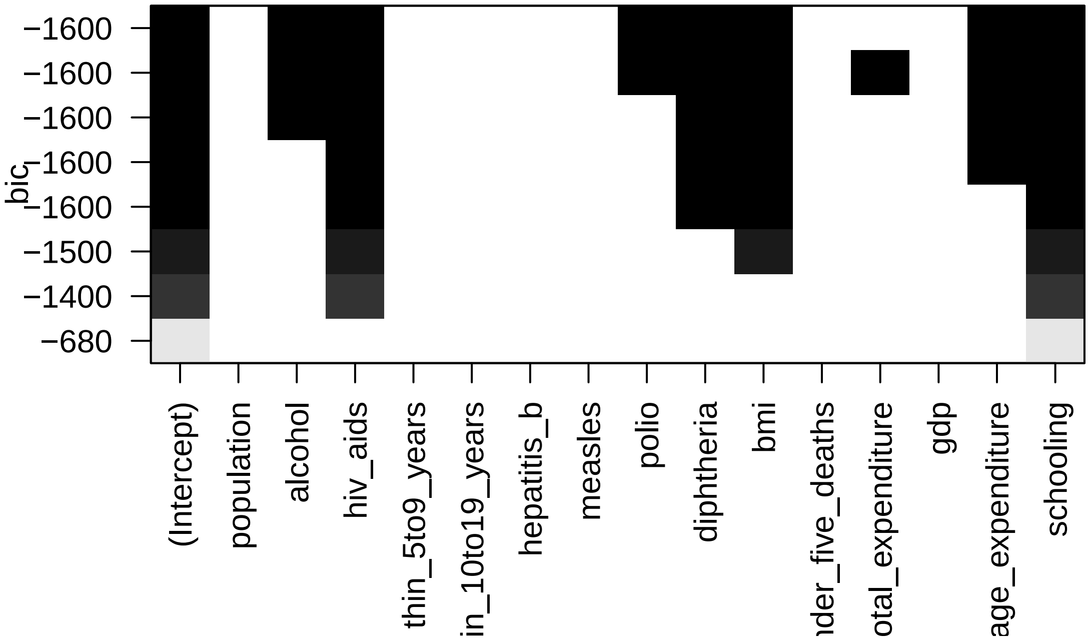
```
plot(regfit_full1, scale="bic")
```



```
coef(regfit_full1, 7)
```

```
##       (Intercept)           alcohol       hepatitis_b               bmi
##      7.153540e+01     -2.881779e-01     -1.250888e-02     -4.930307e-03
## total_expenditure               gdp         schooling          hiv_aids
##     -2.717757e-02      4.912535e-05      6.913745e-01      0.000000e+00
```

```
plot(regfit_full2, scale="bic")
```



```
coef(regfit_full2, 7)
```

```
##             (Intercept)                 alcohol                hiv_aids 
##            45.380753118            -0.270902825            -0.641698147 
##                   polio              diphtheria                     bmi 
##             0.025363314             0.039163343             0.089645622 
## percentage_expenditure               schooling 
##             0.002403875             1.329845242
```

```
lm = lm(life_expectancy ~ alcohol + hiv_aids + polio + diphtheria + bmi + percentage_expenditure + scho
summary(lm)
```

```
## 
## Call:
## lm(formula = life_expectancy ~ alcohol + hiv_aids + polio + diphtheria + 
##     bmi + percentage_expenditure + schooling, data = train)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -26.4286  -2.7916   0.0404   2.8754  19.8353 
## 
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)    
## (Intercept)             4.417e+01  5.543e-01  79.680  < 2e-16 ***
## alcohol                -1.343e-01  3.600e-02  -3.731 0.000197 ***
## hiv_aids               -6.354e-01  2.071e-02 -30.681  < 2e-16 ***
## polio                   2.767e-02  6.533e-03   4.235 2.42e-05 ***
## diphtheria              3.427e-02  6.616e-03   5.180 2.51e-07 ***
## bmi                     7.027e-02  7.078e-03   9.928  < 2e-16 ***
## percentage_expenditure  5.016e-04  5.738e-05   8.740  < 2e-16 ***
## schooling               1.550e+00  5.375e-02  28.839  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.498 on 1571 degrees of freedom
## Multiple R-squared:  0.7876, Adjusted R-squared:  0.7867
## F-statistic: 832.2 on 7 and 1571 DF,  p-value: < 2.2e-16
```

```
test = test %>%
  mutate(predictions = predict(lm, test))

bic_MSE_test = test %>%
  summarize(bic_MSE_test = mean((life_expectancy-predictions)^2))
bic_MSE_test
```

```
## # A tibble: 1 x 1
##   bic_MSE_test
##          <dbl>
## 1         20.0
```

```
lm1 = lm(life_expectancy ~ alcohol + hepatitis_b + total_expenditure + gdp + bmi + schooling + hiv_aids
summary(lm1)
```

```
##
## Call:
## lm(formula = life_expectancy ~ alcohol + hepatitis_b + total_expenditure +
##     gdp + bmi + schooling + hiv_aids, data = trained)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.0603 -2.3913 -0.1633  1.7966 11.1581
##
## Coefficients: (1 not defined because of singularities)
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.154e+01  2.196e+00  32.582  < 2e-16 ***
## alcohol           -2.882e-01  6.750e-02  -4.269 2.55e-05 ***
## hepatitis_b       -1.251e-02  5.662e-03  -2.209   0.0278 *
## total_expenditure -2.718e-02  6.955e-02  -0.391   0.6962
## gdp                4.913e-05  8.810e-06   5.576 5.03e-08 ***
## bmi               -4.930e-03  1.103e-02  -0.447   0.6553
## schooling          6.914e-01  1.207e-01   5.727 2.25e-08 ***
## hiv_aids                  NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.481 on 340 degrees of freedom
## Multiple R-squared:  0.2737, Adjusted R-squared:  0.2609
## F-statistic: 21.35 on 6 and 340 DF,  p-value: < 2.2e-16
```

```
tested = tested %>%
  mutate(predictions = predict(lm1, tested))
```

```
## Warning in predict.lm(lm1, tested): prediction from a rank-deficient fit may be
## misleading
```

```
bic_MSE_test1 = tested %>%
  summarize(bic_MSE_test1 = mean((life_expectancy-predictions)^2))
bic_MSE_test1
```

```
## # A tibble: 1 x 1
##   bic_MSE_test1
##           <dbl>
## 1          12.0
```

```
lm2 = lm(life_expectancy ~ alcohol + hiv_aids +  polio + diphtheria + bmi + percentage_expenditure + sch
summary(lm2)
```

```
##
## Call:
## lm(formula = life_expectancy ~ alcohol + hiv_aids + polio + diphtheria +
##     bmi + percentage_expenditure + schooling, data = training)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.0461  -2.9064   0.2041   3.0429  20.0688
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            45.3807531  0.6159877  73.672  < 2e-16 ***
## alcohol                -0.2709028  0.0473366  -5.723 1.32e-08 ***
## hiv_aids               -0.6416981  0.0212103 -30.254  < 2e-16 ***
## polio                   0.0253633  0.0068451   3.705 0.000221 ***
## diphtheria              0.0391633  0.0067562   5.797 8.60e-09 ***
## bmi                     0.0896456  0.0084821  10.569  < 2e-16 ***
## percentage_expenditure  0.0024039  0.0003949   6.087 1.53e-09 ***
## schooling               1.3298452  0.0657881  20.214  < 2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 4.575 on 1224 degrees of freedom
## Multiple R-squared:  0.7433, Adjusted R-squared:  0.7418
## F-statistic: 506.2 on 7 and 1224 DF,  p-value: < 2.2e-16
```

```
testing = testing %>%
  mutate(predictions = predict(lm2, testing))

bic_MSE_test2 = testing %>%
  summarize(bic_MSE_test2 = mean((life_expectancy-predictions)^2))
bic_MSE_test2
```

```
## # A tibble: 1 x 1
##   bic_MSE_test2
##           <dbl>
## 1          18.4
```

```
test = test[,1:16]
tested = tested[,1:16]
testing = testing[,1:16]
```

Best Subsets with CV

```
regfit_best_train = regsubsets(life_expectancy ~ ., data=train, nvmax = 15)
summary(regfit_best_train)
```

```
## Subset selection object
## Call: regsubsets.formula(life_expectancy ~ ., data = train, nvmax = 15)
## 15 Variables  (and intercept)
##                        Forced in Forced out
## population                 FALSE      FALSE
## alcohol                    FALSE      FALSE
## hiv_aids                   FALSE      FALSE
## thin_5to9_years            FALSE      FALSE
## thin_10to19_years          FALSE      FALSE
## hepatitis_b                FALSE      FALSE
## measles                    FALSE      FALSE
## polio                      FALSE      FALSE
## diphtheria                 FALSE      FALSE
## bmi                        FALSE      FALSE
## under_five_deaths          FALSE      FALSE
## total_expenditure          FALSE      FALSE
## gdp                        FALSE      FALSE
## percentage_expenditure     FALSE      FALSE
## schooling                  FALSE      FALSE
## 1 subsets of each size up to 15
## Selection Algorithm: exhaustive
##           population alcohol hiv_aids thin_5to9_years thin_10to19_years
## 1  ( 1 )  " "        " "     " "      " "             " "              
## 2  ( 1 )  " "        " "     "*"      " "             " "              
## 3  ( 1 )  " "        " "     "*"      " "             " "              
## 4  ( 1 )  " "        " "     "*"      " "             " "              
## 5  ( 1 )  " "        " "     "*"      " "             " "              
## 6  ( 1 )  " "        " "     "*"      " "             " "              
## 7  ( 1 )  " "        "*"     "*"      " "             " "              
## 8  ( 1 )  " "        "*"     "*"      "*"             " "              
## 9  ( 1 )  " "        "*"     "*"      "*"             " "              
## 10  ( 1 ) " "        "*"     "*"      "*"             " "              
## 11  ( 1 ) " "        "*"     "*"      "*"             " "              
## 12  ( 1 ) " "        "*"     "*"      "*"             "*"              
## 13  ( 1 ) " "        "*"     "*"      "*"             "*"              
## 14  ( 1 ) " "        "*"     "*"      "*"             "*"              
## 15  ( 1 ) "*"        "*"     "*"      "*"             "*"
##           hepatitis_b measles polio diphtheria bmi under_five_deaths
## 1  ( 1 )  " "         " "     " "   " "        " " " "
## 2  ( 1 )  " "         " "     " "   " "        " " " "
## 3  ( 1 )  " "         " "     " "   " "        "*" " "
## 4  ( 1 )  " "         " "     " "   "*"        "*" " "
## 5  ( 1 )  " "         " "     " "   "*"        "*" " "
## 6  ( 1 )  " "         " "     "*"   "*"        "*" " "
## 7  ( 1 )  " "         " "     "*"   "*"        "*" " "
## 8  ( 1 )  " "         " "     "*"   "*"        "*" " "
## 9  ( 1 )  "*"         " "     "*"   "*"        "*" " "
## 10  ( 1 ) "*"         " "     "*"   "*"        "*" " "
## 11  ( 1 ) "*"         " "     "*"   "*"        "*" " "
## 12  ( 1 ) "*"         " "     "*"   "*"        "*" " "
## 13  ( 1 ) "*"         " "     "*"   "*"        "*" "*"
## 14  ( 1 ) "*"         "*"     "*"   "*"        "*" "*"
## 15  ( 1 ) "*"         "*"     "*"   "*"        "*" "*"
##           total_expenditure gdp percentage_expenditure schooling
## 1  ( 1 )  " "               " " " "                    "*"
## 2  ( 1 )  " "               " " " "                    "*"
## 3  ( 1 )  " "               " " " "                    "*"
## 4  ( 1 )  " "               " " " "                    "*"
## 5  ( 1 )  " "               " " "*"                    "*"
## 6  ( 1 )  " "               " " "*"                    "*"
## 7  ( 1 )  " "               " " "*"                    "*"
## 8  ( 1 )  " "               " " "*"                    "*"
## 9  ( 1 )  " "               " " "*"                    "*"
## 10  ( 1 ) " "               "*" "*"                    "*"
## 11  ( 1 ) "*"               "*" "*"                    "*"
## 12  ( 1 ) "*"               "*" "*"                    "*"
## 13  ( 1 ) "*"               "*" "*"                    "*"
## 14  ( 1 ) "*"               "*" "*"                    "*"
## 15  ( 1 ) "*"               "*" "*"                    "*"
```

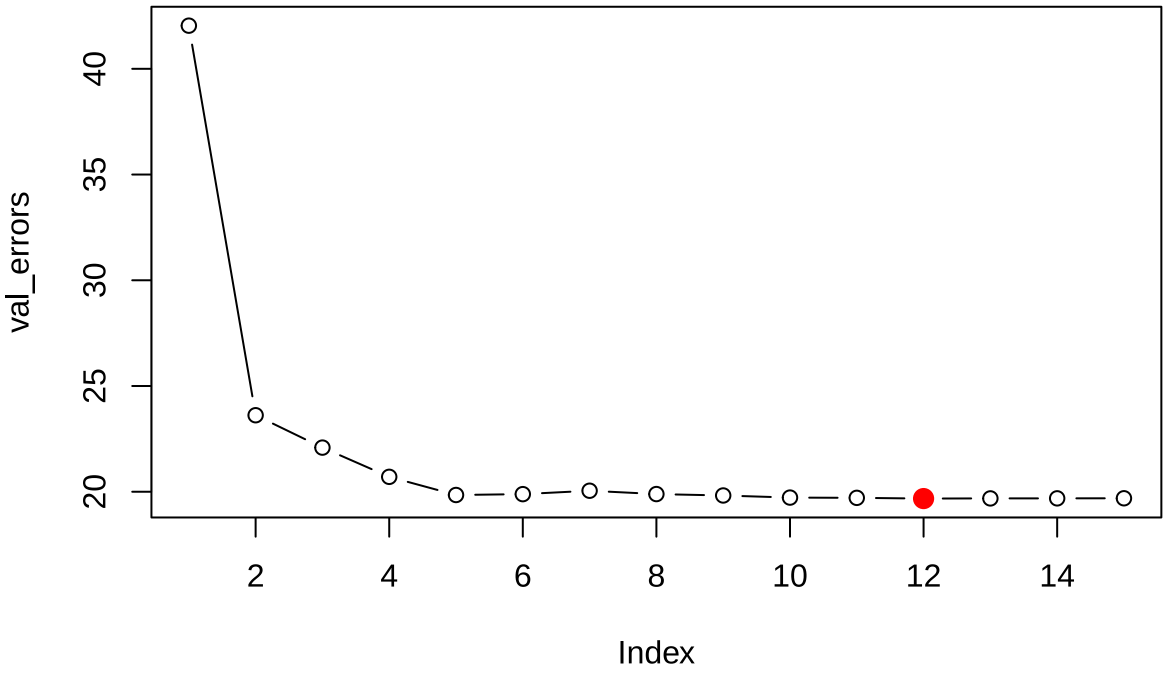
```
test_mat = model.matrix (life_expectancy~., data = test)
```

```
val_errors = rep(NA,15)

# Iterate over each size i
for(i in 1:15){

    # Extract the vector of predictors in the best fit model on i predictors
    coefi = coef(regfit_best_train, id = i)

    # Make predictions using matrix multiplication of the test matirx and the coefficients vector
    pred = test_mat[,names(coefi)]%*%coefi

    # Calculate the MSE
    val_errors[i] = mean((test$life_expectancy-pred)^2)
}
```

```
# Find the model with the smallest error
min = which.min(val_errors)
min
```

```
## [1] 12
```

```
# Plot the errors for each model size
plot(val_errors, type = 'b')
points(min, val_errors[min][1], col = "red", cex = 2, pch = 20)
```



```
#Creating a predict function for regsubsets
predict.regsubsets = function(object,newdata,id,...){
      form = as.formula(object$call[[2]])
      mat = model.matrix(form,newdata)
      coefi = coef(object,id=id)
      xvars = names(coefi)
      mat[,xvars]%*%coefi
}
```

```
regfit_best = regsubsets(life_expectancy~., data = train, nvmax = 15)
coef(regfit_best_train, 12)
```

```
##           (Intercept)               alcohol              hiv_aids
##          4.530151e+01         -1.567614e-01         -6.321580e-01
##        thin_5to9_years      thin_10to19_years           hepatitis_b
##         -4.474862e-02         -3.503033e-02         -1.011659e-02
##                 polio            diphtheria                   bmi
##          2.848517e-02          3.930029e-02          6.380365e-02
##     total_expenditure                   gdp percentage_expenditure
##           3.392370e-02           3.916549e-05           2.369521e-04
##              schooling
##           1.518588e+00
```

```
lm = lm(life_expectancy ~ alcohol + hiv_aids +thin_5to9_years + thin_10to19_years + hepatitis_b + polio
summary(lm)
```

```
##
## Call:
## lm(formula = life_expectancy ~ alcohol + hiv_aids + thin_5to9_years +
##     thin_10to19_years + hepatitis_b + polio + diphtheria + bmi +
##     total_expenditure + percentage_expenditure + gdp + schooling,
##     data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.7830  -2.8180   0.0764   2.8194  19.2803
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             4.530e+01  6.991e-01  64.800  < 2e-16 ***
## alcohol                -1.568e-01  3.654e-02  -4.290 1.89e-05 ***
## hiv_aids               -6.322e-01  2.082e-02 -30.361  < 2e-16 ***
## thin_5to9_years        -4.475e-02  6.881e-02  -0.650   0.5155
## thin_10to19_years      -3.503e-02  6.973e-02  -0.502   0.6155
## hepatitis_b            -1.012e-02  4.626e-03  -2.187   0.0289 *
## polio                   2.849e-02  6.559e-03   4.343 1.49e-05 ***
## diphtheria              3.930e-02  6.983e-03   5.628 2.15e-08 ***
## bmi                     6.380e-02  7.633e-03   8.359  < 2e-16 ***
## total_expenditure       3.392e-02  5.097e-02   0.666   0.5057
## percentage_expenditure  2.370e-04  1.525e-04   1.553   0.1205
## gdp                     3.917e-05  2.537e-05   1.544   0.1229
## schooling               1.519e+00  5.480e-02  27.709  < 2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 4.485 on 1566 degrees of freedom
## Multiple R-squared:  0.7895, Adjusted R-squared:  0.7879
## F-statistic: 489.5 on 12 and 1566 DF,  p-value: < 2.2e-16
```

```
test = test %>%
  mutate(predictions = predict(lm, test))

bscv_MSE_test = test %>%
  summarize(bscv_MSE_test = mean((life_expectancy-predictions)^2))
bscv_MSE_test
```

```
## # A tibble: 1 x 1
##   bscv_MSE_test
##           <dbl>
## 1          19.7
```

```r
regfit_best_train1 = regsubsets(life_expectancy ~ ., data=trained, nvmax = 15)
```

```
## Warning in leaps.setup(x, y, wt = wt, nbest = nbest, nvmax = nvmax, force.in =
## force.in, : 1 linear dependencies found

## Reordering variables and trying again:
```

```r
summary(regfit_best_train1)
```

```
## Subset selection object
## Call: regsubsets.formula(life_expectancy ~ ., data = trained, nvmax = 15)
## 15 Variables  (and intercept)
##                        Forced in Forced out
## population                 FALSE      FALSE
## alcohol                    FALSE      FALSE
## thin_5to9_years            FALSE      FALSE
## thin_10to19_years          FALSE      FALSE
## hepatitis_b                FALSE      FALSE
## measles                    FALSE      FALSE
## polio                      FALSE      FALSE
## diphtheria                 FALSE      FALSE
## bmi                        FALSE      FALSE
## under_five_deaths          FALSE      FALSE
## total_expenditure          FALSE      FALSE
## gdp                        FALSE      FALSE
## percentage_expenditure     FALSE      FALSE
## schooling                  FALSE      FALSE
## hiv_aids                   FALSE      FALSE
## 1 subsets of each size up to 14
## Selection Algorithm: exhaustive
##           population alcohol hiv_aids thin_5to9_years thin_10to19_years
## 1  ( 1 )  " "        " "     " "      "*"             " "
## 2  ( 1 )  " "        "*"     " "      "*"             " "
## 3  ( 1 )  " "        "*"     " "      " "             "*"
## 4  ( 1 )  " "        "*"     " "      " "             "*"
## 5  ( 1 )  " "        "*"     " "      " "             "*"
## 6  ( 1 )  " "        "*"     " "      " "             "*"
## 7  ( 1 )  " "        "*"     " "      " "             "*"
## 8  ( 1 )  " "        "*"     " "      " "             "*"
## 9  ( 1 )  " "        "*"     " "      " "             "*"
## 10  ( 1 ) " "        "*"     " "      " "             "*"
## 11  ( 1 ) "*"        "*"     " "      " "             "*"
## 12  ( 1 ) "*"        "*"     " "      "*"             "*"
## 13  ( 1 ) "*"        "*"     " "      "*"             "*"
## 14  ( 1 ) "*"        "*"     " "      "*"             "*"
##           hepatitis_b measles polio diphtheria bmi under_five_deaths
## 1  ( 1 )  " "         " "     " "   " "        " " " "
## 2  ( 1 )  " "         " "     " "   " "        " " " "
## 3  ( 1 )  " "         " "     " "   " "        " " " "
## 4  ( 1 )  " "         " "     " "   " "        " " "*"
## 5  ( 1 )  " "         " "     " "   " "        " " "*"
## 6  ( 1 )  " "         " "     " "   "*"        " " "*"
## 7  ( 1 )  " "         " "     " "   "*"        " " "*"
## 8  ( 1 )  " "         " "     " "   "*"        "*" "*"
## 9  ( 1 )  "*"         " "     " "   "*"        "*" "*"
## 10  ( 1 ) "*"         " "     " "   "*"        "*" "*"
## 11  ( 1 ) "*"         " "     " "   "*"        "*" "*"
## 12  ( 1 ) "*"         " "     " "   "*"        "*" "*"
## 13  ( 1 ) "*"         "*"     " "   "*"        "*" "*"
## 14  ( 1 ) "*"         "*"     "*"   "*"        "*" "*"
##           total_expenditure gdp percentage_expenditure schooling
## 1  ( 1 )  " "               " " " "                    " "
## 2  ( 1 )  " "               " " " "                    " "
## 3  ( 1 )  " "               "*" " "                    " "
## 4  ( 1 )  " "               "*" " "                    " "
## 5  ( 1 )  " "               "*" " "                    "*"
## 6  ( 1 )  " "               "*" " "                    "*"
## 7  ( 1 )  "*"               " " "*"                    "*"
## 8  ( 1 )  "*"               " " "*"                    "*"
## 9  ( 1 )  "*"               " " "*"                    "*"
## 10  ( 1 ) "*"               "*" "*"                    "*"
## 11  ( 1 ) "*"               "*" "*"                    "*"
## 12  ( 1 ) "*"               "*" "*"                    "*"
## 13  ( 1 ) "*"               "*" "*"                    "*"
## 14  ( 1 ) "*"               "*" "*"                    "*"
```

Note j will be 14 here

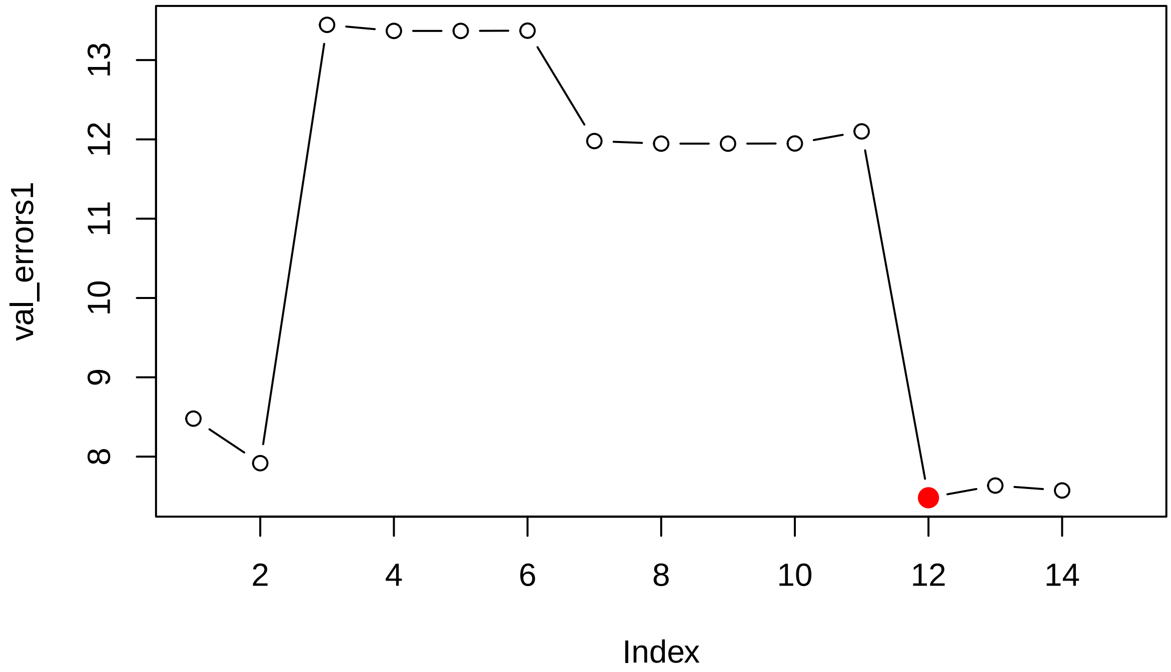
```r
test_mat1 = model.matrix (life_expectancy~., data = tested)
```

```r
val_errors1 = rep(NA,15)

# Iterate over each size j
for(j in 1:14){

    # Extract the vector of predictors in the best fit model on j predictors
    coefi1 = coef(regfit_best_train1, id = j)

    # Make predictions using matrix multiplication of the test matirx and the coefficients vector
    pred1 = test_mat1[,names(coefi1)]%*%coefi1

    # Calculate the MSE
    val_errors1[j] = mean((tested$life_expectancy-pred1)^2)
}
```

```r
# Find the model with the smallest error
min1 = which.min(val_errors1)
min1
```

```
## [1] 12
```

```r
# Plot the errors for each model size
plot(val_errors1, type = 'b')
points(min1, val_errors1[min1][1], col = "red", cex = 2, pch = 20)
```



```
regfit_best1 = regsubsets(life_expectancy~., data = trained, nvmax = 15)
```

```
## Warning in leaps.setup(x, y, wt = wt, nbest = nbest, nvmax = nvmax, force.in =
## force.in, : 1 linear dependencies found

## Reordering variables and trying again:
```

```
coef(regfit_best_train1, 12)
```

```
##           (Intercept)            population               alcohol
##          8.151080e+01         -2.028594e-09         -2.860808e-01
##      thin_10to19_years           hepatitis_b               measles
##         -3.073751e+00          4.662812e-03          1.186363e-05
##                   bmi      under_five_deaths     total_expenditure
##         -1.446335e-02          3.478374e-01         -1.379137e-01
##                   gdp percentage_expenditure             schooling
##          1.661539e-05          8.273472e-05          3.260455e-01
##              hiv_aids
##          0.000000e+00
```

```
lm1 = lm(life_expectancy ~ population + alcohol + hiv_aids  + thin_10to19_years + hepatitis_b + measles
summary(lm1)
```

```
##
## Call:
## lm(formula = life_expectancy ~ population + alcohol + hiv_aids +
##     thin_10to19_years + hepatitis_b + measles + bmi + under_five_deaths +
##     total_expenditure + percentage_expenditure + gdp + schooling,
##     data = trained)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6462 -1.9966 -0.5283  1.1311  9.1806
##
## Coefficients: (1 not defined because of singularities)
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             8.151e+01  2.068e+00  39.425  < 2e-16 ***
## population             -2.029e-09  1.163e-08  -0.174  0.86168
## alcohol                -2.861e-01  5.626e-02  -5.085 6.13e-07 ***
## hiv_aids                       NA         NA      NA       NA
## thin_10to19_years      -3.074e+00  2.494e-01 -12.324  < 2e-16 ***
## hepatitis_b             4.663e-03  4.929e-03   0.946  0.34487
## measles                 1.186e-05  5.594e-05   0.212  0.83219
## bmi                    -1.446e-02  9.351e-03  -1.547  0.12289
## under_five_deaths       3.478e-01  1.524e-01   2.282  0.02312 *
## total_expenditure      -1.379e-01  5.919e-02  -2.330  0.02039 *
## percentage_expenditure  8.273e-05  9.759e-05   0.848  0.39715
## gdp                     1.662e-05  1.678e-05   0.990  0.32276
## schooling               3.260e-01  1.053e-01   3.097  0.00212 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.867 on 335 degrees of freedom
## Multiple R-squared:  0.5147, Adjusted R-squared:  0.4987
## F-statistic: 32.29 on 11 and 335 DF,  p-value: < 2.2e-16
```

```
tested = tested %>%
  mutate(predictions = predict(lm1, tested))
```

```
## Warning in predict.lm(lm1, tested): prediction from a rank-deficient fit may be
## misleading
```

```
bscv_MSE_test1 = tested %>%
  summarize(bscv_MSE_test1 = mean((life_expectancy-predictions)^2))
bscv_MSE_test1
```

```
## # A tibble: 1 x 1
##   bscv_MSE_test1
##            <dbl>
## 1           7.48
```

```
regfit_best_train2 = regsubsets(life_expectancy ~ ., data=training, nvmax = 15)
summary(regfit_best_train2)
```

```
## Subset selection object
## Call: regsubsets.formula(life_expectancy ~ ., data = training, nvmax = 15)
## 15 Variables  (and intercept)
##                        Forced in Forced out
## population                 FALSE      FALSE
## alcohol                    FALSE      FALSE
## hiv_aids                   FALSE      FALSE
## thin_5to9_years            FALSE      FALSE
## thin_10to19_years          FALSE      FALSE
## hepatitis_b                FALSE      FALSE
## measles                    FALSE      FALSE
## polio                      FALSE      FALSE
## diphtheria                 FALSE      FALSE
## bmi                        FALSE      FALSE
## under_five_deaths          FALSE      FALSE
## total_expenditure          FALSE      FALSE
## gdp                        FALSE      FALSE
## percentage_expenditure     FALSE      FALSE
## schooling                  FALSE      FALSE
## 1 subsets of each size up to 15
## Selection Algorithm: exhaustive
##            population alcohol hiv_aids thin_5to9_years thin_10to19_years
## 1  ( 1 )  " "        " "     " "      " "             " "
## 2  ( 1 )  " "        " "     "*"      " "             " "
## 3  ( 1 )  " "        " "     "*"      " "             " "
## 4  ( 1 )  " "        " "     "*"      " "             " "
## 5  ( 1 )  " "        " "     "*"      " "             " "
## 6  ( 1 )  " "        "*"     "*"      " "             " "
## 7  ( 1 )  " "        "*"     "*"      " "             " "
## 8  ( 1 )  " "        "*"     "*"      " "             " "
## 9  ( 1 )  " "        "*"     "*"      " "             " "
## 10  ( 1 ) " "        "*"     "*"      " "             " "
## 11  ( 1 ) " "        "*"     "*"      "*"             "*"
## 12  ( 1 ) " "        "*"     "*"      "*"             "*"
## 13  ( 1 ) " "        "*"     "*"      "*"             "*"
## 14  ( 1 ) "*"        "*"     "*"      "*"             "*"
## 15  ( 1 ) "*"        "*"     "*"      "*"             "*"
##            hepatitis_b measles polio diphtheria bmi under_five_deaths
## 1  ( 1 )  " "         " "     " "   " "        " " " "
## 2  ( 1 )  " "         " "     " "   " "        " " " "
## 3  ( 1 )  " "         " "     " "   " "        "*" " "
## 4  ( 1 )  " "         " "     " "   "*"        "*" " "
## 5  ( 1 )  " "         " "     " "   "*"        "*" " "
## 6  ( 1 )  " "         " "     " "   "*"        "*" " "
## 7  ( 1 )  " "         " "     "*"   "*"        "*" " "
## 8  ( 1 )  " "         " "     "*"   "*"        "*" " "
## 9  ( 1 )  "*"         " "     "*"   "*"        "*" " "
## 10  ( 1 ) "*"         "*"     "*"   "*"        "*" " "
## 11  ( 1 ) "*"         " "     "*"   "*"        "*" " "
## 12  ( 1 ) "*"         "*"     "*"   "*"        "*" " "
## 13  ( 1 ) "*"         "*"     "*"   "*"        "*" " "
## 14  ( 1 ) "*"         "*"     "*"   "*"        "*" " "
## 15  ( 1 ) "*"         "*"     "*"   "*"        "*" "*"
##            total_expenditure gdp percentage_expenditure schooling
## 1  ( 1 )  " "               " " " "                    "*"
## 2  ( 1 )  " "               " " " "                    "*"
## 3  ( 1 )  " "               " " " "                    "*"
## 4  ( 1 )  " "               " " " "                    "*"
## 5  ( 1 )  " "               " " "*"                    "*"
## 6  ( 1 )  " "               " " "*"                    "*"
## 7  ( 1 )  " "               " " "*"                    "*"
## 8  ( 1 )  "*"               " " "*"                    "*"
## 9  ( 1 )  "*"               " " "*"                    "*"
## 10  ( 1 ) "*"               " " "*"                    "*"
## 11  ( 1 ) "*"               " " "*"                    "*"
## 12  ( 1 ) "*"               " " "*"                    "*"
## 13  ( 1 ) "*"               "*" "*"                    "*"
## 14  ( 1 ) "*"               "*" "*"                    "*"
## 15  ( 1 ) "*"               "*" "*"                    "*"
```

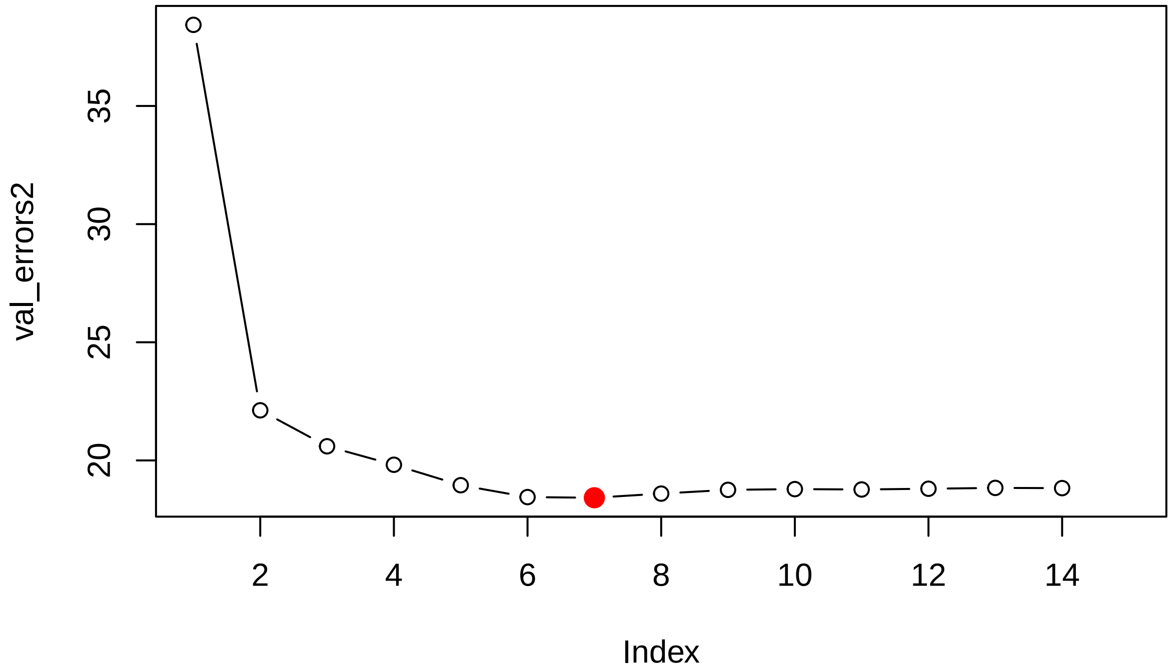
```
test_mat2 = model.matrix (life_expectancy~., data = testing)
```

```
val_errors2 = rep(NA,15)

# Iterate over each size k
for(k in 1:14){

    # Extract the vector of predictors in the best fit model on k predictors
    coefi2 = coef(regfit_best_train2, id = k)

    # Make predictions using matrix multiplication of the test matirx and the coefficients vector
    pred2 = test_mat2[,names(coefi2)]%*%coefi2

    # Calculate the MSE
    val_errors2[k] = mean((testing$life_expectancy-pred2)^2)
}
```

```
# Find the model with the smallest error
min2 = which.min(val_errors2)
min2
```

```
## [1] 7
```

```
# Plot the errors for each model size
plot(val_errors2, type = 'b')
points(min2, val_errors2[min2][1], col = "red", cex = 2, pch = 20)
```



```
regfit_best2 = regsubsets(life_expectancy~., data = training, nvmax = 15)
coef(regfit_best_train2, 7)
```

```
##           (Intercept)               alcohol              hiv_aids
##          45.380753118          -0.270902825          -0.641698147
##                 polio            diphtheria                   bmi
##           0.025363314           0.039163343           0.089645622
## percentage_expenditure             schooling
##           0.002403875           1.329845242
```

```
lm2 = lm(life_expectancy ~ alcohol + hiv_aids  + polio + bmi + diphtheria + percentage_expenditure + scl
summary(lm1)
```

```
##
## Call:
## lm(formula = life_expectancy ~ population + alcohol + hiv_aids +
##     thin_10to19_years + hepatitis_b + measles + bmi + under_five_deaths +
##     total_expenditure + percentage_expenditure + gdp + schooling,
##     data = trained)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6462 -1.9966 -0.5283  1.1311  9.1806
##
```

```
## Coefficients: (1 not defined because of singularities)
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)               8.151e+01  2.068e+00  39.425  < 2e-16 ***
## population               -2.029e-09  1.163e-08  -0.174  0.86168
## alcohol                  -2.861e-01  5.626e-02  -5.085 6.13e-07 ***
## hiv_aids                         NA         NA      NA       NA
## thin_10to19_years        -3.074e+00  2.494e-01 -12.324  < 2e-16 ***
## hepatitis_b               4.663e-03  4.929e-03   0.946  0.34487
## measles                   1.186e-05  5.594e-05   0.212  0.83219
## bmi                      -1.446e-02  9.351e-03  -1.547  0.12289
## under_five_deaths         3.478e-01  1.524e-01   2.282  0.02312 *
## total_expenditure        -1.379e-01  5.919e-02  -2.330  0.02039 *
## percentage_expenditure  8.273e-05  9.759e-05   0.848  0.39715
## gdp                       1.662e-05  1.678e-05   0.990  0.32276
## schooling                 3.260e-01  1.053e-01   3.097  0.00212 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.867 on 335 degrees of freedom
## Multiple R-squared:  0.5147, Adjusted R-squared:  0.4987
## F-statistic: 32.29 on 11 and 335 DF,  p-value: < 2.2e-16
```

```
testing = testing %>%
  mutate(predictions = predict(lm2, testing))

bscv_MSE_test2 = testing %>%
  summarize(bscv_MSE_test2 = mean((life_expectancy-predictions)^2))
bscv_MSE_test2
```

```
## # A tibble: 1 x 1
##   bscv_MSE_test2
##            <dbl>
## 1           18.4
```

##Ridge Regression

```
library(glmnet)
```

```
## Warning: package ’glmnet’ was built under R version 4.1.3
```

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

```
##
## Attaching package: ’Matrix’
```

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

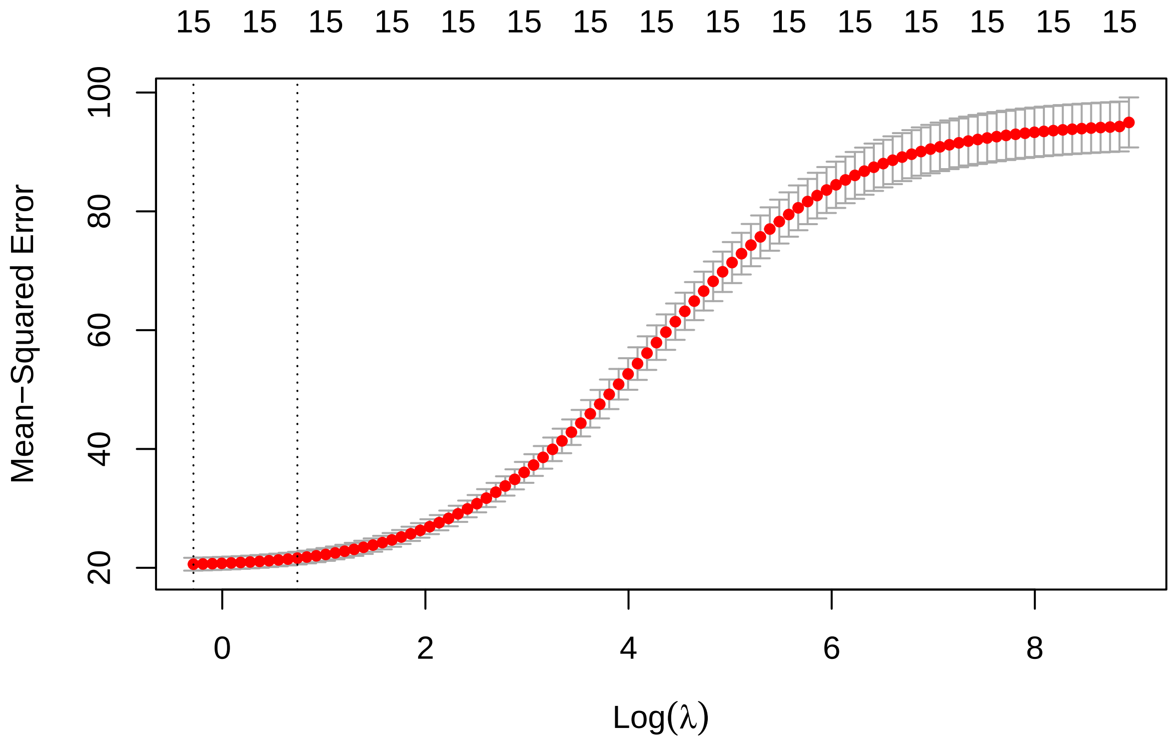
```
## Loaded glmnet 4.1-3
```

```
library(pls)
```

```
## Warning: package 'pls' was built under R version 4.1.3

##
## Attaching package: 'pls'

## The following object is masked from 'package:stats':
##
##     loadings
```

```
#remove column 17: prediction
test = test[,1:16]
tested = tested[,1:16]
testing = testing[,1:16]
```

```
set.seed(seed)
#remove life_expectancy column
x_train = model.matrix(life_expectancy~., train)[,-1]
x_test = model.matrix(life_expectancy~., test)[,-1]



y_train = train %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

y_test = test %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

grid = 10^seq(10, -2, length = 100)

ridge_mod = cv.glmnet(x_train, y_train, alpha = 0, lambda=grid, thresh = 1e-12)


cv.out00 = cv.glmnet(x_train, y_train, alpha = 0)
plot(cv.out00)
```

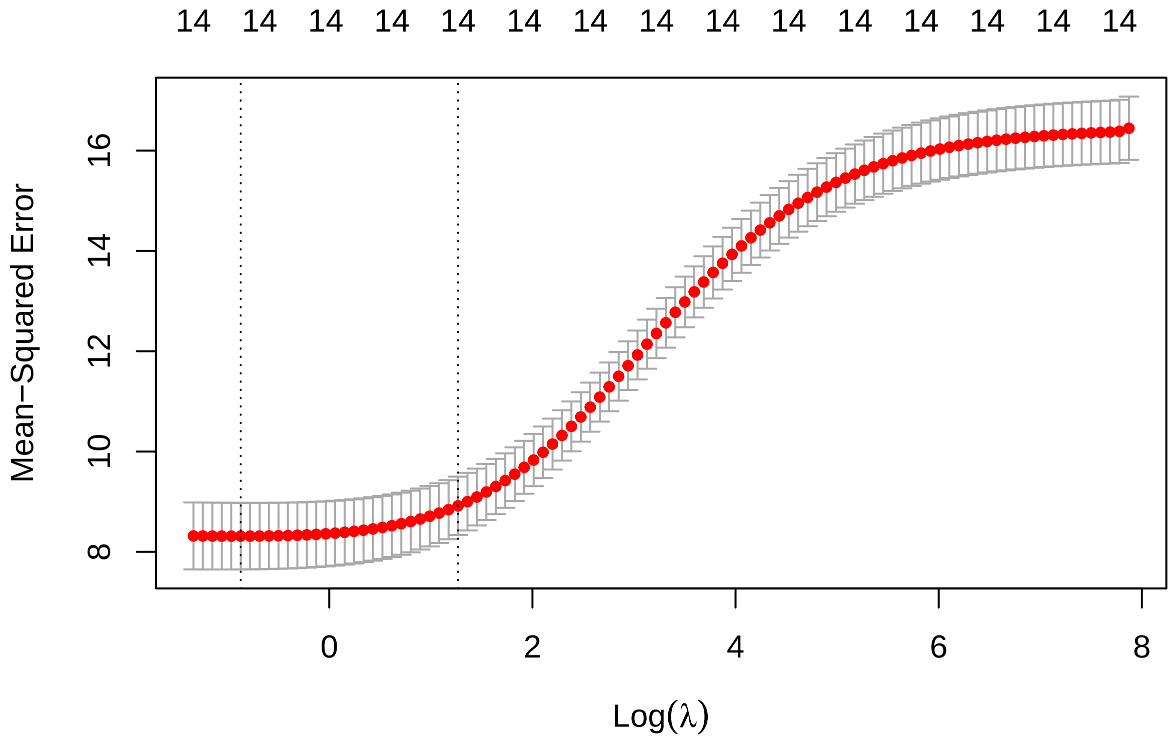


```
bestlam = cv.out00$lambda.min


ridge_pred = predict(ridge_mod, s = bestlam, newx = x_test)

rr_MSE_test = mean((ridge_pred - y_test)^2)
rr_MSE_test
```

```
## [1] 19.71535
```

```
#remove life_expectancy column
x_train1 = model.matrix(life_expectancy~., trained)[,-1]
x_test1 = model.matrix(life_expectancy~., tested)[,-1]




y_train1 = trained %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

y_test1 = tested %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()
```

```
grid = 10^seq(10, -2, length = 100)

ridge_mod1 = cv.glmnet(x_train1, y_train1, alpha = 0, lambda=grid, thresh = 1e-12)


cv.out11 = cv.glmnet(x_train1, y_train1, alpha = 0)
plot(cv.out11)
```

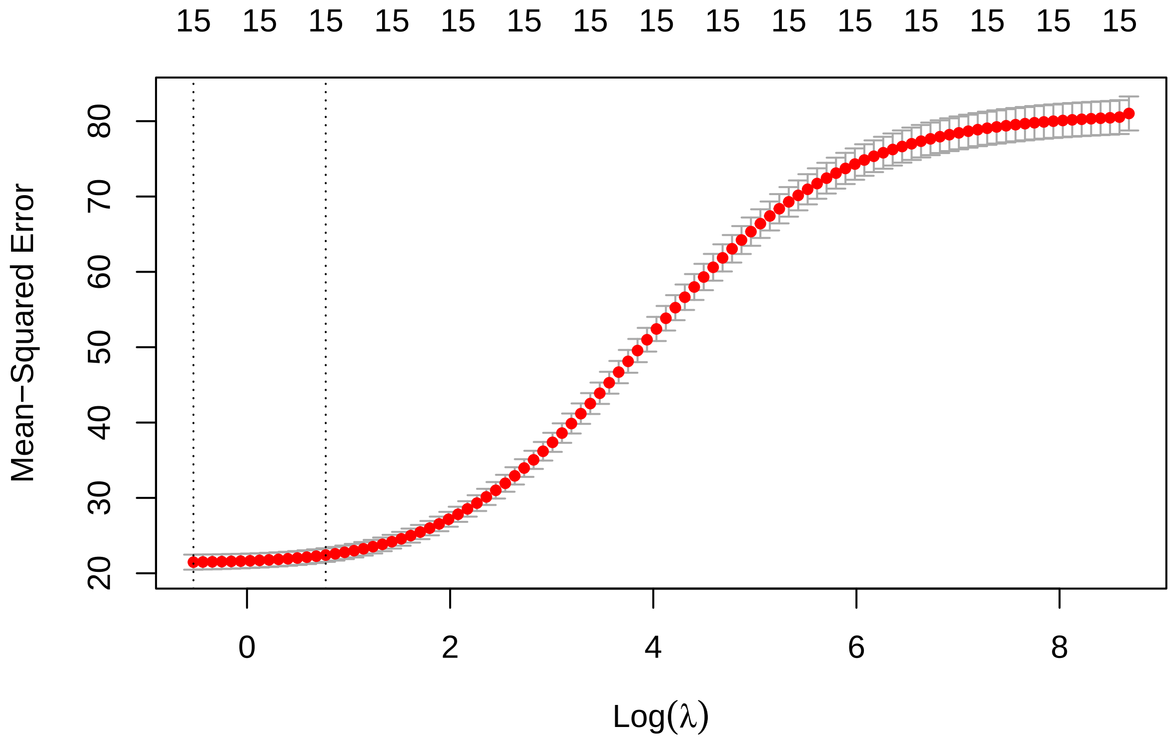


```
bestlam1 = cv.out11$lambda.min


ridge_pred1 = predict(ridge_mod1, s = bestlam1, newx = x_test1)

rr_MSE_test1 = mean((ridge_pred1 - y_test1)^2)
rr_MSE_test1
```

```
## [1] 7.549211
```

```
#remove life_expectancy column
x_train2 = model.matrix(life_expectancy~., training)[,-1]
x_test2 = model.matrix(life_expectancy~., testing)[,-1]
```

```
y_train2 = training %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

y_test2 = testing %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

grid = 10^seq(10, -2, length = 100)

ridge_mod2 = cv.glmnet(x_train2, y_train2, alpha = 0, lambda=grid, thresh = 1e-12)


cv.out22 = cv.glmnet(x_train2, y_train2, alpha = 0)
plot(cv.out22)
```

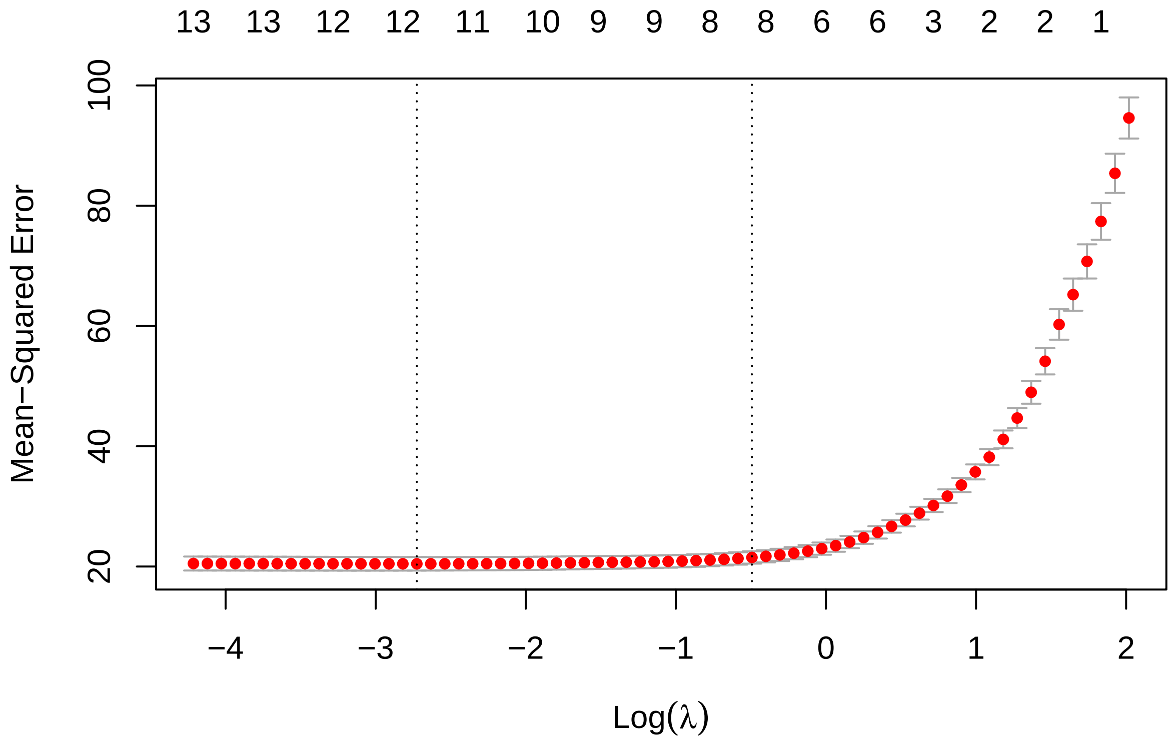


```
bestlam2 = cv.out22$lambda.min


ridge_pred2 = predict(ridge_mod2, s = bestlam2, newx = x_test2)

rr_MSE_test2 = mean((ridge_pred2 - y_test2)^2)
rr_MSE_test2
```

```
## [1] 18.81358
```

##LASSO

```r
set.seed(seed)
#remove life_expectancy column
x_train = model.matrix(life_expectancy~., train)[,-1]
x_test = model.matrix(life_expectancy~., test)[,-1]
x_test = x_test[,1:15]

y_train = train %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

y_test = test %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

grid = 10^seq(10, -2, length = 100)

lasso_mod = glmnet(x_train,y_train, alpha = 1, lambda = grid)


cv.out = cv.glmnet(x_train, y_train, alpha = 1)
plot(cv.out)
```

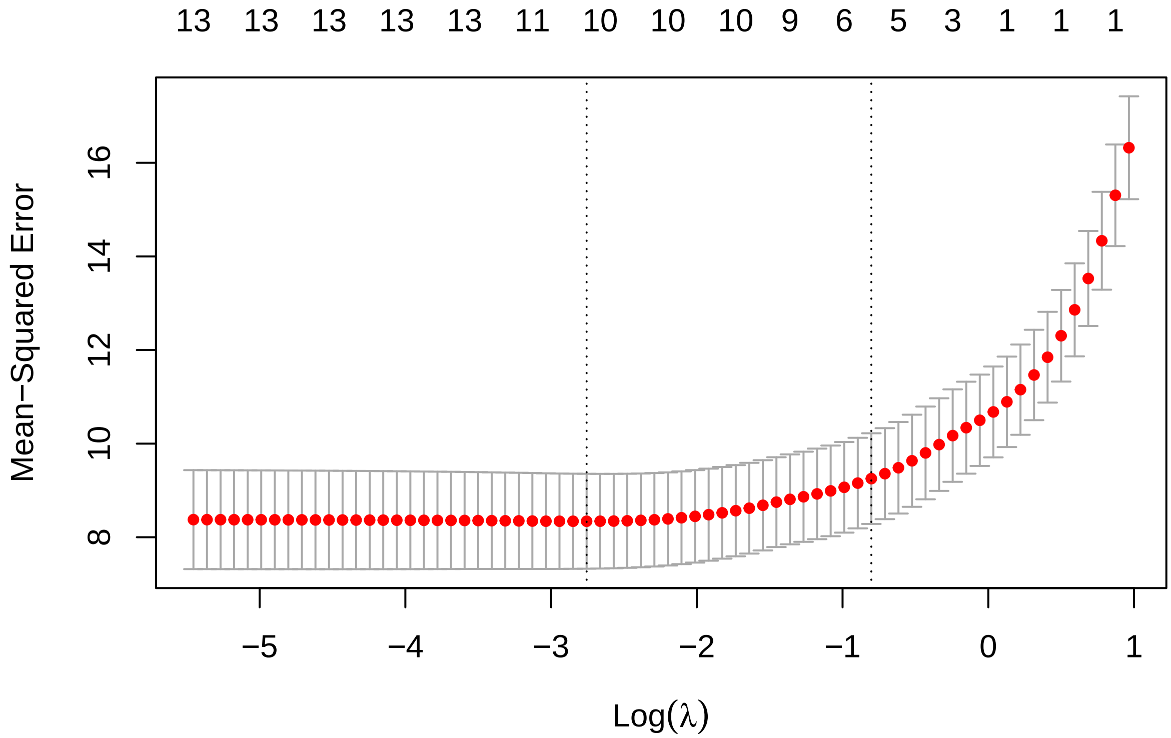


```
bestlam = cv.out$lambda.min

lasso_pred = predict(lasso_mod, s = bestlam, newx = x_test) # Use best lambda to predict test data
lasso_MSE_test = mean((lasso_pred - y_test)^2)
lasso_MSE_test
```

```
## [1] 19.67224
```

```
#rsq calc
rss <-sum((lasso_pred - y_test)^2)
tss <-sum((lasso_pred - mean(y_test))^2)
rsq_lasso <- 1 -rss/tss


#remove life_expectancy column
x_train1 = model.matrix(life_expectancy~., trained)[,-1]
x_test1 = model.matrix(life_expectancy~., tested)[,-1]



y_train1 = trained %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()
```

```
y_test1 = tested %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

grid = 10^seq(10, -2, length = 100)

lasso_mod1 = cv.glmnet(x_train1, y_train1, alpha = 1, lambda=grid, thresh = 1e-12)


cv.out1 = cv.glmnet(x_train1, y_train1, alpha = 1)
plot(cv.out1)
```

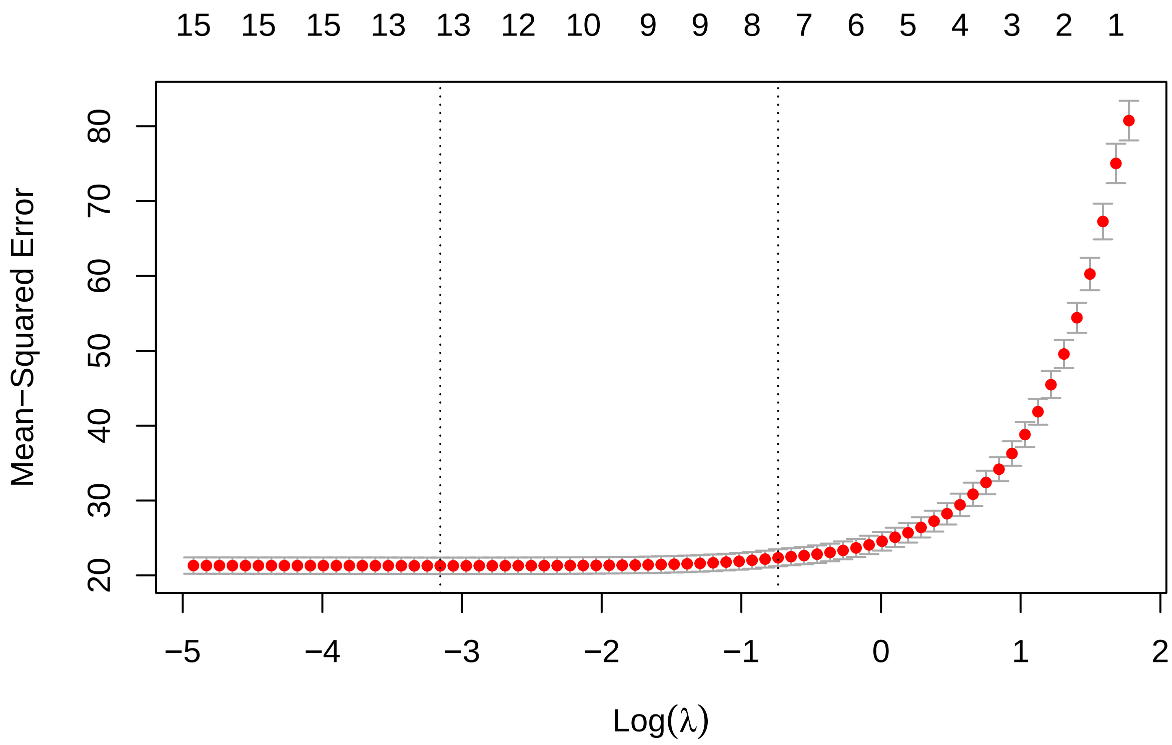


```
bestlam1 = cv.out1$lambda.min


lasso_pred1 = predict(lasso_mod1, s = bestlam1, newx = x_test1)
lasso_MSE_test1 = mean((lasso_pred1 - y_test1)^2)
lasso_MSE_test1
```

```
## [1] 7.44295
```

```
#remove life_expectancy column
x_train2 = model.matrix(life_expectancy~., training)[,-1]
```

```
x_test2 = model.matrix(life_expectancy~., testing)[,-1]



y_train2 = training %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

y_test2 = testing %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

grid = 10^seq(10, -2, length = 100)

lasso_mod2 = cv.glmnet(x_train2, y_train2, alpha = 1, lambda=grid, thresh = 1e-12)


cv.out2 = cv.glmnet(x_train2, y_train2, alpha = 1)
plot(cv.out2)
```



```
bestlam2 = cv.out2$lambda.min
```

```
lasso_pred2 = predict(lasso_mod2, s = bestlam2, newx = x_test2)
lasso_MSE_test2 = mean((lasso_pred2 - y_test2)^2)
lasso_MSE_test2
```

```
## [1] 18.64986
```

PCR

```
set.seed(seed)

pcr_fit = pcr(life_expectancy~., data = train, scale = TRUE, validation = "CV")
summary(pcr_fit)
```

```
## Data:    X dimension: 1579 15
##  Y dimension: 1579 1
## Fit method: svdpc
## Number of components considered: 15
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV           9.742    6.684    6.645    6.254    5.923    5.416    5.299
## adjCV        9.742    6.680    6.642    6.251    5.920    5.400    5.336
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV       5.159    4.946    4.906     4.897     4.778     4.732     4.516
## adjCV    5.159    4.943    4.905     4.895     4.757     4.735     4.513
##        14 comps  15 comps
## CV        4.516     4.518
## adjCV     4.514     4.515
##
## TRAINING: % variance explained
##                  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X                  31.14    44.45    55.72    64.00    70.62    76.01    81.31
## life_expectancy    53.50    54.55    59.06    63.47    70.19    70.26    72.20
##                  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## X                  85.65    89.46      92.9     95.14     97.29     99.19
## life_expectancy    74.63    75.03      75.2     76.88     77.00     78.95
##                  14 comps  15 comps
## X                   99.63    100.00
## life_expectancy     78.95     78.95
```

```
#have to remove variable hiv_aids as it is causing an infite loop
pcr_fit1 = pcr(life_expectancy~.-hiv_aids, data = trained, scale = TRUE, validation = "CV")
summary(pcr_fit1)
```

```
## Data:    X dimension: 347 14
##  Y dimension: 347 1
## Fit method: svdpc
## Number of components considered: 14
##
```

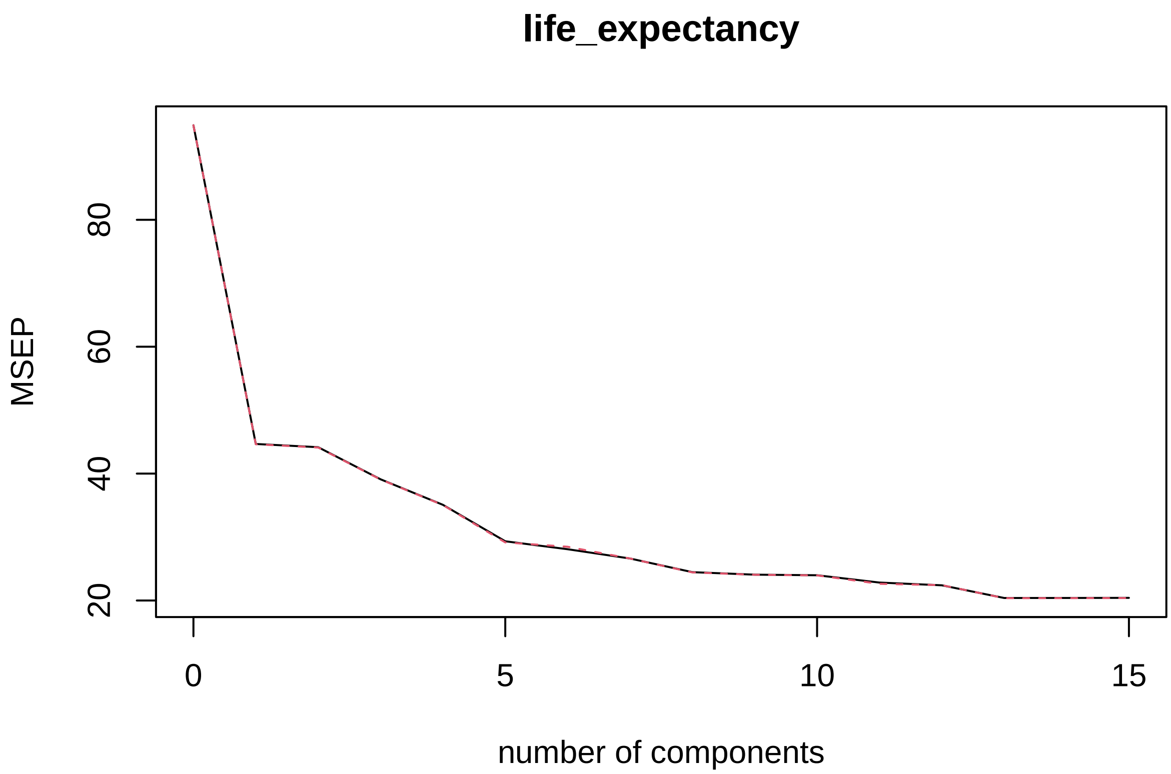
```
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV           4.055    3.164    3.129    3.115    3.009    3.024    3.026
## adjCV        4.055    3.162    3.124    3.116    3.006    3.022    3.022
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV       3.003    3.004     3.01     2.914     2.896     2.888     2.895
## adjCV    2.999    3.001     3.01     2.909     2.891     2.883     2.889
##        14 comps
## CV        2.908
## adjCV     2.901
##
## TRAINING: % variance explained
##                  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X                  22.17    35.06    47.32    57.89    66.65    73.73    80.08
## life_expectancy    39.23    40.98    41.22    45.51    45.55    46.14    47.17
##                  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## X                  85.79    90.34     94.49     97.40     99.33     99.95
## life_expectancy    47.26    47.32     51.50     52.08     52.47     52.47
##                  14 comps
## X                  100.00
## life_expectancy     52.53
```

```r
pcr_fit2 = pcr(life_expectancy~., data = training, scale = TRUE, validation = "CV")
summary(pcr_fit2)
```

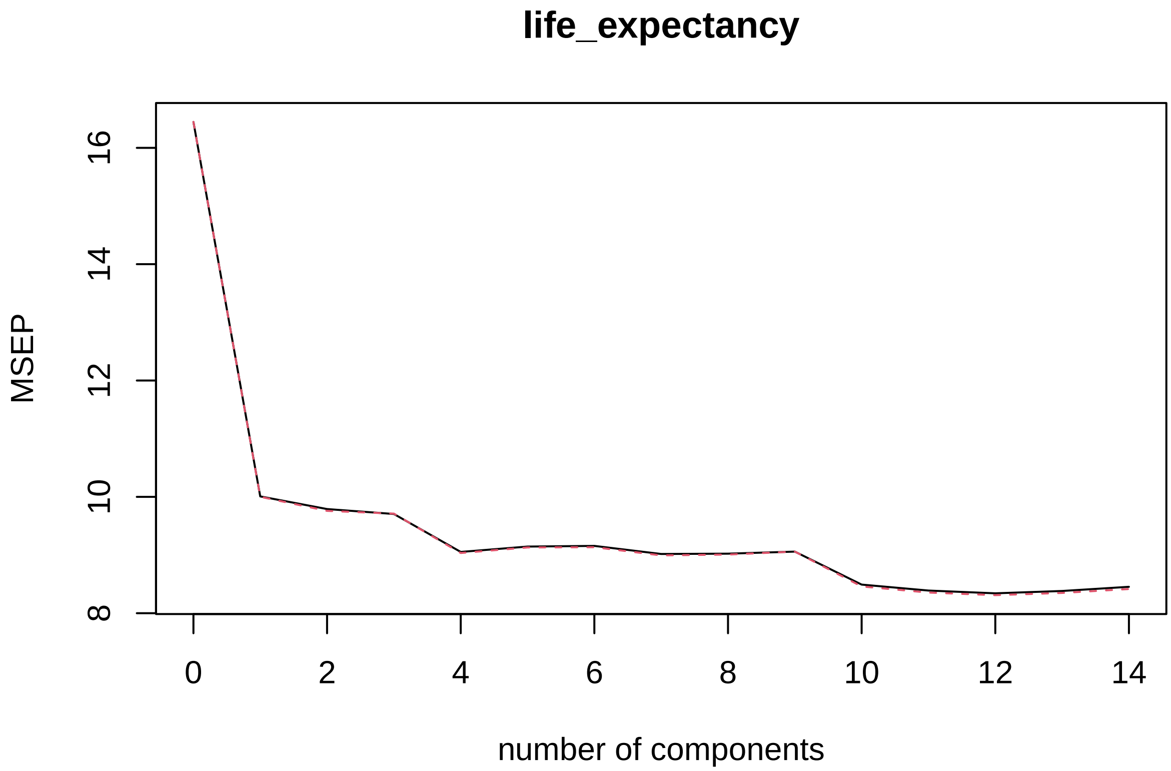
```
## Data:    X dimension: 1232 15
##  Y dimension: 1232 1
## Fit method: svdpc
## Number of components considered: 15
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV           9.007    7.063    6.665    6.665    5.976    5.225    5.209
## adjCV        9.007    7.060    6.661    6.661    5.971    5.219    5.202
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV       5.188    4.801    4.724     4.733     4.652     4.656     4.646
## adjCV    5.185    4.798    4.721     4.730     4.644     4.649     4.642
##        14 comps  15 comps
## CV        4.637     4.635
## adjCV     4.632     4.630
##
## TRAINING: % variance explained
##                  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X                  27.76    40.99    52.95    61.73    68.98    75.06    80.25
## life_expectancy    39.23    46.10    46.12    56.56    67.06    67.39    67.52
##                  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## X                  84.78    88.56     91.73     94.07     96.17     98.14
## life_expectancy    72.16    73.08     73.08     74.19     74.27     74.29
##                  14 comps  15 comps
## X                   99.53    100.00
## life_expectancy     74.45     74.48
```

Finding number of components

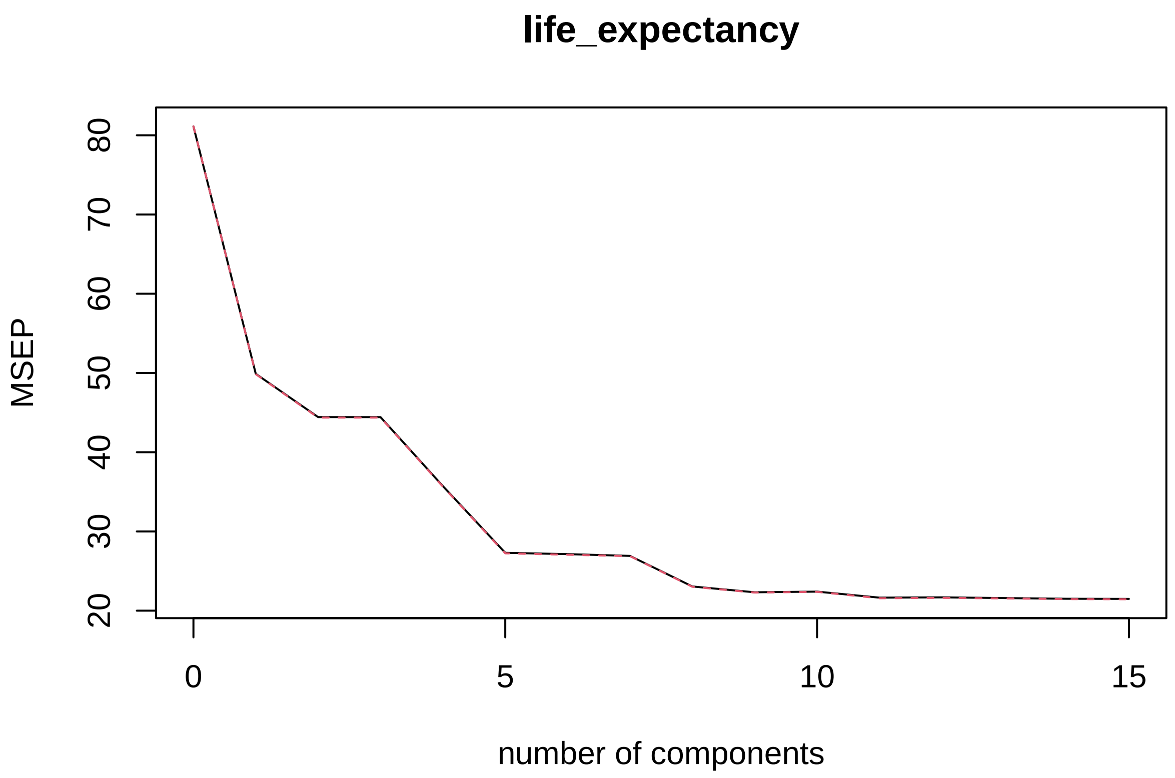
```
validationplot(pcr_fit, val.type = "MSEP")
```



```
validationplot(pcr_fit1, val.type = "MSEP")
```



```
validationplot(pcr_fit2, val.type = "MSEP")
```



```
x_train = model.matrix(life_expectancy~., train)[,-1]
x_test = model.matrix(life_expectancy~., test)[,-1]

y_train = train %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

y_test = test %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

pcr_pred = predict(pcr_fit, x_test, ncomp=13)
pcr_MSE_test = mean((pcr_pred-y_test)^2)
pcr_MSE_test
```

```
## [1] 19.66008
```

```
x_train1 = model.matrix(life_expectancy~.-hiv_aids, trained)[,-1]
x_test1 = model.matrix(life_expectancy~.-hiv_aids, tested)[,-1]

y_train1 = trained %>%
  select(life_expectancy) %>%
  unlist() %>%
```

```r
  as.numeric()

y_test1 = tested %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

pcr_pred1 = predict(pcr_fit1, x_test1, ncomp=12)
pcr_MSE_test1 = mean((pcr_pred1-y_test1)^2)
pcr_MSE_test1
```

```
## [1] 7.59744
```

```r
x_train2 = model.matrix(life_expectancy~., training)[,-1]
x_test2 = model.matrix(life_expectancy~., testing)[,-1]

y_train2 = training %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

y_test2 = testing %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

pcr_pred2 = predict(pcr_fit2, x_test2, ncomp=15)
pcr_MSE_test2 = mean((pcr_pred2-y_test2)^2)
pcr_MSE_test2
```

```
## [1] 18.8309
```

##PLS

```r
set.seed(seed)

pls_fit = plsr(life_expectancy~., data = train, scale = TRUE, validation = "CV")
summary(pls_fit)
```

```
## Data:    X dimension: 1579 15
##  Y dimension: 1579 1
## Fit method: kernelpls
## Number of components considered: 15
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV           9.742    5.914    4.823    4.623    4.541    4.523    4.519
## adjCV        9.742    5.911    4.821    4.620    4.539    4.520    4.516
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV       4.517    4.517    4.517     4.518     4.518     4.518     4.518
## adjCV    4.515    4.514    4.514     4.515     4.515     4.515     4.515
##        14 comps  15 comps
## CV        4.518     4.518
## adjCV     4.515     4.515
##
## TRAINING: % variance explained
##                  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X                  30.53    39.86    48.34    55.19    63.59    70.47    76.56
## life_expectancy    63.64    75.92    78.00    78.71    78.90    78.95    78.95
##                  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## X                  80.38    83.84     85.67     88.43     92.26     96.14
## life_expectancy    78.95    78.95     78.95     78.95     78.95     78.95
##                  14 comps  15 comps
## X                   98.10    100.00
## life_expectancy     78.95     78.95
```

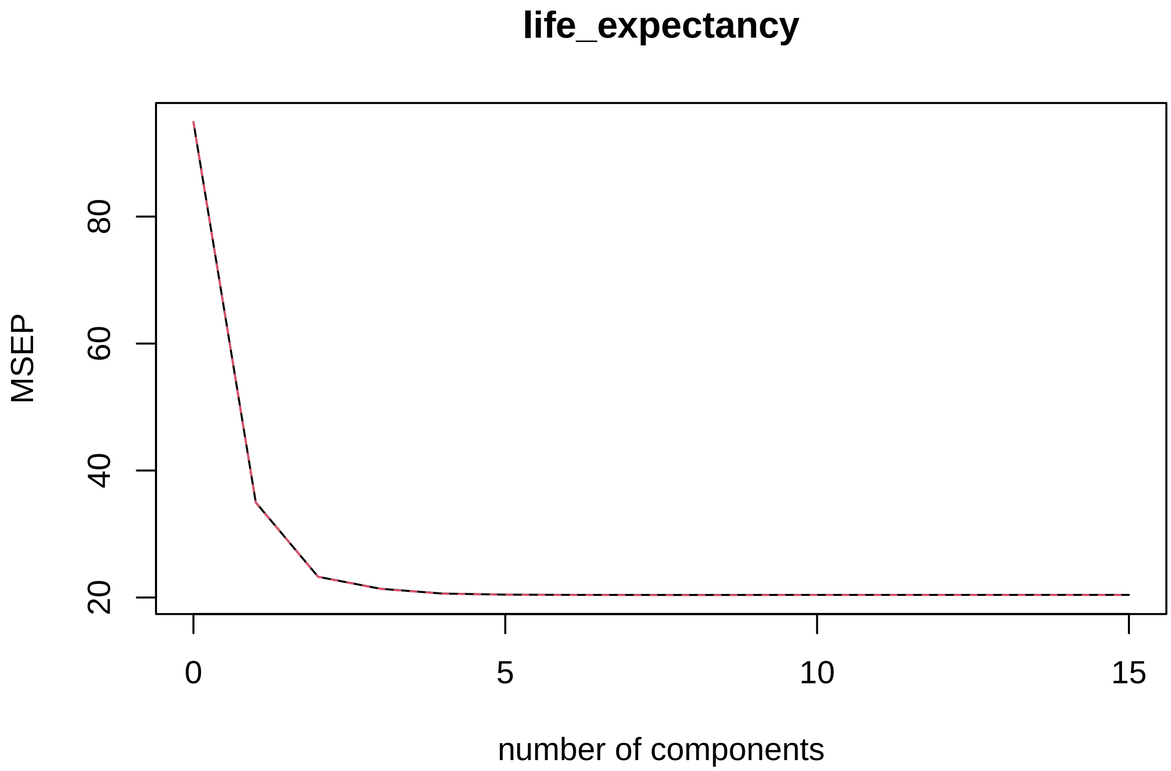
```
#removed hiv_aids variable as it was causing a loop error
pls_fit1 = plsr(life_expectancy~.-hiv_aids, data = trained, scale = TRUE, validation = "CV")
summary(pls_fit1)
```

```
## Data:    X dimension: 347 14
##  Y dimension: 347 1
## Fit method: kernelpls
## Number of components considered: 14
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV           4.055    2.988    2.910    2.901    2.892    2.889    2.893
## adjCV        4.055    2.987    2.906    2.895    2.886    2.884    2.888
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV       2.893    2.897    2.900     2.903     2.906     2.908     2.908
## adjCV    2.887    2.891    2.895     2.897     2.899     2.902     2.901
##        14 comps
## CV        2.908
## adjCV     2.901
##
## TRAINING: % variance explained
##                  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X                  21.61    31.40    38.58    47.84    54.42    58.96    68.21
## life_expectancy    46.69    51.19    52.30    52.42    52.46    52.48    52.48
##                  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## X                  74.38    80.11      83.8     89.21     92.54     95.26
## life_expectancy    52.48    52.49      52.5     52.50     52.51     52.53
##                  14 comps
## X                  100.00
## life_expectancy     52.53
```

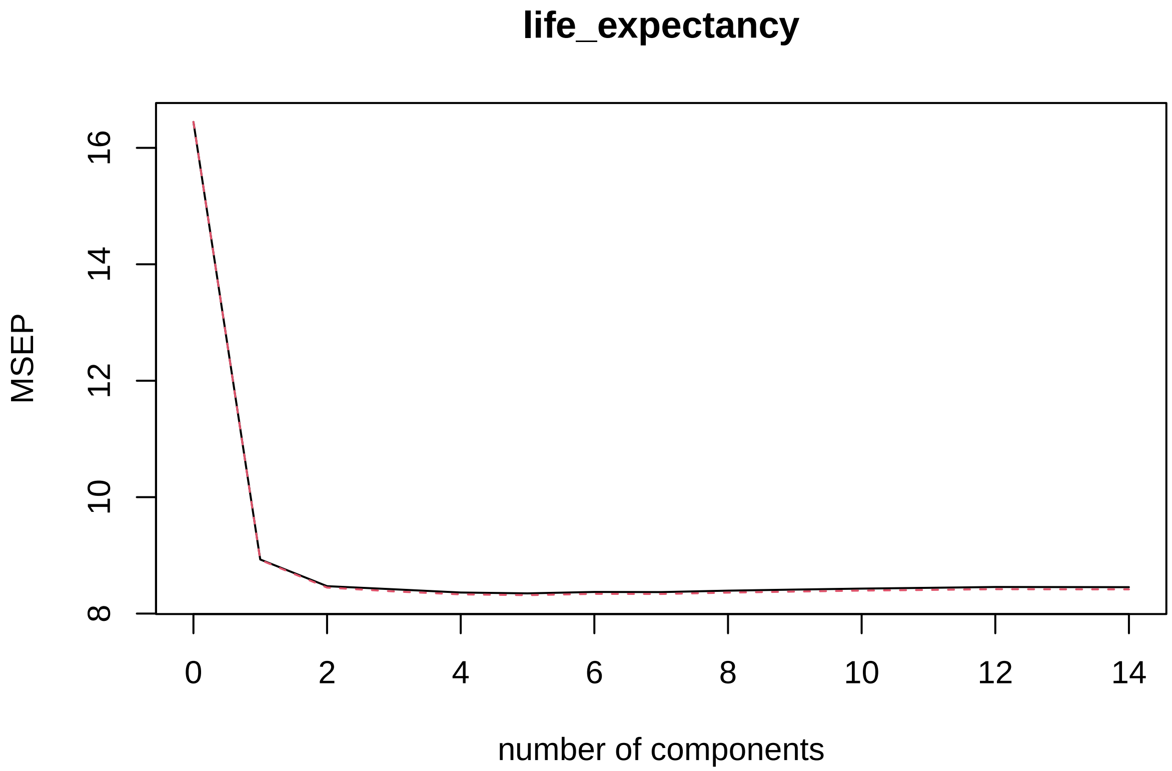
```
pls_fit2 = plsr(life_expectancy~., data = training, scale = TRUE, validation = "CV")
summary(pls_fit2)
```

```
## Data:    X dimension: 1232 15
##  Y dimension: 1232 1
## Fit method: kernelpls
## Number of components considered: 15
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV           9.007    5.912    4.816    4.673    4.638    4.637    4.635
## adjCV        9.007    5.908    4.813    4.670    4.634    4.632    4.631
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV       4.636    4.636    4.635     4.636     4.635     4.635     4.635
## adjCV    4.631    4.631    4.630     4.631     4.631     4.631     4.630
##        14 comps  15 comps
## CV        4.635     4.635
## adjCV     4.630     4.630
##
## TRAINING: % variance explained
##                  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X                  26.33    37.06    47.03    53.35    58.92    63.73    71.77
## life_expectancy    57.69    72.15    73.88    74.35    74.44    74.46    74.47
##                  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## X                  77.56    81.51     84.96     87.65     92.22     94.80
## life_expectancy    74.47    74.48     74.48     74.48     74.48     74.48
##                  14 comps  15 comps
## X                   97.23    100.00
## life_expectancy     74.48     74.48
```

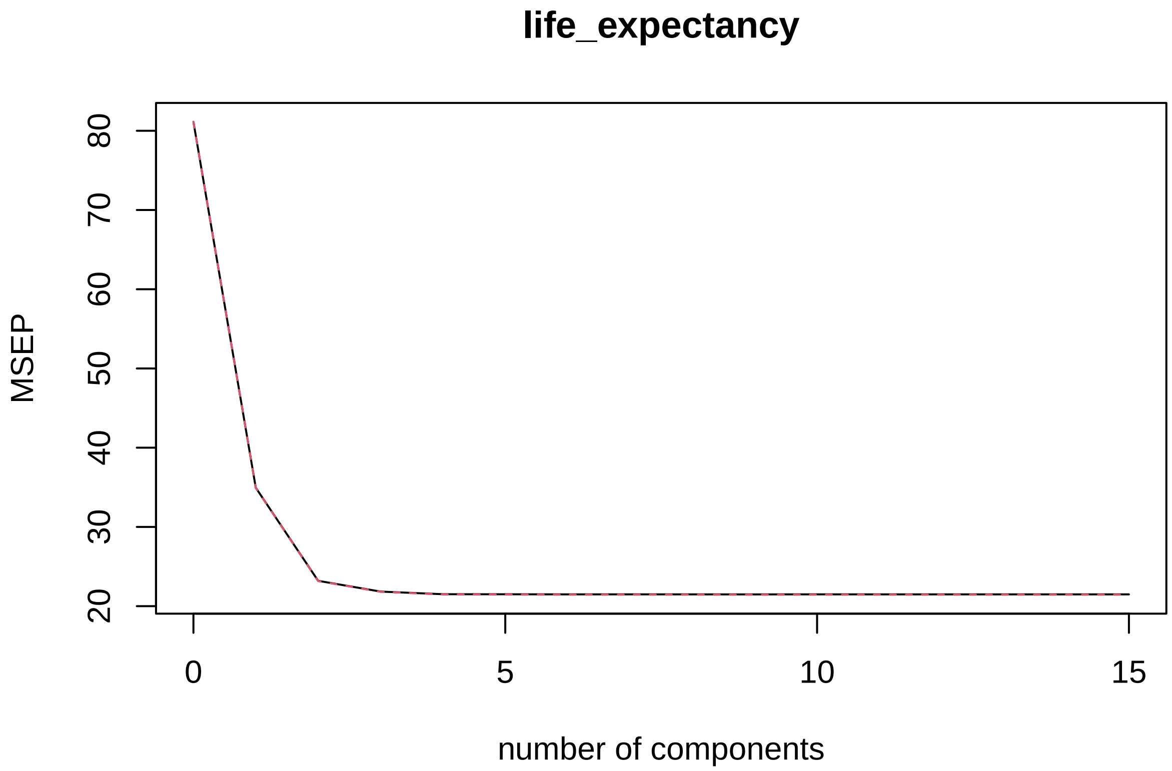
```
validationplot(pls_fit, val.type = "MSEP")
```



```
validationplot(pls_fit1, val.type = "MSEP")
```



```
validationplot(pls_fit2, val.type = "MSEP")
```



```
x_train = model.matrix(life_expectancy~., train)[,-1]
x_test = model.matrix(life_expectancy~., test)[,-1]


y_train = train %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

y_test = test %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

pls_pred = predict(pls_fit, x_test, ncomp = 8)
pls_MSE_test = mean((pls_pred - y_test)^2)
pls_MSE_test
```

```
## [1] 19.66842
```

```
x_train1 = model.matrix(life_expectancy~.-hiv_aids, trained)[,-1]
x_test1 = model.matrix(life_expectancy~.-hiv_aids, tested)[,-1]

y_train1 = trained %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

y_test1 = tested %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

pls_pred1 = predict(pls_fit1, x_test1, ncomp = 5)
pls_MSE_test1 = mean((pls_pred1 - y_test1)^2)
pls_MSE_test1
```

```
## [1] 7.601446
```

```
x_train2 = model.matrix(life_expectancy~., training)[,-1]
x_test2 = model.matrix(life_expectancy~., testing)[,-1]

y_train2 = training %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

y_test2 = testing %>%
  select(life_expectancy) %>%
  unlist() %>%
  as.numeric()

pls_pred2 = predict(pls_fit2, x_test2, ncomp = 9)
pls_MSE_test2 = mean((pls_pred2 - y_test2)^2)
pls_MSE_test2
```

```
## [1] 18.82288
```

Load caret, see if there is any variance inflation

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.1.3
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:pls':
##
##     R2
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

##All Countries >The lowest MSE average from all multiple seeds was found using Lasso Regression method.

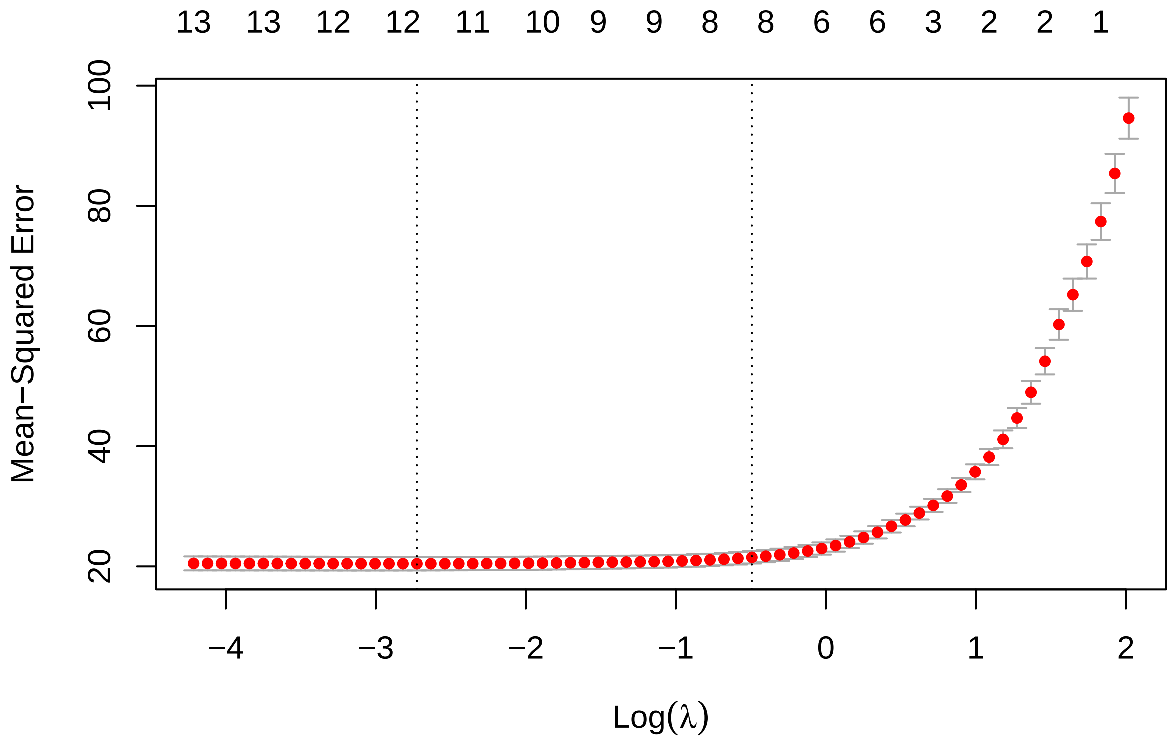
```
coef(cv.out)
```

```
## 16 x 1 sparse Matrix of class "dgCMatrix"
##                                  s1
## (Intercept)             4.697349e+01
## population              .
## alcohol                 .
## hiv_aids               -5.677281e-01
## thin_5to9_years        -5.411015e-03
## thin_10to19_years       .
## hepatitis_b             .
## measles                 .
## polio                   1.685117e-02
## diphtheria              2.317090e-02
## bmi                     5.232275e-02
## under_five_deaths       .
## total_expenditure       .
## gdp                     3.923977e-05
## percentage_expenditure  3.800951e-05
## schooling               1.469963e+00
```

```
rsq_lasso
```

```
## [1] 0.7399873
```

```
plot(cv.out)
```

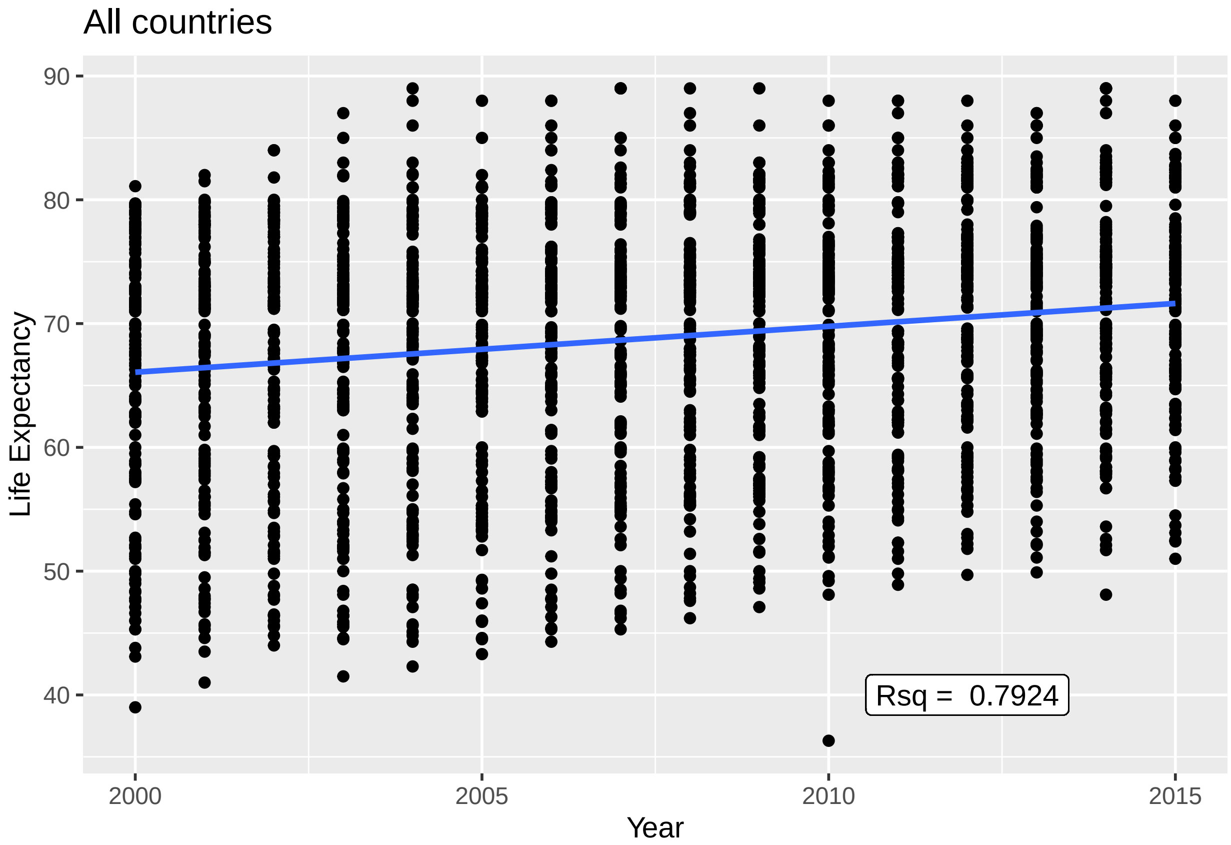


```
lm = lm(life_expectancy ~population+alcohol+hiv_aids+thin_5to9_years+polio+diphtheria+bmi+gdp+percentag
modsum =summary(lm)
r2 = modsum$r.squared
r2 = round(r2, digits=4)
r2
```

```
## [1] 0.7924
```

```
rsq = "Rsq = "
ggplot(le_adj,aes(x=year,y=life_expectancy))+
  geom_point()+
  labs(title= "All countries", y="Life Expectancy", x = "Year")+
  geom_smooth(se=FALSE, method="lm")+
  #geom_label(label=r2, x=2012, y=40,)+
  geom_label(label=paste(rsq,r2), x=2012, y=40)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



>best model is life_expectancy~ hiv_aids+thin_5to9_years+polio+diphtheria+bmi+gdp+percentage_expenditure+schooli
>mse is 19.67224

##All Developed >The lowest MSE average from all 10 seeds was found using the OLS method.

```
#removed hiv_aids due to errors

lm1 = lm(life_expectancy ~.-hiv_aids, trained)
slr_MSE_test1
```

```
## # A tibble: 1 x 1
##   slr_MSE_test1
##           <dbl>
## 1          7.57
```

```
car::vif(lm1)
```

```
##              population                alcohol        thin_5to9_years
##                1.365911               1.101269              75.944711
##       thin_10to19_years            hepatitis_b                measles
##               73.969837               1.424282               1.353066
##                   polio             diphtheria                    bmi
##                2.124567               2.159218               1.107501
##       under_five_deaths      total_expenditure                    gdp
##                1.633443               1.131464               6.087966
## percentage_expenditure              schooling
##                6.092383               1.266894
```

```r
summary(lm1)
```

```
##
## Call:
## lm(formula = life_expectancy ~ . - hiv_aids, data = trained)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7130 -1.9625 -0.4352  1.0678  9.4173
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)              7.731e+01  2.681e+00  28.836  < 2e-16 ***
## population              -4.491e-09  1.165e-08  -0.385 0.700194
## alcohol                 -2.832e-01  5.607e-02  -5.051 7.26e-07 ***
## thin_5to9_years         -4.584e-01  1.645e+00  -0.279 0.780642
## thin_10to19_years       -2.636e+00  1.759e+00  -1.499 0.134926
## hepatitis_b              5.012e-03  5.081e-03   0.986 0.324693
## measles                  9.689e-06  5.681e-05   0.171 0.864670
## polio                   -1.133e-03  2.219e-02  -0.051 0.959289
## diphtheria               3.945e-02  2.047e-02   1.927 0.054805 .
## bmi                     -1.514e-02  9.370e-03  -1.616 0.107057
## under_five_deaths        3.557e-01  1.522e-01   2.337 0.020012 *
## total_expenditure       -1.604e-01  5.939e-02  -2.701 0.007264 **
## gdp                      1.352e-05  1.676e-05   0.807 0.420478
## percentage_expenditure   9.971e-05  9.746e-05   1.023 0.306999
## schooling                3.755e-01  1.075e-01   3.491 0.000546 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.848 on 332 degrees of freedom
## Multiple R-squared:  0.5253, Adjusted R-squared:  0.5052
## F-statistic: 26.24 on 14 and 332 DF,  p-value: < 2.2e-16
```

```r
#removed hiv_aids due to errors

lm1 = lm(life_expectancy ~.-hiv_aids-thin_5to9_years, trained)
tested = tested %>%
  mutate(predictions = predict(lm1, tested))

slr_MSE_test1 = tested %>%
  summarize(slr_MSE_test1 = mean((life_expectancy-predictions)^2))
slr_MSE_test1
```

```
## # A tibble: 1 x 1
##   slr_MSE_test1
##           <dbl>
## 1          7.57
```

```r
car::vif(lm1)
```

```
##             population                alcohol      thin_10to19_years
##              1.354168               1.094926               1.476411
##           hepatitis_b                measles                  polio
##              1.326204               1.297892               2.117529
##            diphtheria                    bmi      under_five_deaths
##              2.157446               1.091740               1.624915
##     total_expenditure                    gdp percentage_expenditure
##              1.130500               6.057969               6.060323
##             schooling
##              1.241761
```

```
summary(lm1)
```

```
##
## Call:
## lm(formula = life_expectancy ~ . - hiv_aids - thin_5to9_years,
##     data = trained)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.663 -1.962 -0.468  1.056  9.296
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             7.730e+01  2.677e+00  28.878  < 2e-16 ***
## population             -4.792e-09  1.159e-08  -0.414 0.679444
## alcohol                -2.844e-01  5.583e-02  -5.094 5.88e-07 ***
## thin_10to19_years      -3.121e+00  2.481e-01 -12.578  < 2e-16 ***
## hepatitis_b             4.640e-03  4.897e-03   0.948 0.343988
## measles                 1.289e-05  5.556e-05   0.232 0.816728
## polio                  -7.775e-04  2.212e-02  -0.035 0.971982
## diphtheria              3.962e-02  2.043e-02   1.939 0.053385 .
## bmi                    -1.545e-02  9.290e-03  -1.663 0.097182 .
## under_five_deaths       3.526e-01  1.516e-01   2.327 0.020588 *
## total_expenditure      -1.600e-01  5.929e-02  -2.698 0.007332 **
## gdp                     1.319e-05  1.670e-05   0.790 0.430048
## percentage_expenditure  1.017e-04  9.707e-05   1.048 0.295611
## schooling               3.797e-01  1.063e-01   3.571 0.000408 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.844 on 333 degrees of freedom
## Multiple R-squared:  0.5251, Adjusted R-squared:  0.5066
## F-statistic: 28.33 on 13 and 333 DF,  p-value: < 2.2e-16
```

```
#removed hiv_aids due to run error.

lm1 = lm(life_expectancy ~.-hiv_aids-thin_5to9_years-percentage_expenditure-total_expenditure, trained)
tested = tested %>%
  mutate(predictions = predict(lm1, tested))

slr_MSE_test1 = tested %>%
  summarize(slr_MSE_test1 = mean((life_expectancy-predictions)^2))
slr_MSE_test1
```

```
## # A tibble: 1 x 1
##   slr_MSE_test1
##           <dbl>
## 1          7.38
```

```
car::vif(lm1)
```

```
##        population           alcohol thin_10to19_years       hepatitis_b
##          1.343033          1.069199          1.421890          1.294455
##           measles             polio        diphtheria               bmi
##          1.282356          2.116889          2.126985          1.090565
## under_five_deaths               gdp         schooling
##          1.600479          1.213680          1.217514
```

```
summary(lm1)
```

```
##
## Call:
## lm(formula = life_expectancy ~ . - hiv_aids - thin_5to9_years -
##     percentage_expenditure - total_expenditure, data = trained)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6103 -1.9126 -0.5398  1.1282 10.1674
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.700e+01  2.692e+00  28.598  < 2e-16 ***
## population        -3.009e-09  1.164e-08  -0.258 0.796251
## alcohol           -2.657e-01  5.567e-02  -4.773 2.72e-06 ***
## thin_10to19_years -3.005e+00  2.457e-01 -12.232  < 2e-16 ***
## hepatitis_b        2.779e-03  4.881e-03   0.569 0.569540
## measles           -3.976e-06  5.573e-05  -0.071 0.943158
## polio              3.111e-04  2.232e-02   0.014 0.988887
## diphtheria         3.274e-02  2.047e-02   1.599 0.110677
## bmi               -1.464e-02  9.369e-03  -1.562 0.119152
## under_five_deaths  4.049e-01  1.518e-01   2.668 0.008002 **
## gdp                2.910e-05  7.542e-06   3.858 0.000137 ***
## schooling          3.380e-01  1.062e-01   3.182 0.001600 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.87 on 335 degrees of freedom
## Multiple R-squared:  0.5136, Adjusted R-squared:  0.4976
## F-statistic: 32.16 on 11 and 335 DF,  p-value: < 2.2e-16
```

```
#removed hiv_aids due to errors

lm1 = lm(life_expectancy ~.-hiv_aids-thin_5to9_years-percentage_expenditure-total_expenditure-polio, tra
tested = tested %>%
  mutate(predictions = predict(lm1, tested))
```

```
slr_MSE_test1 = tested %>%
  summarize(slr_MSE_test1 = mean((life_expectancy-predictions)^2))
slr_MSE_test1
```

```
## # A tibble: 1 x 1
##   slr_MSE_test1
##           <dbl>
## 1          7.38
```

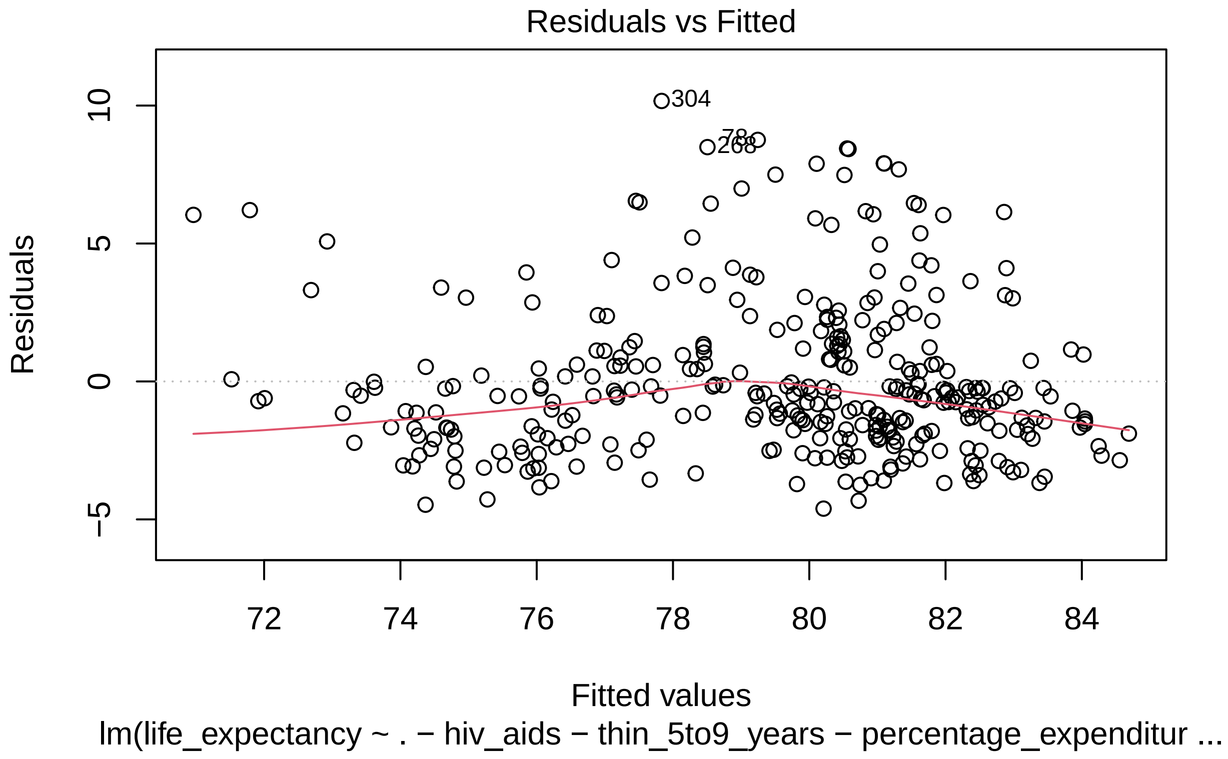
```
car::vif(lm1)
```

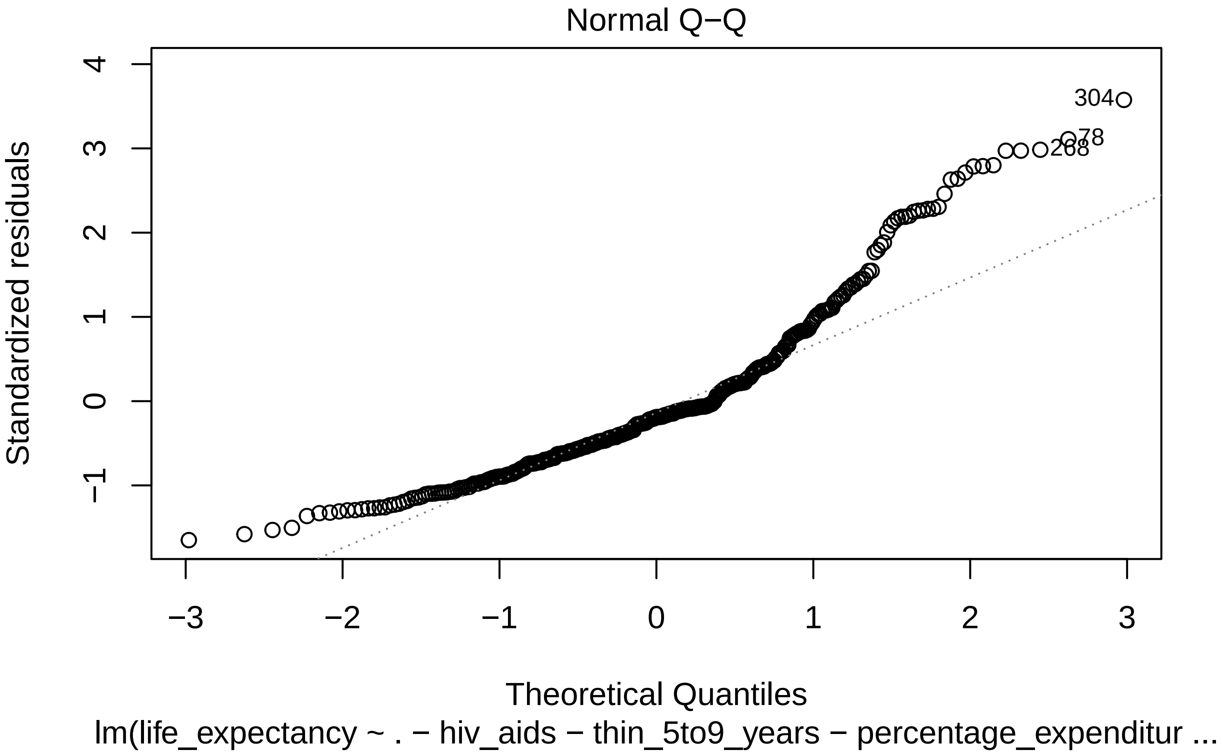
```
##        population           alcohol thin_10to19_years       hepatitis_b
##          1.342997          1.068705          1.420915          1.291456
##           measles        diphtheria               bmi under_five_deaths
##          1.279534          1.050257          1.089271          1.593079
##               gdp         schooling
##          1.209650          1.216850
```

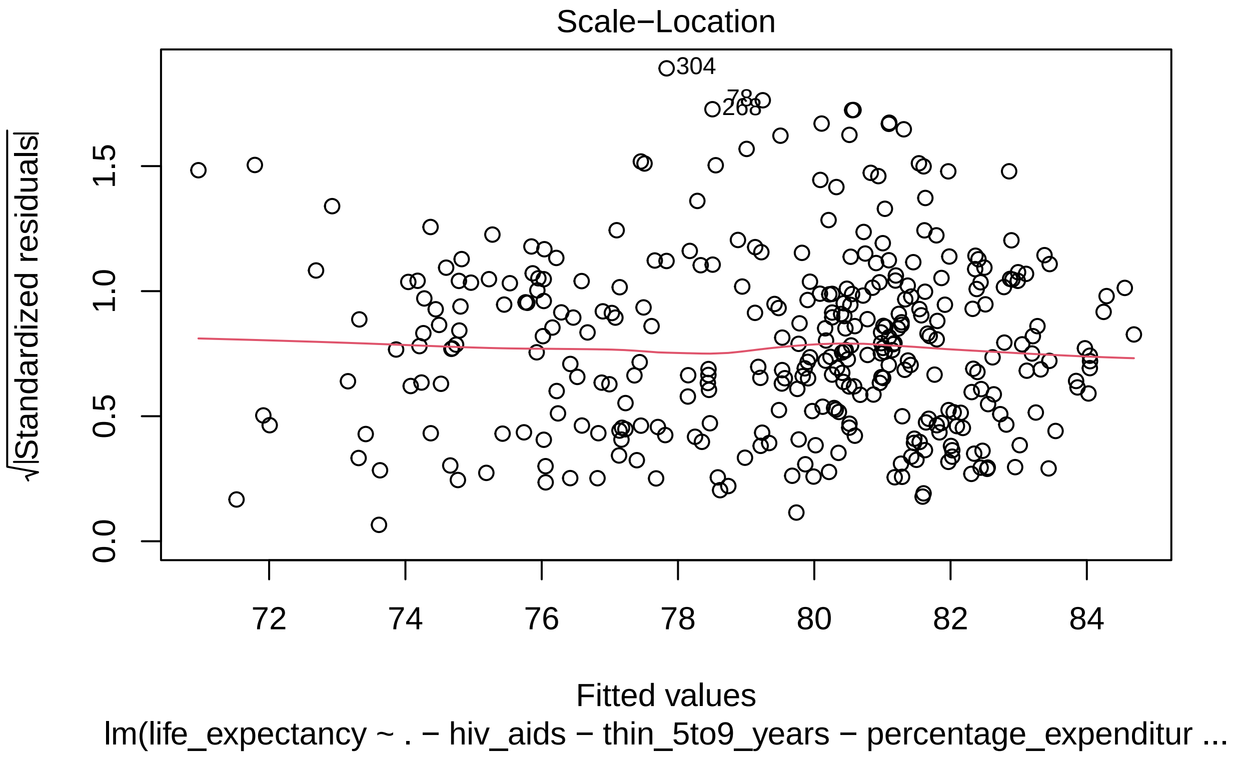
```
summary(lm1)
```

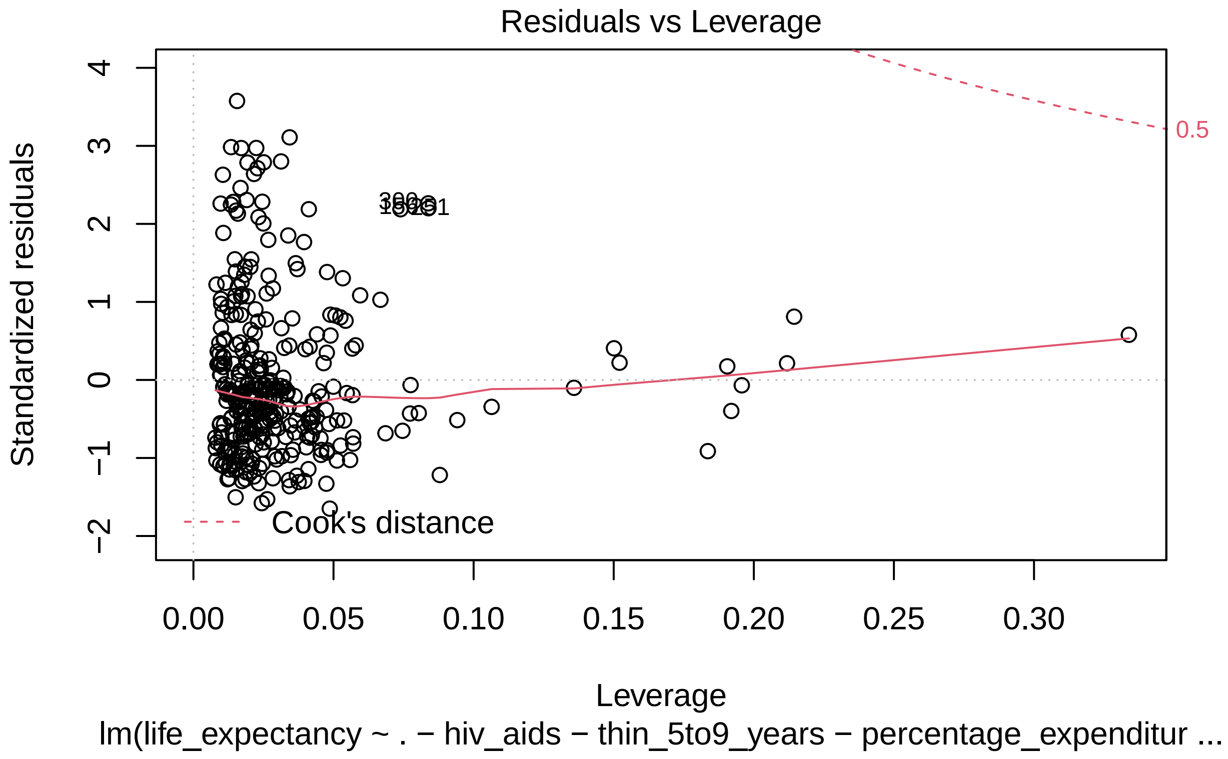
```
##
## Call:
## lm(formula = life_expectancy ~ . - hiv_aids - thin_5to9_years -
##     percentage_expenditure - total_expenditure - polio, data = trained)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6099 -1.9126 -0.5396  1.1283 10.1667
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.701e+01  2.590e+00  29.731  < 2e-16 ***
## population        -3.008e-09  1.163e-08  -0.259  0.79601
## alcohol           -2.657e-01  5.558e-02  -4.781 2.61e-06 ***
## thin_10to19_years -3.005e+00  2.452e-01 -12.254  < 2e-16 ***
## hepatitis_b        2.782e-03  4.868e-03   0.571  0.56806
## measles           -4.013e-06  5.558e-05  -0.072  0.94249
## diphtheria         3.295e-02  1.436e-02   2.294  0.02243 *
## bmi               -1.463e-02  9.350e-03  -1.565  0.11850
## under_five_deaths  4.051e-01  1.512e-01   2.679  0.00775 **
## gdp                2.910e-05  7.518e-06   3.871  0.00013 ***
## schooling          3.380e-01  1.061e-01   3.187  0.00157 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.866 on 336 degrees of freedom
## Multiple R-squared:  0.5136, Adjusted R-squared:  0.4991
## F-statistic: 35.48 on 10 and 336 DF,  p-value: < 2.2e-16
```

```
plot(lm1)
```


Residuals vs Fitted
Residuals
Fitted values
lm(life_expectancy ~ . − hiv_aids − thin_5to9_years − percentage_expenditur ...
304
78
268
10
5
0
−5
72
74
76
78
80
82
84


Normal Q−Q
Standardized residuals
4
3
2
1
0
−1
304
78
268
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(life_expectancy ~ . − hiv_aids − thin_5to9_years − percentage_expenditur ...


Scale−Location
304
78
268
√|Standardized residuals|
0.0
0.5
1.0
1.5
72
74
76
78
80
82
84
Fitted values
lm(life_expectancy ~ . − hiv_aids − thin_5to9_years − percentage_expenditur ...

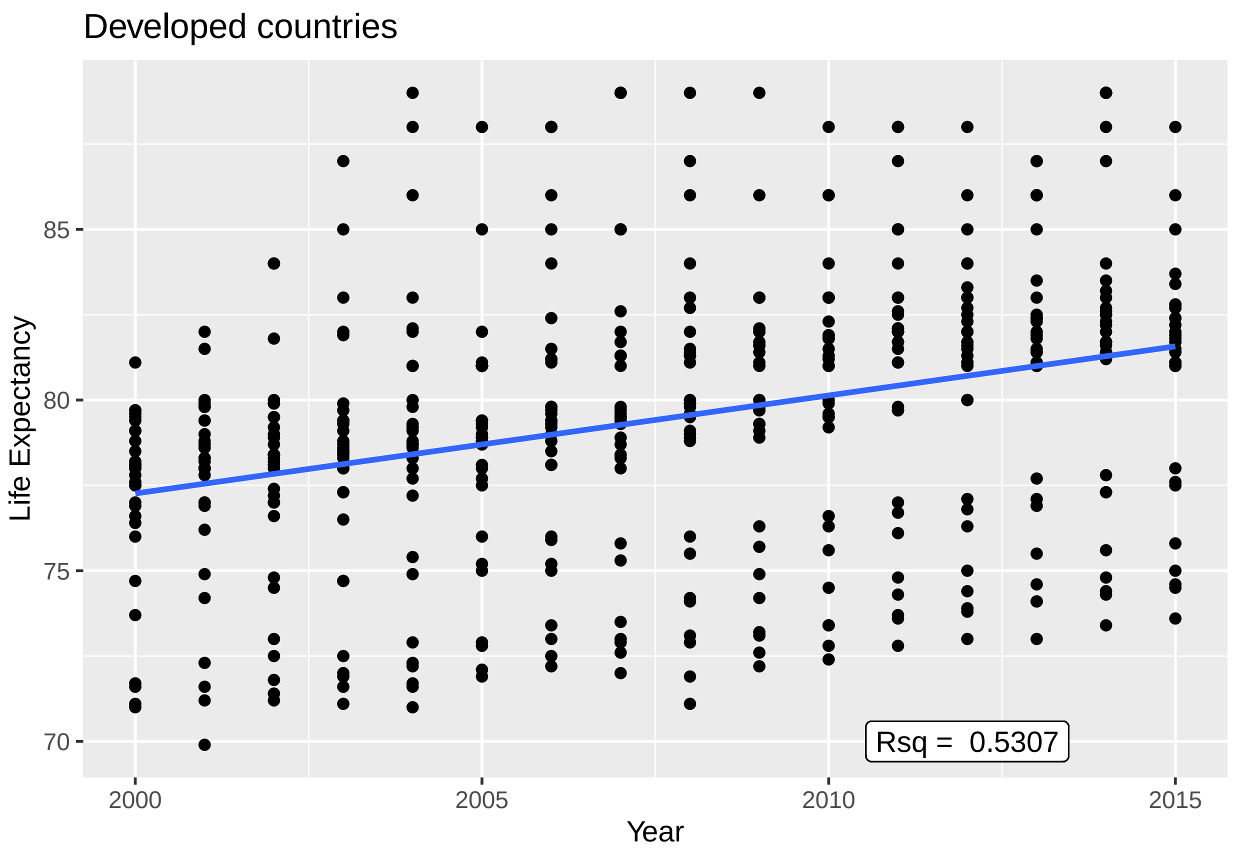


```
lm = lm(life_expectancy ~population+alcohol+thin_10to19_years+hepatitis_b+measles+polio+diphtheria+bmi+
modsum =summary(lm)
r2 = modsum$r.squared
r2 = round(r2, digits=4)
r2
```

```
## [1] 0.5307
```

```
rsq = "Rsq = "
ggplot(le_developed,aes(x=year,y=life_expectancy))+
  geom_point()+
  labs(title= "Developed countries", y="Life Expectancy", x = "Year")+
  geom_smooth(se=FALSE, method="lm")+
  #geom_label(label=r2, x=2012, y=40,)+
  geom_label(label=paste(rsq,r2), x=2012, y=70)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Best variable selection is life_expectancy ~ population + alcohol + thin_10to19_years +hepatitis_b+measles+diphtheria+bmi+under_five_deaths+gdp_schooling if there is absolute colinearity.

Otherwise the variuable selection is life_expectancy~population+alcohol+thin_10to19_years+hepatitis_b+measles+p

##All Developing

The lowest MSE average from all 10 seeds was found using the Best Subset with Cross-Validation method.

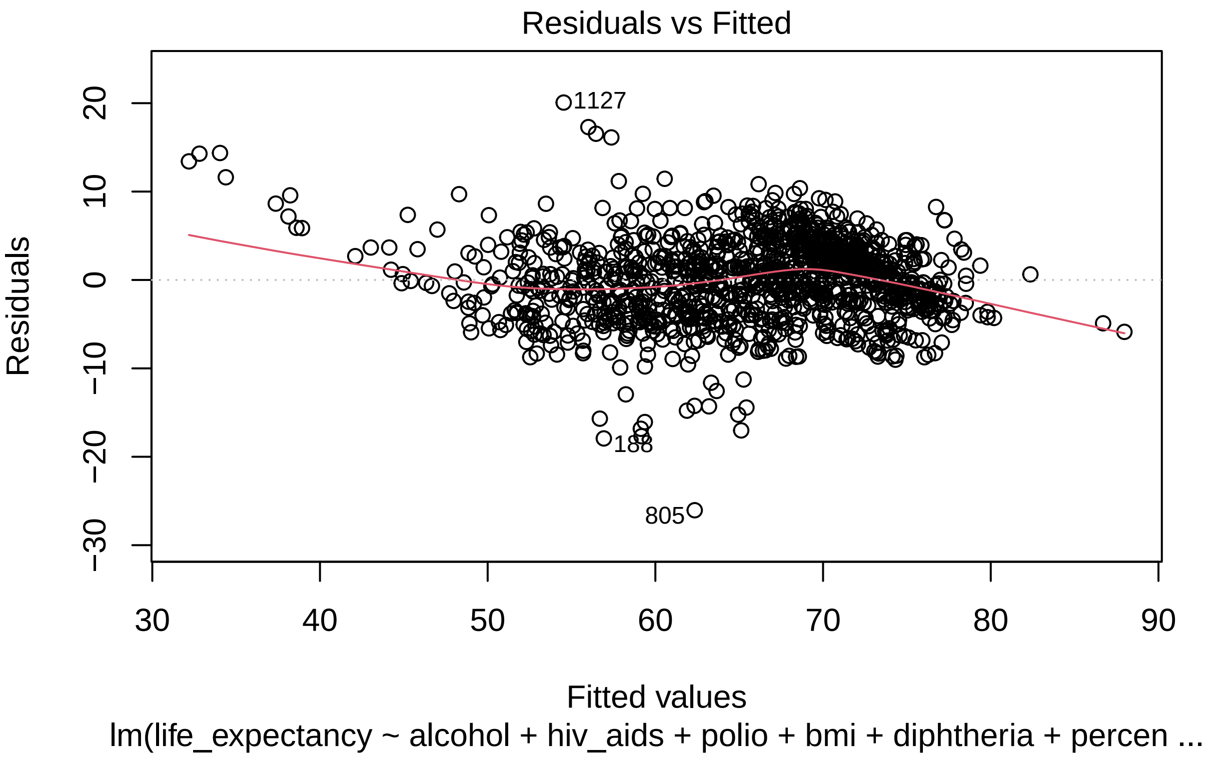
```
lm2 = lm(life_expectancy ~ alcohol + hiv_aids  + polio + bmi + diphtheria + percentage_expenditure + sc

car::vif(lm2)
```

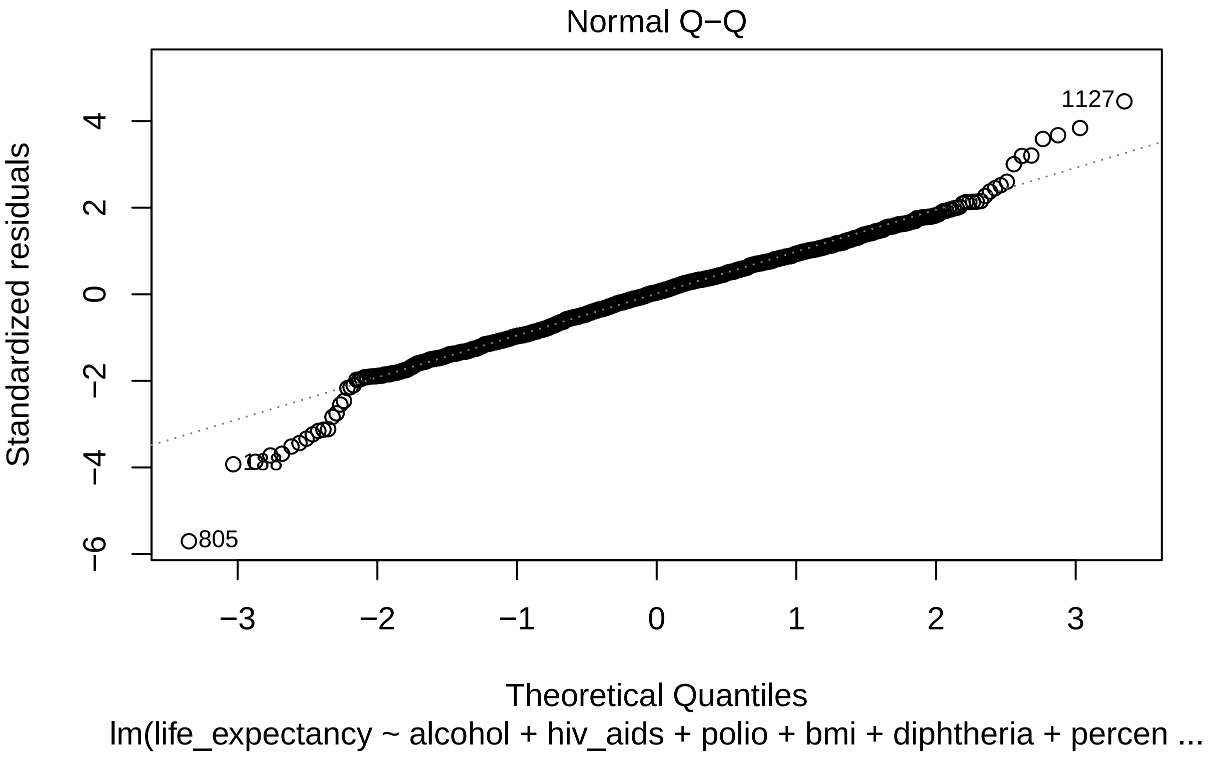
```
##                alcohol               hiv_aids                  polio
##               1.256671               1.093068               1.740806
##                    bmi             diphtheria percentage_expenditure
##               1.452100               1.799922               1.207542
##              schooling
##               1.965378
```

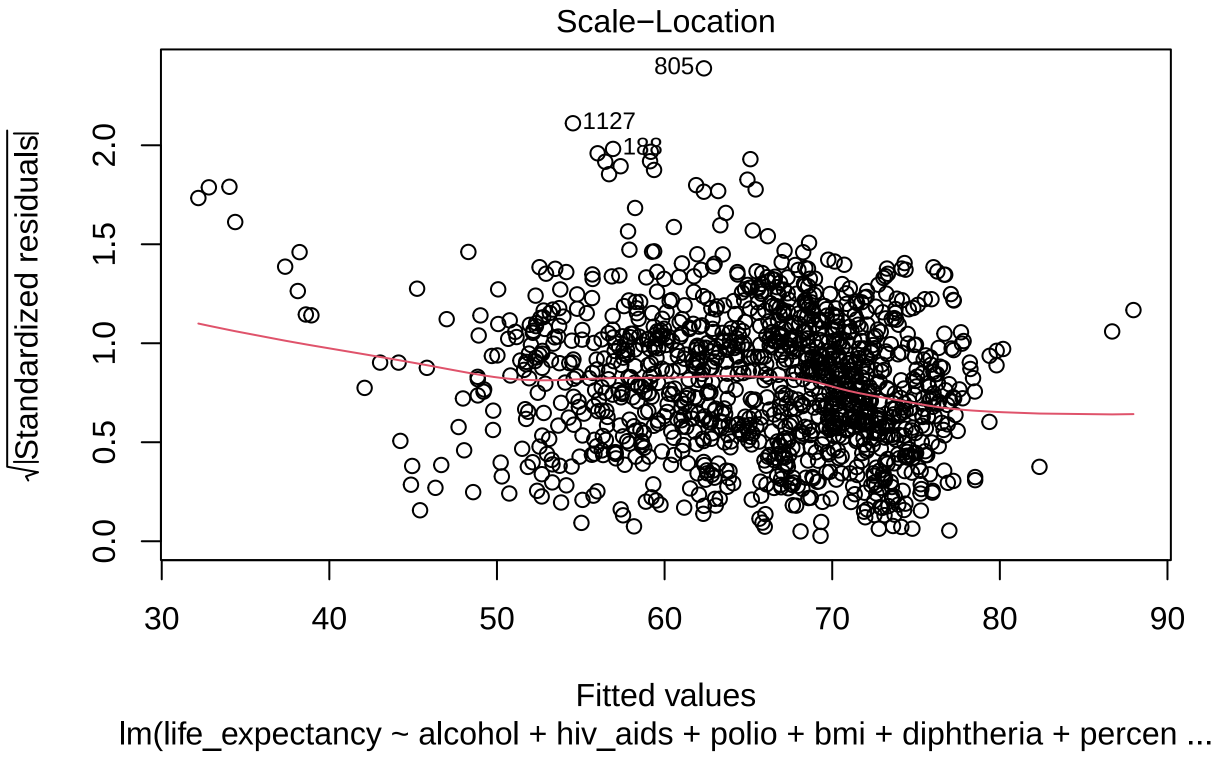
```
bscv_MSE_test2
```

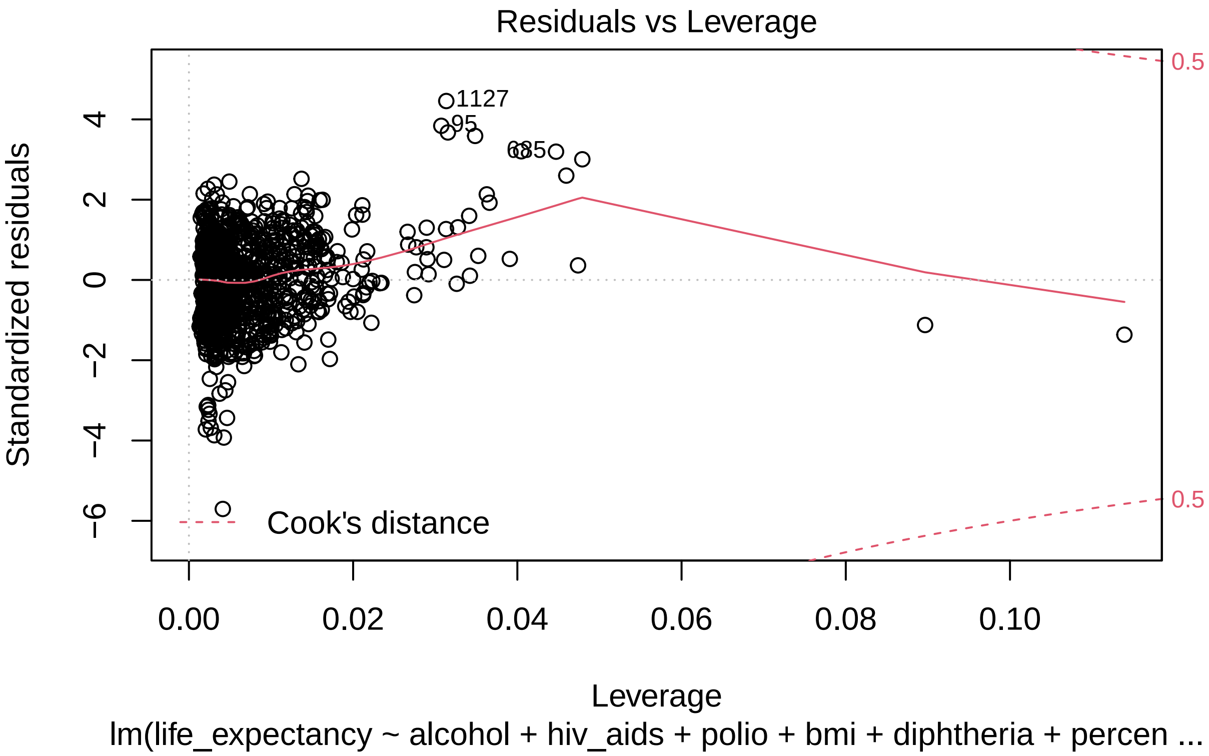
```
## # A tibble: 1 x 1
##   bscv_MSE_test2
##            <dbl>
## 1           18.4
```

```
plot(lm2)
```




Normal Q−Q
Standardized residuals
4
2
0
−2
−4
−6
1127
805
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(life_expectancy ~ alcohol + hiv_aids + polio + bmi + diphtheria + percen ...


Scale−Location
805
1127
188
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
30
40
50
60
70
80
90
Fitted values
lm(life_expectancy ~ alcohol + hiv_aids + polio + bmi + diphtheria + percen ...

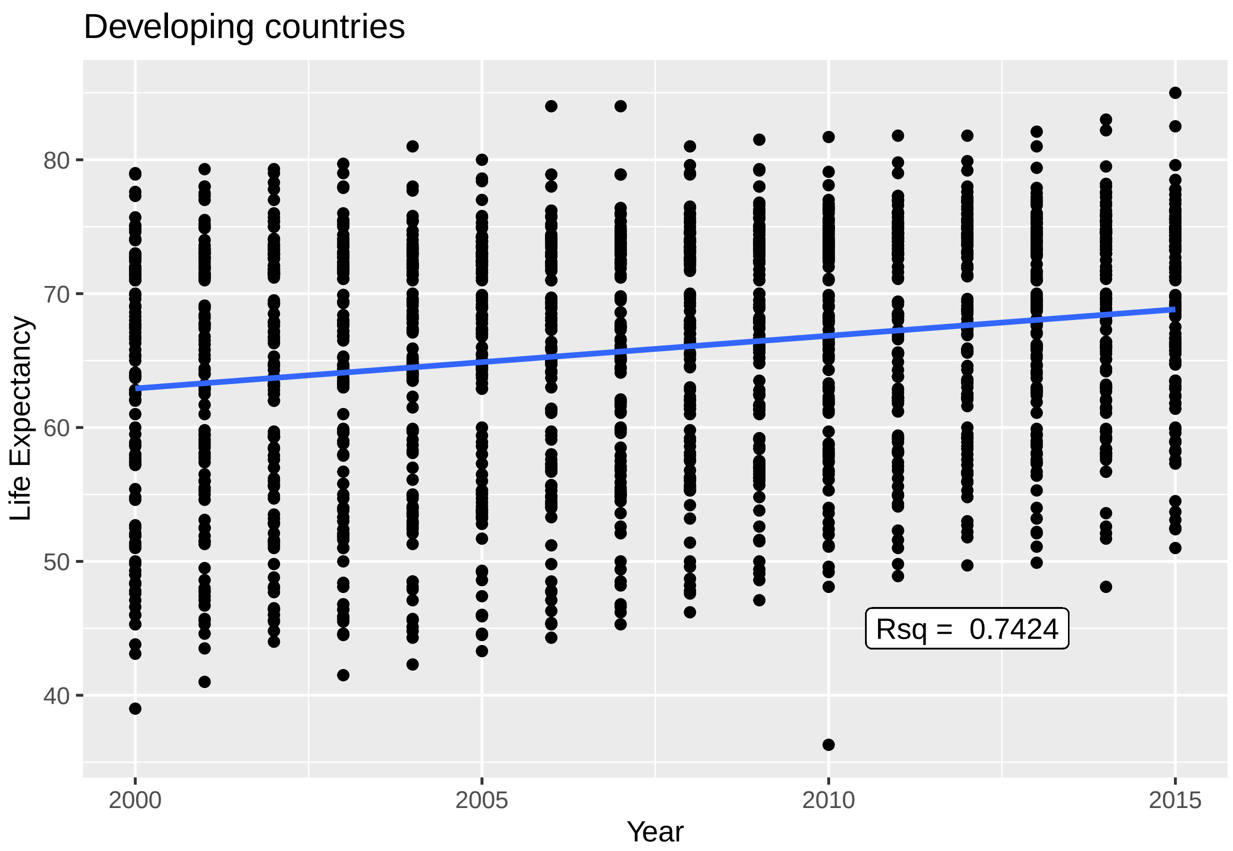


```
lm = lm(life_expectancy ~ alcohol+hiv_aids+polio+bmi+diphtheria+percentage_expenditure+schooling, le_de
modsum =summary(lm)
r2 = modsum$r.squared
r2 = round(r2, digits=4)
r2
```

```
## [1] 0.7424
```

```
rsq = "Rsq = "
ggplot(le_developing,aes(x=year,y=life_expectancy))+
  geom_point()+
  labs(title= "Developing countries", y="Life Expectancy", x = "Year")+
  geom_smooth(se=FALSE, method="lm")+
  #geom_label(label=r2, x=2012, y=40,)+
  geom_label(label=paste(rsq,r2), x=2012, y=45)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



all vif for variables are low

variable selection for developing countries is: life_expectancy ~ alcohol + hiv_aids + polio + bmi + diphtheria + percentage_expenditure + schooling

.