End-Semester Project

Bivariate Analysis on GDP per capita, Sanitation and Life Expectancy across Nations in 2010

Aman Das [BS2206] Raj Pratap Singh [BS2219] Shreyansh Mukhopadhyay [BS2147]

Table of contents

1 Introduction

1.1 Overview

This presentation demonstrates the capabilities of *Bivariate Analysis* on datasets, to infer relationship between various features of Nations.

- **log of GDP per capita**: Logarithm (base e) of Gross Domestic Product (in $) per citizen. Adjusted for Inflation. [*lngdp*]
- **Sanitation Access %**: Percentage of people using at least basic Sanitation facilities, not shared with other households. [*snt*]
- **Life Expectancy**: The average number of years a newly born child would live, provided current mortality patterns hold. [*lfx*]

1.2 Data

```r
script.dir <- getSrcDirectory(function(x) {x})
setwd(script.dir)

numerise = function(x){
  x[grepl("k$", x)] <- as.numeric(sub("k$", "", x[grepl("k$", x)]))*10^3
  x <- as.numeric(x)
  return(x)
}

d1_raw = read.csv(file.path(".","Data","gdp.csv"), fileEncoding = 'UTF-8-BOM')
d2_raw = read.csv(file.path(".","Data","sanitation.csv"), fileEncoding = 'UTF-8-BOM')
d3_raw = read.csv(file.path(".","Data","life_expectancy.csv"), fileEncoding = 'UTF-8-BOM')

yearname = "X2010"

d1 = d1_raw[!is.na(numerise(d1_raw[, yearname])),][,c("country", yearname)]
colnames(d1)[2] = "lngdp"
d2 = d2_raw[!is.na(numerise(d2_raw[, yearname])),][,c("country", yearname)]
colnames(d2)[2] = "snt"
d3 = d3_raw[!is.na(numerise(d3_raw[, yearname])),][,c("country", yearname)]
colnames(d3)[2] = "lfx"

dtemp = merge(x = d1, y = d2, by = "country")
d = merge(x = dtemp, y = d3, by = "country")

d$lngdp = log(numerise(d$lngdp))

write.csv(d, "./Data/assembled.csv")

kable(head(d, 6L))
```

country	lngdp	snt	lfx
Afghanistan	6.265301	34.9	60.5
Albania	8.183118	95.2	78.1
Algeria	8.273847	87.0	74.5
Andorra	10.454495	100.0	81.8
Angola	8.291547	41.1	60.2
Antigua and Barbuda	9.546813	86.3	75.9

FREE DATA FROM UN, WORLD BANK, WHO, IMHE VIA GAPMINDER.ORG, CC-BY LICENSE.

2 Univariate Statistics

2.1 Measures of Central Tendency

Mean or Arithmetic Mean $\bar{x}$, *Geometric Mean* $\mathrm{GM}(x)$, *Harmonic Mean* $\mathrm{HM}(x)$, *Median* $\mathrm{median}(x)$ and *Mode* $\mathrm{mode}(x)$ are some measures of *central tendency* in the sample.

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n}(x_i) \quad \mathrm{GM}(x) = \sqrt[n]{\prod_{i=1}^{n} a_i} \quad \mathrm{HM}(x) = n\sum_{i=1}^{n} x_i^{-1}$$

$$\mathrm{median}(x) = \begin{cases} x_{(n+1)/2} & : n = 1 \mod 2 \\ \frac{x_{(n/2)}+x_{((n/2)+1)}}{2} & : n = 0 \mod 2 \end{cases} \quad \mathrm{mode}(x) = x_{(n)}$$

```
getmode <- function(v) {
 uniqv <- unique(v)
 freq = max(tabulate(match(v, uniqv)))
 res = uniqv[which.max(tabulate(match(v, uniqv)))]
 if (freq == 1) res = NULL
 return(res)
}

d_central = data.frame(
  row.names = "Variable",
  Variable = c(
    "*ln(GDP)*",
    "*Sanitation*",
    "*Life Exp.*"
  ),
  Mean = c(
    mean(d$lngdp),
    mean(d$snt),
    mean(d$lfx)
  ),
  GM = c(
```

```r
    geometric.mean(d$lngdp),
    geometric.mean(d$snt),
    geometric.mean(d$lfx)
  ),
  HM = c(
    harmonic.mean(d$lngdp),
    harmonic.mean(d$snt),
    harmonic.mean(d$lfx)
  ),
  Median = c(
    median(d$lngdp),
    median(d$snt),
    median(d$lfx)
  ),
  Mode = c(
    getmode(d$lngdp),
    getmode(d$snt),
    getmode(d$lfx)
  )
)

kable(
  d_central,
  col.names = c(
    "$\\bar{x}$",
    "$\\operatorname{GM}(x)$",
    "$\\operatorname{HM}(x)$",
    "$\\operatorname{median}(x)$",
    "$\\operatorname{mode}(x)$"
  ),
  digits=5
)
```

	$\bar{x}$	$\operatorname{GM}(x)$	$\operatorname{HM}(x)$	$\operatorname{median}(x)$	$\operatorname{mode}(x)$
ln(GDP)	8.54124	8.42229	8.30248	8.48673	9.23014
Sanitation	72.43857	62.58904	47.61862	85.60000	100.00000
Life Exp.	70.54603	69.95538	69.28316	72.40000	73.20000

2.2 Measures of Dispersion

Range(x), Semi-int.. SIR(x), Mean Deviation about x' $\operatorname{MD}_{(x')}(x)$, Variance s_x^2, Standard Deviation s_x are some measures of *dispersion* in the sample.

Note: x_i is the ith observation. $x_{(i)}$ is the ith largest observation.

$$\mathrm{Range}(x) = |x_{(n)} - x_{(1)}| \qquad Q_1 = \mathrm{median}(x_{(1)}, \ldots, x_{(\lfloor \frac{n+1}{2} \rfloor)}) \qquad Q_3 = \mathrm{median}(x_{(\lfloor \frac{n+2}{2} \rfloor)}, \ldots, x_{(n)})$$

$$\mathrm{MD}_{(x')}(x) = \frac{\sum_{i=1}^{n} |x_i - x'|}{n} \qquad \mathrm{SIR}(x) = \frac{|Q_1 - Q_3|}{2} \qquad s_x = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n}} \qquad s_x^2 = (s_x)^2$$

```
getmd = function(x, center = mean(x)){
  md = mean(
    abs(
      x - rep(center, length(x))
      )
    )
  return(md)
}
d_disp = data.frame(
  row.names = "Variable",
  Variable = c(
    "*ln(GDP)*",
    "*Sanitation*",
    "*Life Exp.*"
  ),
  Range = c(
    max(d$lngdp) - min(d$lngdp),
    max(d$snt) - min(d$snt),
    max(d$lfx) - min(d$lfx)
  ),
  SIR = c(
    IQR(d$lngdp)/2,
    IQR(d$snt)/2,
    IQR(d$lfx)/2
  ),
  MD = c(
    getmd(d$lngdp),
    getmd(d$snt),
    getmd(d$lfx)
  ),
  variance = c(
    (sd(d$lngdp))^2,
    (sd(d$snt))^2,
    (sd(d$lfx))^2
  ),
  SD = c(
    sd(d$lngdp),
    sd(d$snt),
    sd(d$lfx)
```

```
  )
)

kable(
  d_disp,
  col.names = c(
    "$\\operatorname{Range}(x)$",
    "$\\operatorname{SIR}(x)$",
    "$\\operatorname{MD}_{(\\bar{x})}(x)$",
    "$\\quad \\quad \\quad \\quad s_x^2$",
    "$\\quad \\quad \\quad \\quad s_x$"
  ),
  digits=5
)
```

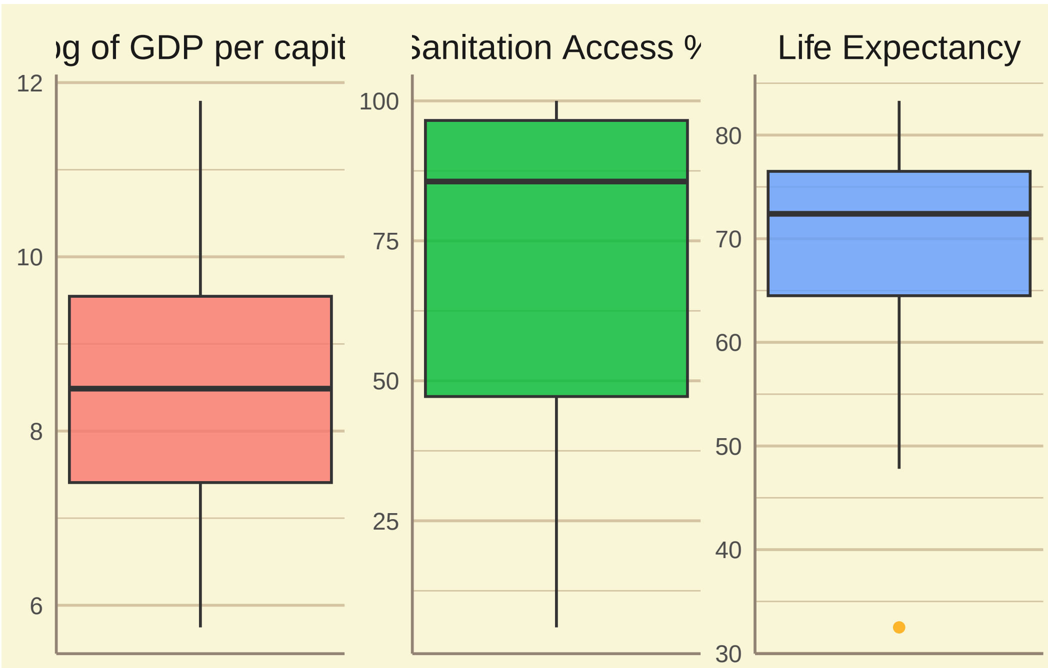
	$\operatorname{Range}(x)$	$\operatorname{SIR}(x)$	$\operatorname{MD}_{(\bar{x})}(x)$	s_x^2	s_x
ln(GDP)	6.04435	1.06914	1.17229	2.01791	1.42053
Sanitation	94.03000	24.65000	25.50487	872.29346	29.53461
Life Exp.	50.80000	6.00000	6.98712	75.33494	8.67957

2.3 Box Plot

About?

```
labelfunction = function(val1){
  return(list(c(
    "log of GDP per capita",
    "Sanitation Access %",
    "Life Expectancy"
  )))
}
ggplot(stack(d[2:4]), mapping = aes(y = values))+
geom_boxplot(aes(fill=ind), alpha=0.8, outlier.color = "orange")+
labs(
  x=NULL,
  y=NULL
  )+
mytheme+
mycolor+
facet_wrap(~ind, scales="free", labeller = labelfunction)+
  theme(axis.text.x=element_blank(),
        legend.position="none",
```

```
        strip.text.x = element_text(size = rel(1.5)),
        panel.grid.minor.x = element_blank(),
        panel.grid.major.x = element_blank()
  )
```



2.4 Inferences

3 Scatter Plot

A *Scatter plot* is a type of Plot using Cartesian coordinate system to display values for two variables for a set of data. The data are displayed as a collection of points, each having one variable determining the *abscissa* and the other variable determining the *ordinate*. It helps us:

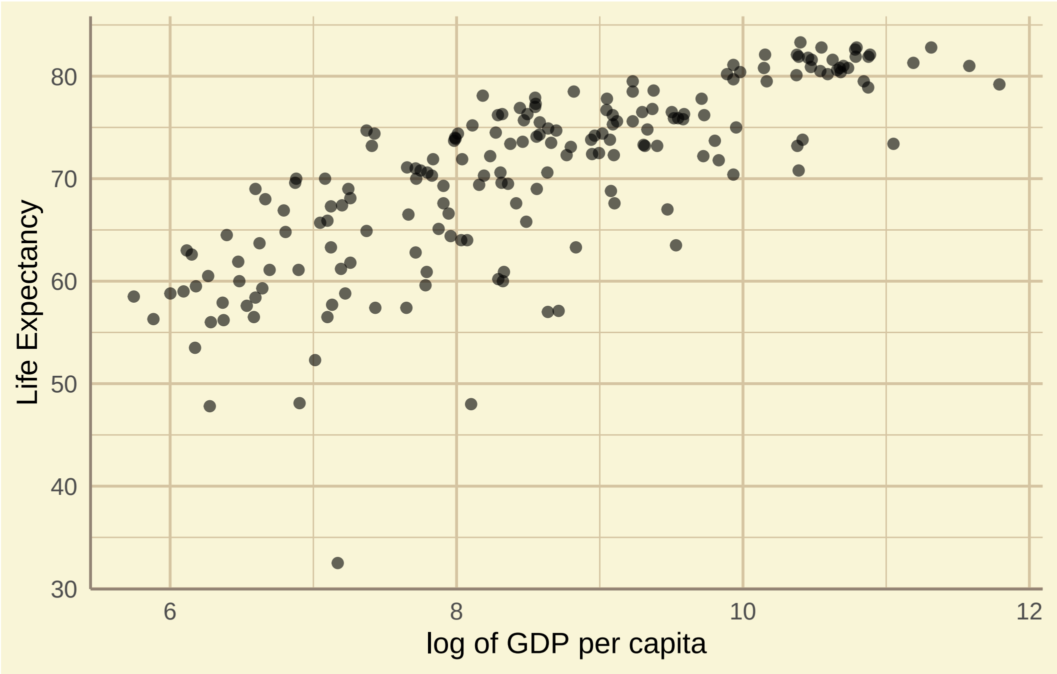
- take a short glance at effect of two variables.
- suggest kinds of correlations between variables.
- estimate the direction of correlation.

```
sctrplot = function(
    d, x_map, y_map,
    x_lab=waiver(), y_lab=waiver(),
    title=waiver()
){
```

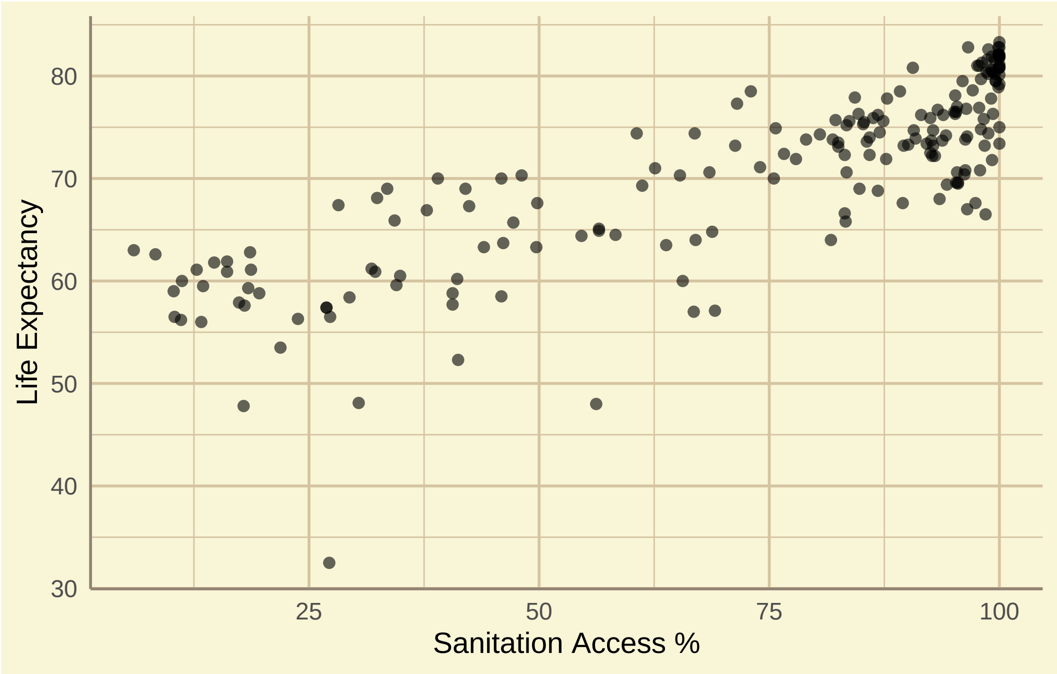
```
  plot1 = ggplot(d, mapping = aes(x = x_map, y = y_map))+
    geom_point(
      alpha=0.6
    )+
    mytheme+
    labs(
      x=x_lab,
      y=y_lab,
      title=title
    )

  return(plot1)
}
```

3.1 Sanitation vs. GDP per Capita

3.2 Life Expectancy vs. GDP per Capita



3.3 Life Expectation vs. Sanitation



3.4 Inferences

seems like Linear correlation

4 Bivariate Statistics

4.1 Covariance and Correlation Matrices

Covariance $\mathrm{cov}(x, y)$ is a measure of the joint variability of two random variables x, y.

Correlation $r_{x,y}$ is any relationship, causal or spurious, between two random variables x, y. *Pearson's* r correlation coefficient is considered here.

$$\mathrm{cov}(x, y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n} \quad r_{x,y} = \frac{\mathrm{cov}(x, y)}{s_x s_x}$$

```
cov_mat = cov(d[, 2:4])

kable(cov_mat, digits=5)
```

	lngdp	snt	lfx
lngdp	2.01791	33.84045	9.52202
snt	33.84045	872.29346	208.54155
lfx	9.52202	208.54155	75.33494

$$A_{i,j} = \text{cov}(x_i, x_j)$$

```
cor_mat = cor(d[, 2:4])

kable(cor_mat, digits=5)
```

	lngdp	snt	lfx
lngdp	1.00000	0.80659	0.77229
snt	0.80659	1.00000	0.81351
lfx	0.77229	0.81351	1.00000

$$A_{i,j} = r_{x_i,x_j}$$

4.2 Other Correlation Coefficients

Pearson, *Spearman*, *Kendall* #TODO

```
d_cor = data.frame(
  row.names = "Variable",
  Variable = c(
    "*Sanitation vs. ln(GDP)*",
    "*Life Exp. vs. ln(GDP)*",
    "*Life Exp. vs. Sanitation*"
  ),
  Pearson = c(
    cor(d$snt, d$lngdp, method="pearson"),
    cor(d$lfx, d$lngdp, method="pearson"),
    cor(d$lfx, d$snt, method="pearson")
  ),
  Spearman = c(
    cor(d$snt, d$lngdp, method="spearman"),
    cor(d$lfx, d$lngdp, method="spearman"),
    cor(d$lfx, d$snt, method="spearman")
  ),
```

```r
  Kendall = c(
    cor(d$snt, d$lngdp, method="kendall"),
    cor(d$lfx, d$lngdp, method="kendall"),
    cor(d$lfx, d$snt, method="kendall")
  )
)

kable(
  d_cor,
  digit = 5,
  col.names = c(
    "*Pearson's* $r$",
    "*Spearman's* $r_s$",
    "*Kendall's* $\\tau$"
  )
)
```

	Pearson's r	*Spearman's* r_s	*Kendall's* τ
Sanitation vs. ln(GDP)	0.80659	0.85920	0.67458
Life Exp. vs. ln(GDP)	0.77229	0.81639	0.62168
Life Exp. vs. Sanitation	0.81351	0.83513	0.63744

4.3 Partial Correlation

Partial

	Partial Correlation
Sanitation vs. ln(GDP)	0.4826925
Life Exp. vs. ln(GDP)	0.3377892
Life Exp. vs. Sanitation	0.5075384

4.4 Inferences

Good linear correlation lets try to observe line of best fit.

5 Linear Regression

Simple Univariate Linear Regression is a method for estimating the relationship $y_i = f(x_i)$ of a *response* variable y with a *predictor* variable x, as a line that closely fits the y vs. x *scatter plot*.

$$y_i = \hat{a} + \hat{b}x_i + e_i.$$

Where $\hat{a}$ is the *intercept*, $\hat{b}$ is the *slope*, and e_i is the ith residual *error*. We aim to minimize e_i for better fit.

5.1 Ordinary Least Squares

Ordinary Least squares method reduces e_i by minimizing *error sum of squares* $\sum e_i^2$.

```
olssmry = function(
    d, x_map, y_map,
    x_lab=waiver(), y_lab=waiver(),
    title=waiver()
){
  model = lm(formula=y_map~x_map)
  smry = summary(model, signif.stars=TRUE)

  smryvec = c(
    as.numeric(model$coefficients["(Intercept)"]),
    as.numeric(model$coefficients["x_map"]),
    smry$r.squared
  )

  return(smryvec)
}

olstab = t(data.frame(
  SvG = olssmry(d, d$lngdp, d$snt),
  LvG = olssmry(d, d$lngdp, d$lfx),
  LvS = olssmry(d, d$snt, d$lfx)
))

row.names(olstab) = c(
  "*Sanitation vs. ln(GDP)*",
  "*Life Exp. vs. ln(GDP)*",
  "*Life Exp. vs. Sanitation*"
)

kable(
  olstab,
  digit = 5,
  col.names=c(
  "$\\hat{a}$",
  "$\\hat{b}$",
  "$R^2$"
  )
)
```

	$\hat{a}$	$\hat{b}$	R^2
Sanitation vs. ln(GDP)	-70.79844	16.77006	0.65059
Life Exp. vs. ln(GDP)	30.24203	4.71876	0.59643
Life Exp. vs. Sanitation	53.22795	0.23907	0.66180

R^2: Coefficient of Determination

5.2 Least Absolute Deviation

Least absolute Deviation method reduces e_i by minimizing the *sum of absolute deviations* $\sum |e_i|$.

```
ladsmry = function(
    d, x_map, y_map,
    x_lab=waiver(), y_lab=waiver(),
    title=waiver()
){
  model = rq(formula=y_map~x_map)
  smry = summary(model)

  smryvec = c(
    as.numeric(model$coefficients[1]),
    as.numeric(model$coefficients[2])
  )

  return(smryvec)
}

olstab = t(data.frame(
  SvG = ladsmry(d, d$lngdp, d$snt),
  LvG = ladsmry(d, d$lngdp, d$lfx),
  LvS = ladsmry(d, d$snt, d$lfx)
))

row.names(olstab) = c(
  "*Sanitation vs. ln(GDP)*",
  "*Life Exp. vs. ln(GDP)*",
  "*Life Exp. vs. Sanitation*"
)

kable(
  olstab,
  digit = 5,
  col.names=c(
  "$\\hat{a}$",
```

```
    "$\\hat{b}$"
    )
  )
```

	$\hat{a}$	$\hat{b}$
Sanitation vs. ln(GDP)	-71.23153	16.80472
Life Exp. vs. ln(GDP)	31.99047	4.61340
Life Exp. vs. Sanitation	53.73041	0.23963

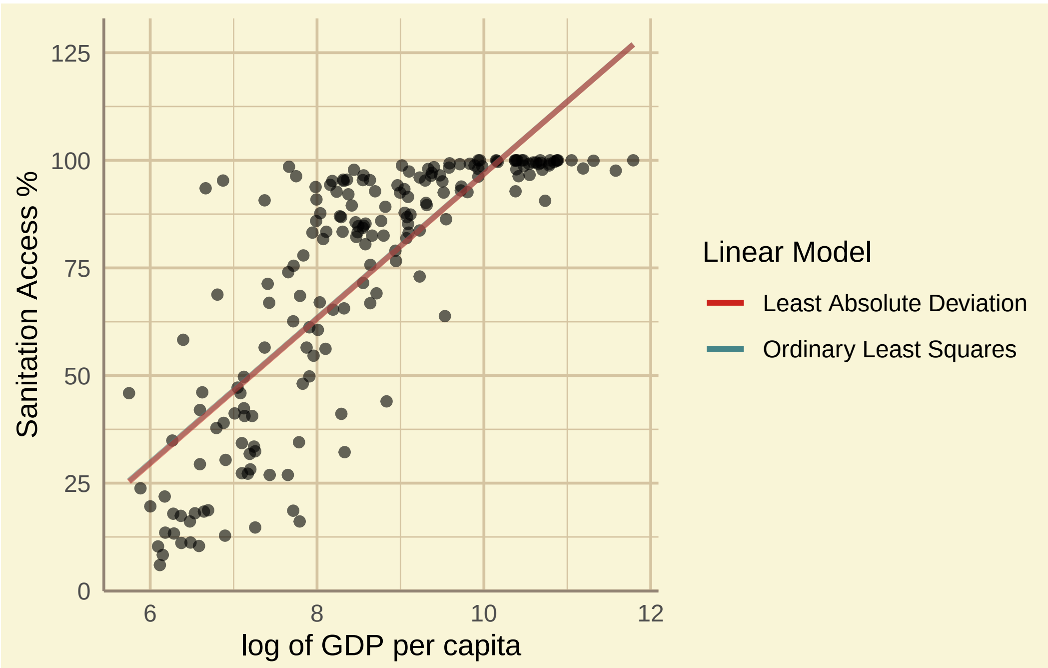
5.3 Line fitting

Plotting the estimated *Linear Model* on the Scatter Plot.

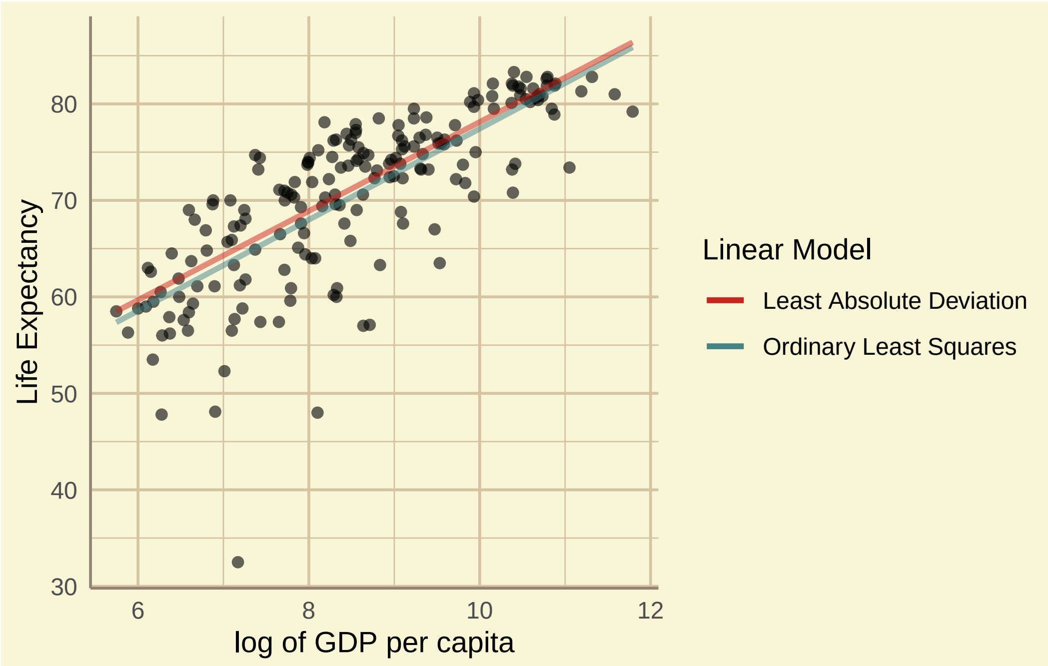
```
linearplot = function(
    d, x_map, y_map,
    x_lab=waiver(), y_lab=waiver(),
    title=waiver()
){
  plot1 = ggplot(d, mapping = aes(x = x_map, y = y_map))+
    geom_point(
      alpha=0.6
    )+
    mytheme+
    labs(
      x=x_lab,
      y=y_lab,
      title=title
    )+
    geom_smooth(
      method="lm",
      formula=y~x,
      se=FALSE,
      aes(color = "Ordinary Least Squares")
    )+
    geom_smooth(
      method="rq",
      formula=y~x,
      se=FALSE,
      aes(color = "Least Absolute Deviation")
    )+
    labs(
      color="Linear Model"
    )+
```

```
    mycolor

  return(plot1)
}
```

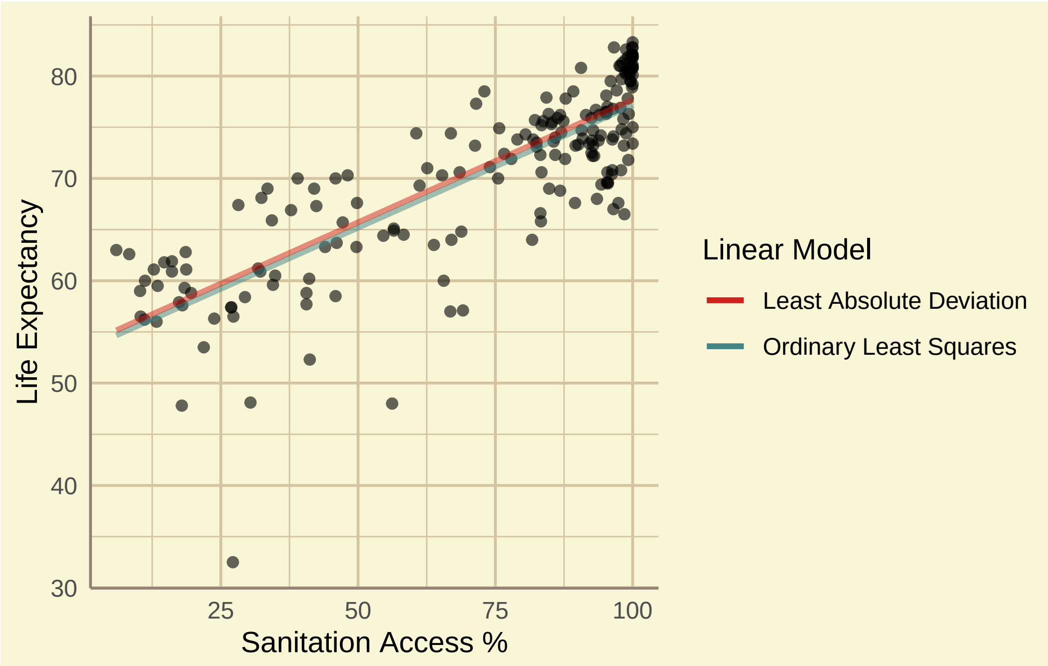
5.4 Sanitation vs. GDP per Capita



5.5 Life Expectancy vs. GDP per Capita



5.6 Life Expectancy vs. Sanitation



5.7 Inferences

6 Conclusion