Dataset used-

World Happiness Index(2019 dataset)

The World Happiness Report is a landmark survey of the state of global happiness. The first report was published in 2012, the second in 2013, the third in 2015, and the fourth in the 2016 Update.The World Happiness 2017, which ranks 155 countries by their happiness levels, was released at the United Nations at an event celebrating International Day of Happiness on March 20th.

The report continues to gain global recognition as governments, organizations and civil society increasingly use happiness indicators to inform their policy-making decisions.

Leading experts across fields – economics, psychology, survey analysis, national statistics, health, public policy and more – describe how measurements of well-being can be used effectively to assess the progress of nations.

The reports review the state of happiness in the world today and show how the new science of happiness explains personal and national variations in happiness

```
> df<-read.csv(file.choose())
```

```
> colnames(df)
[1] "Overall.rank"                  "Country.or.region"
[3] "Score"                         "GDP.per.capita"
[5] "Social.support"                "Healthy.life.expectancy"
[7] "Freedom.to.make.life.choices" "Generosity"
[9] "Perceptions.of.corruption"
```

```
>View(df)
```

This command is used to view the entire dataset

```
> summary(df$Score)
   Min. 1st Qu.  Median    Mean 3rd Qu.
                                  Max. 2.853 4.545
  5.380 5.407 6.184  7.769
```

```
> summary(df$GDP.per.capita)
   Min. 1st Qu.  Median Mean 3rd Qu.  Max.
 0.0      0.6028 0.9600 0.9051 1.2325
 1.6840

> summary(df$Social.support)
   Min. 1st Qu.  Median Mean 3rd Qu.
                                Max. 0.000 1.056
  1.272 1.209 1.452  1.624
> summary(df$Healthy.life.expectancy)
   Min. 1st Qu.  Median Mean 3rd Qu.  Max.
 0.0      0.5477 0.7890 0.7252 0.8818
 1.1410

> summary(df$Perceptions.of.corruption)
   Min. 1st Qu.  Median Mean 3rd Qu.  Max.
 0.0      0.0470 0.0855 0.1106 0.1412
 0.4530
```

Here columns 1 and 2 are being removed from the dataframe and this is stored in dfn as they are not required in finding the correlation coefficient.

```
> dfn<-df[-c(1,2) ]
> dfn
```

Correlation and Regression

Correlation coefficient ranges from **-1 to 1**. It gives us an indication on two things:

1. The direction of the relationship between the 2 variables

   (**Positive correlation coefficient** shows that two variables under consideration vary in the **same direction**, i.e., if a variable increases the other one increases and if one decreases the other one decreases as well.

   A **negative correlation coefficient** implies that the two variables under consideration vary in **opposite directions**, that is, if a variable increases the other decreases and vice versa.)

2. The strength of the relationship between the 2 variables

The **more extreme** the correlation coefficient (the closer to -1 or 1), the **stronger the relationship**.

A **correlation close to 0** indicates that the two variables are **independent**, that is, as one variable increases, there is no tendency in the other variable to either decrease or increase.

Both Pearson and Spearman are used for measuring the correlation but the difference between them lies in the kind of analysis we want.

Pearson correlation: Pearson correlation evaluates the linear relationship between two continuous variables.

Spearman correlation: Spearman correlation evaluates the monotonic relationship. The Spearman correlation coefficient is based on the ranked values for each variable rather than the raw data.

Here we are using Pearson correlation

```
> coef<-cor(dfn,method="pearson")
> coef
```

	Score	GDP.per.capita	Social.support
Score	1.00000000	0.79388287	0.77705779
GDP.per.capita	0.79388287	1.00000000	0.75490573
Social.support	0.77705779	0.75490573	1.00000000
Healthy.life.expectancy	0.77988315	0.83546212	0.71900946
Freedom.to.make.life.choices	0.56674183	0.37907907	0.44733316
Generosity	0.07582369	-0.07966231	-0.04812645
Perceptions.of.corruption	0.38561307	0.29891985	0.18189946

	Healthy.life.expectancy	Freedom.to.make.life.choices
Score	0.77988315	0.5667418

GDP.per.capita	0.83546212	0.3790791
Social.support	0.71900946	0.4473332
Healthy.life.expectancy	1.00000000	0.3903948
Freedom.to.make.life.choices	0.39039478	1.0000000

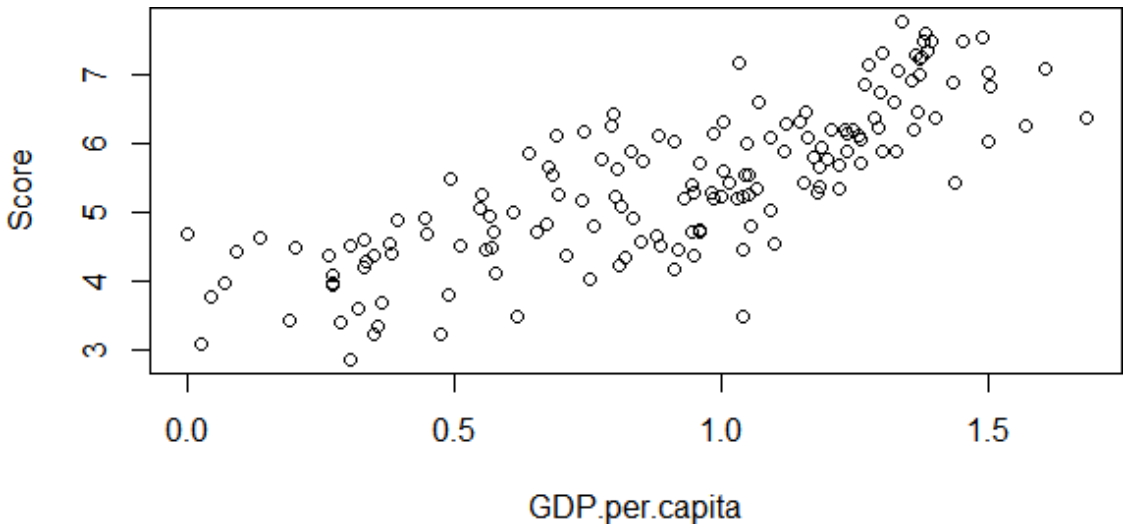
Generosity	-0.02951086	0.2697418
Perceptions.of.corruption	0.29528281	0.4388433

	Generosity	Perceptions.of.corruption
Score	0.07582369	0.3856131
GDP.per.capita	-0.07966231	0.2989198
Social.support	-0.04812645	0.1818995
Healthy.life.expectancy	-0.02951086	0.2952828
Freedom.to.make.life.choices		0.4388433
0.26974181		
Generosity	1.00000000	0.3265375
Perceptions.of.corruption	0.32653754	1.0000000

GDP.per.capita has highest correlation with Score(0.7938).Also Social.support(and Healthy Life Expectency has high positive correlation with Score.
Freedom.to.make.life.choices also has a high correlation with Score(0.56674183).Generosity doesn't affect the Score much(only 0.07582369 correlation coefficient).

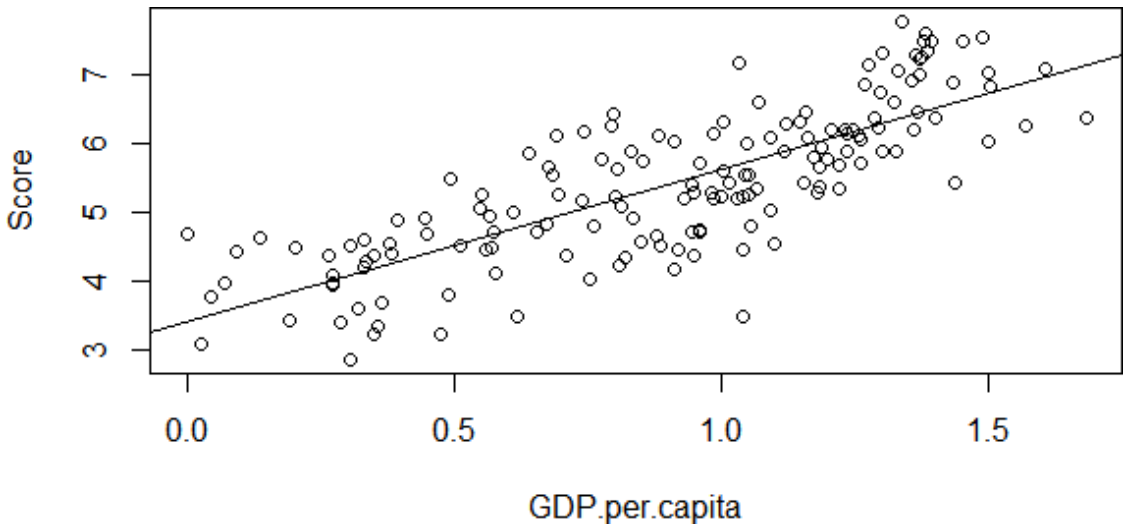
We fit regression models on these variables.

```
>x<-df[,c("Score")]
> y<-df[,c("GDP.per.capita")]
>coefSG<-cor(x,y,method="pearson")
> coefSG
[1] 0.7938829
>plot(y,x,xlab="GDP.per.capita",ylab="Score")
```

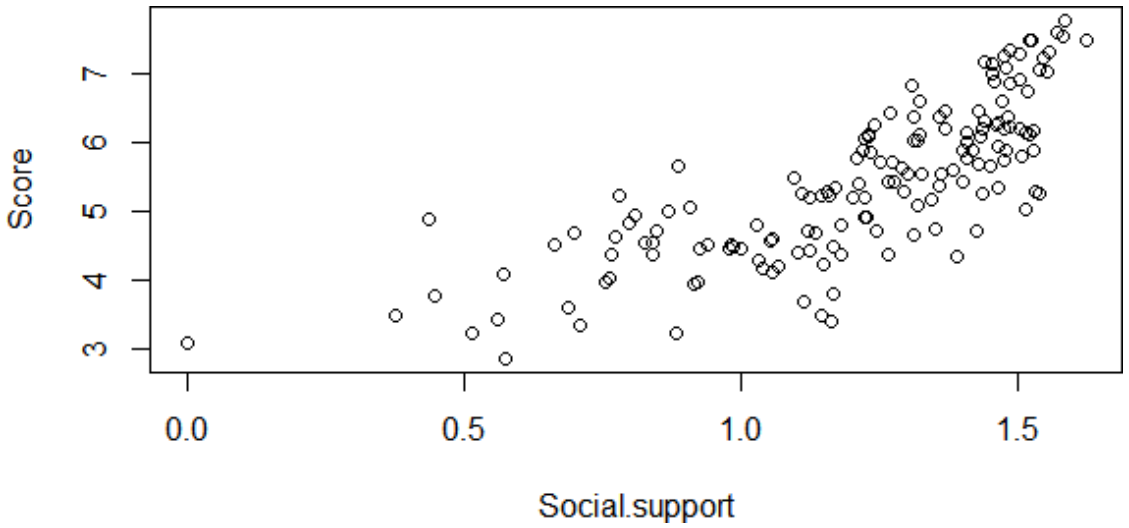


Scatter Plot showing how Score varies with GDP.per.capita

```
>modelSG=lm(Score~GDP.per.capita,data=data.frame(df))
> coefSG=coefficients(modelSG)
> coefSG
  (Intercept) GDP.per.capita
    3.399345     2.218148.per.capita,data=data.frame(df))
> abline(lm(Score~GDP.per.capita,data=data.frame(df)))
```



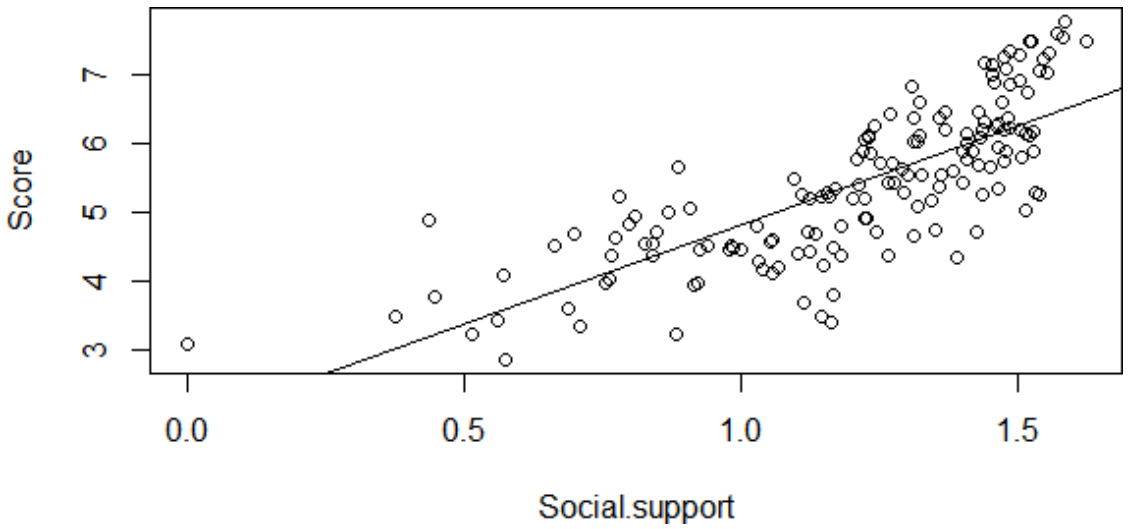

```
>y<-df[,c("Social.support")]
plot(y,x,xlab="Social.support",ylab="Sco
re")
```



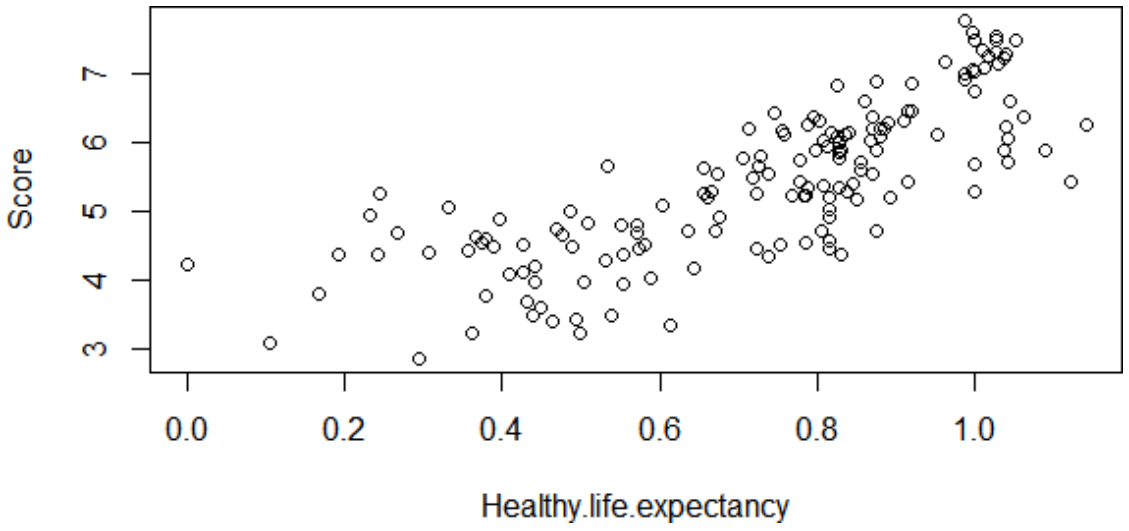

```
>modelSS=lm(Score~Social.support,data=data.frame(df))
>summary(modelSS)
```

```
Call:
lm(formula = Score ~ Social.support, data = data.frame(df))

Residuals:
    Min      1Q  Median      3Q     Max
-1.89465 -0.45762 -0.01993 0.54720  1.70721

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    1.9124     0.2349    8.14 1.25e-13
***
Social.support   2.8910     0.1887   15.32  < 2e-16 ***
---
Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7029 on 154 degrees of
freedom Multiple R-squared: 0.6038, Adjusted R-
squared: 0.6012 F-statistic: 234.7 on 1 and 154 DF, p-
value: < 2.2e-16
```

```
>abline(lm(Score~Social.support,data = data.frame(df)))
```




```
> y<-df[,c("Healthy.life.expectancy")]
> plot(y,x,xlab="Healthy.life.expectancy",ylab="Score")
```

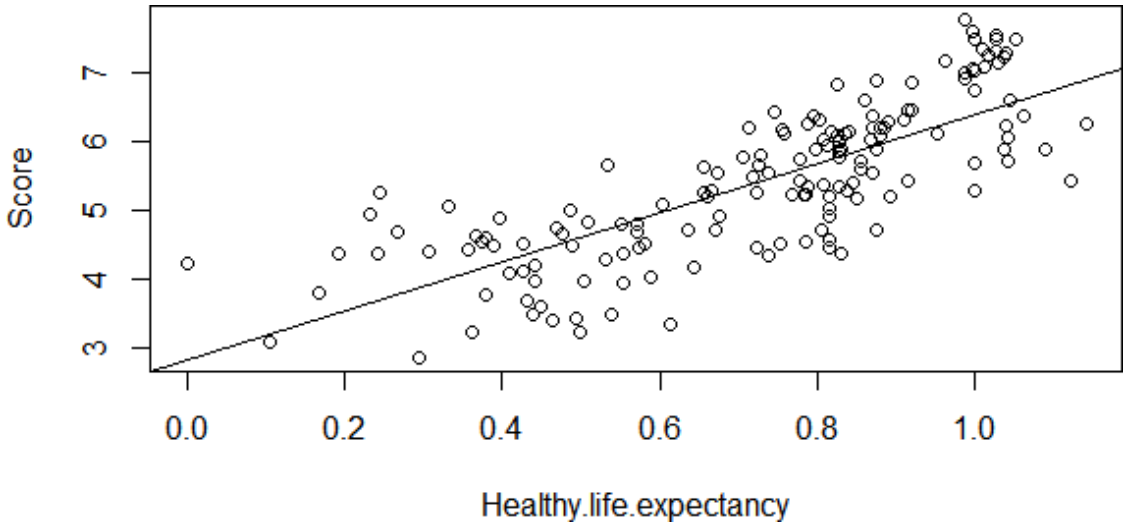



```
> modelSH=lm(Score~Healthy.life.expectancy,data=data.frame(df))
> coefsh<-coefficients(modelSH)
> coefsh
```

(Intercept) Healthy.life.expectancy

2.806832 3.585367

> abline(lm(Score~Healthy.life.expectancy,data=data.frame(df)))



All variables except Generosity and Trust appear to have a linear relationship with the dependent variable (Score). GDP, Social Support and Health Life are strongly correlated.

Multiple Linear Regression Model

It is used for modelling the relationship between a dependent variable and multiple independent variables.

> model1<-
lm(Score~GDP.per.capita+Social.support+Healthy.life.expectancy,data=
data.fr ame(df))

> coeffs1<-coefficients(model1)

> coeffs1

(Intercept) GDP.per.capita

Social.support 2.1350470

0.8098204 1.3218863

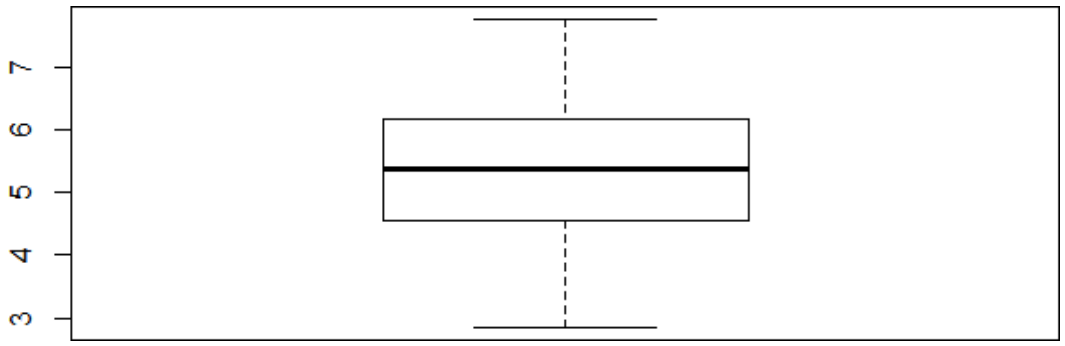
Healthy.life.expectancy

1.2976710

> summary(model1)

```
Call:
lm(formula = Score ~ GDP.per.capita + Social.support +
    Healthy.life.expectancy, data = data.frame(df))

Residuals:
    Min      1Q  Median    3Q    Max
-1.7018 -0.4155 -0.0520 0.4535 1.3369

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)           2.1350    0.2116 10.088 < 2e-16 ***
GDP.per.capita        0.8098    0.2358 3.434 0.000766 ***
Social.support        1.3219    0.2483 5.324 3.58e-07 ***
Healthy.life.expectancy   1.2977 0.3661 3.544 0.000523 ***
---
Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.588 on 152 degrees of
freedom Multiple R-squared: 0.7263, Adjusted R-
squared: 0.7209 F-statistic: 134.5 on 3 and 152 DF,
p-value: < 2.2e-16
```

```
> boxplot(df$Score)
```



From the box plot we can see the mean of Happiness Score is around 5.5

<u>We take null hypothesis that mean happiness score is 5.5 and perform one sample t test to check if the null hypothesis is true or not.</u>

The One Sample *t* Test has been used to find Statistical difference between a mean and a known or hypothesized value of the mean in the population

The null hypothesis (H_o) and (two-tailed) alternative hypothesis (H_1) of the one sample *T* test can be expressed as:

H_o: $\mu = \mu_o$ ("the population mean is equal to the [proposed] population mean")
H_1: $\mu \neq \mu_o$ ("the population mean is not equal to the [proposed] population mean")

where μ is the "true" population mean and μ_o is the proposed value of the population mean.

Here we are performing 2 sided t test in R. By default the t test in R is 2 sided.

```
> t.test(df$Score,mu=5.5,conf.level = 0.95)

        One Sample t-test

data:  df$Score
t = -1.0424, df = 155, p-value = 0.2988
alternative hypothesis: true mean is not equal to
5.5 95 percent confidence interval:
 5.231048 5.583144
sample
estimates: mean
of x 5.407096
```

The null hypothesis is false. The mean happiness score is not equal to 5.5