World Happiness Report Analysis

Archit

10/07/2021

INTRODUCTION

The World Happiness Report is a landmark survey of the state of global happiness, 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.

data source - from Kaggle

Task- To analyse and Visualise the World Happiness Report of year 2015-17.

Importing data sets

```
#importing data
whr_2015 <- read.csv("D:\\world happiness report\\2015_whr.csv")
whr_2016 <- read.csv("D:\\world happiness report\\2016_whr.csv")
whr_2017 <- read.csv("D:\\world happiness report\\2017_whr.csv")
```

Including all thr required packages

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

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

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

```
library(ggplot2)
library(corrplot)
```

```
## corrplot 0.90 loaded
```

```
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

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

```
## The following object is masked from 'package:graphics':
##
##     layout
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v tibble  3.1.2     v purrr   0.3.4
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

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

```
library(wildcard)
library(PerformanceAnalytics)
```

```
## Loading required package: xts
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
##
## Attaching package: 'xts'

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

##
## Attaching package: 'PerformanceAnalytics'

## The following object is masked from 'package:graphics':
##
##     legend
```

```
library(DT)
```

adding "year" column to all imported datasets

```
whr_2015$year <- 2015
whr_2016$year <- 2016
whr_2017$year <- 2017
```

The analysis process has been classified into 3 parts :

1) Region wise analysis
2) Country wise analysis
3) Combined of all years analysis

We will start with Region wise analysis-

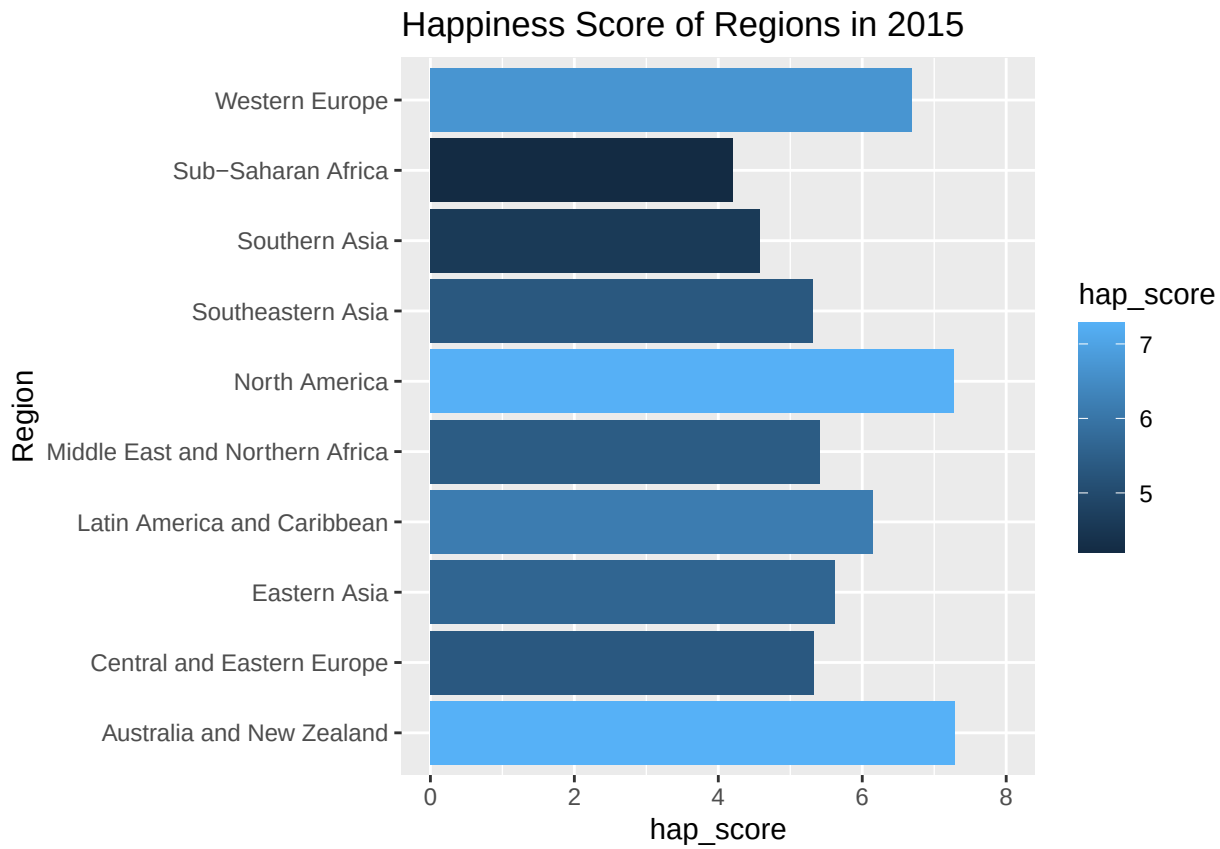
processing data into usefull format

```
subset_2015 <- whr_2015 %>%
  select(Country,Region)
add_region_2017 <- merge(whr_2017,subset_2015, by= "Country")


whr_2016$Standard.Error = ((whr_2016$Upper.Confidence.Interval-whr_2016$Happiness.Score)+(whr_2016$Happ
whr_clean1_2016 <- whr_2016 %>% select(- Lower.Confidence.Interval, -Upper.Confidence.Interval)

add_region_2017$Standard.Error = ((add_region_2017$Whisker.high - add_region_2017$Happiness.Score)+(add_
whr_clean1_2017 <- add_region_2017 %>% select(- Whisker.high, -Whisker.low)

# now we have required dataset

region_2015 <- whr_2015 %>%
                group_by(Region) %>%
                  summarize(hap_score = mean(Happiness.Score), econ= mean(Economy..GDP.per.Capita.),
                            freedom = mean(Freedom), health = mean(Health..Life.Expectancy.),
```

```
                          gov_trust = mean(Trust..Government.Corruption.),family= mean(Family))
region_2015$year=2015

region_2016 <- whr_2016 %>%
  group_by(Region) %>%
  summarize(hap_score = mean(Happiness.Score), econ= mean(Economy..GDP.per.Capita.),
            freedom = mean(Freedom), health = mean(Health..Life.Expectancy.),
            gov_trust = mean(Trust..Government.Corruption.),family= mean(Family))

region_2016$year= 2016

region_2017 <- add_region_2017%>%
  group_by(Region) %>%
  summarize(hap_score = mean(Happiness.Score), econ= mean(Economy..GDP.per.Capita.),
            freedom = mean(Freedom), health = mean(Health..Life.Expectancy.),
            gov_trust = mean(Trust..Government.Corruption.),family= mean(Family))
region_2017$year= 2017
```

now our data is ready to be visualised and analysed

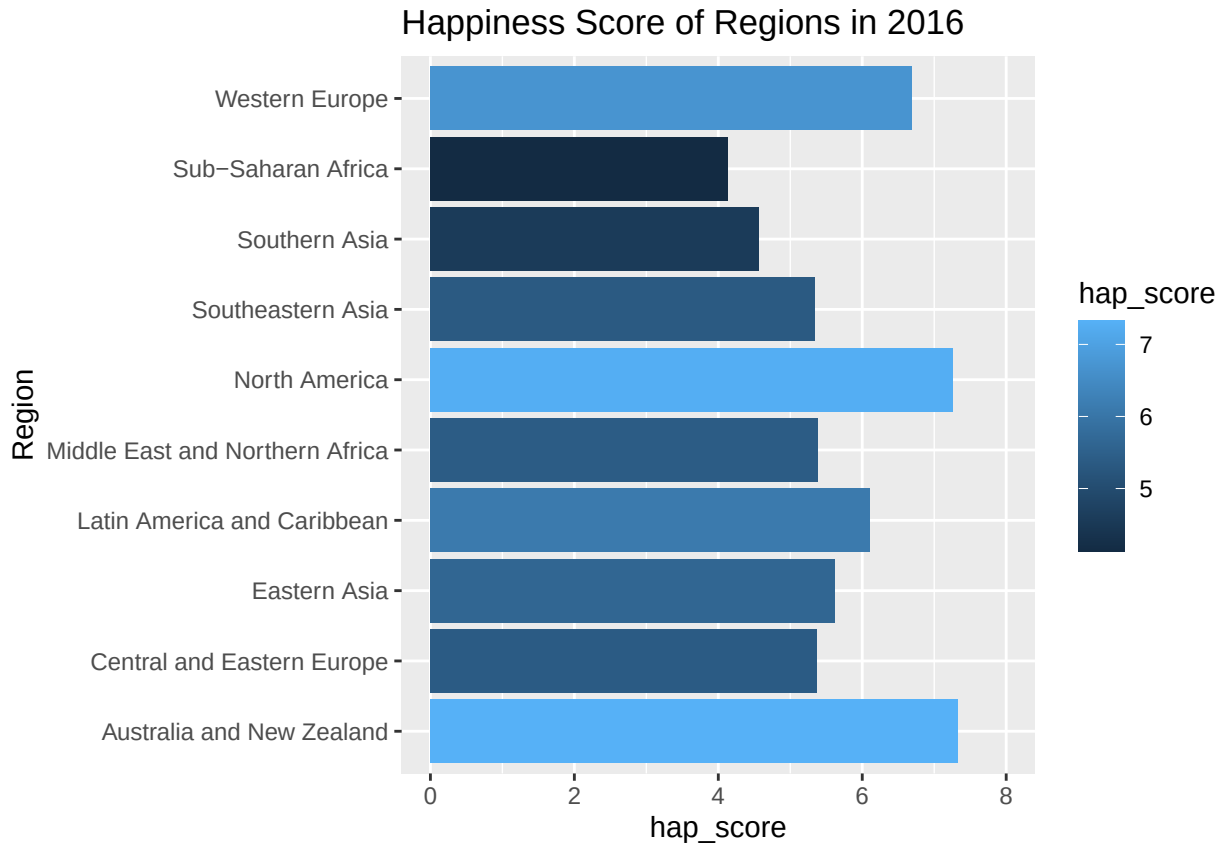
```
ggplot(data = region_2015, aes(x= hap_score, y= Region, fill= hap_score))+
  geom_bar(stat="identity")+
  xlim(0,8)+
  ggtitle("Happiness Score of Regions in 2015")
```



In 2015, the Sub-Saharan Africa region had the lowest happiness score and North America, Australia and

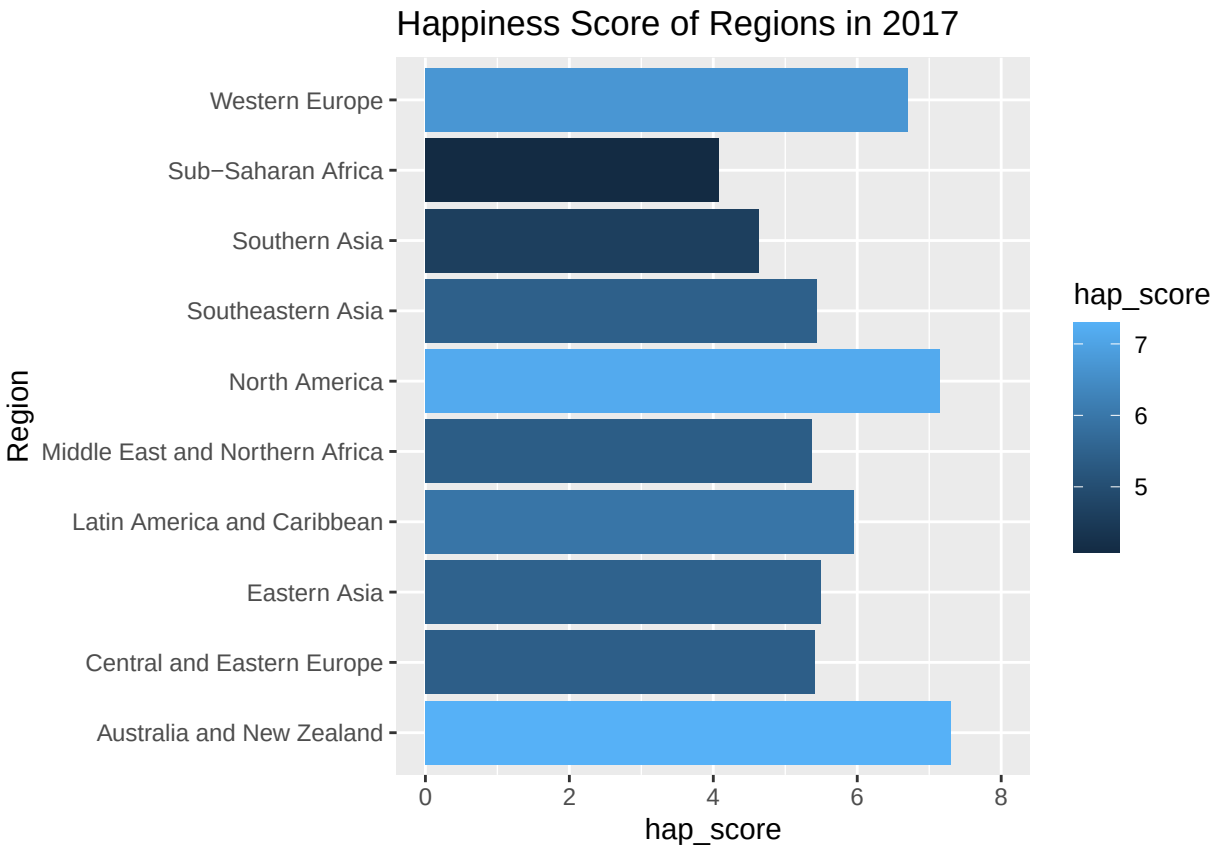
New Zealand regions having the highest.

```
ggplot(data = region_2016, aes(x= hap_score, y= Region,fill= hap_score ))+
  geom_bar(stat="identity")+
  xlim(0,8)+
  ggtitle("Happiness Score of Regions in 2016")
```



2016 had the same pattern with the Sub-Saharan Africa region with the lowest happiness score and North America, Australia and New Zealand regions having the highest.

```
ggplot(data = region_2017, aes(x= hap_score, y= Region,fill= hap_score ))+
  geom_bar(stat="identity")+
  xlim(0,8)+
  ggtitle("Happiness Score of Regions in 2017")
```



2017 too had similar observation but here the Australia New-Zealand significantly over passes North America.

Now, the Country wise analysis-

```
# we notice that 2017 doesnt have regions listed to we use 2015 dataset to add regions column to 2017
subset_2015 <- whr_2015 %>%
  select(Country,Region)
add_region_2017 <- merge(whr_2017,subset_2015, by= "Country")


whr_2016$Standard.Error = ((whr_2016$Upper.Confidence.Interval-whr_2016$Happiness.Score)+(whr_2016$Happ
whr_clean1_2016 <- whr_2016 %>% select(- Lower.Confidence.Interval, -Upper.Confidence.Interval)

add_region_2017$Standard.Error = ((add_region_2017$Whisker.high - add_region_2017$Happiness.Score)+(add_
whr_clean1_2017 <- add_region_2017 %>% select(- Whisker.high, -Whisker.low)
# now we have required dataset


wh15_17 <- bind_rows(whr_2015,whr_clean1_2016,whr_clean1_2017)
wh15_17 <- wh15_17 %>% select(Country:year, -Standard.Error)
colnames(wh15_17)[3] ="Happiness_Rank"
colnames(wh15_17)[4] ="Happiness_Score"
colnames(wh15_17)[5] ="Economy_GDP"
colnames(wh15_17)[7] ="Health"
```

```
colnames(wh15_17)[9] ="Gov_Trust"
colnames(wh15_17)[11] ="Dystopia_Residue"
# to check which countries didnt appear in all 3 years
countries.didnt.appear.3years <- wh15_17 %>% group_by(Country) %>% mutate(count = sum(year))

countries.didnt.appear.3years %>% filter(count != 6048) %>% select(Country, Happiness_Rank, year) %>% a
```

```
## # A tibble: 26 x 3
## # Groups:   Country [18]
##    Country                   Happiness_Rank  year
##    <chr>                              <int> <dbl>
##  1 Belize                                52  2016
##  2 Central African Republic             148  2015
##  3 Central African Republic             155  2017
##  4 Comoros                              140  2015
##  5 Comoros                              138  2016
##  6 Djibouti                             126  2015
##  7 Hong Kong                             72  2015
##  8 Hong Kong                             75  2016
##  9 Laos                                  99  2015
## 10 Laos                                 102  2016
## # ... with 16 more rows
```

to check all rows that have null values in them and get insights to data frame.

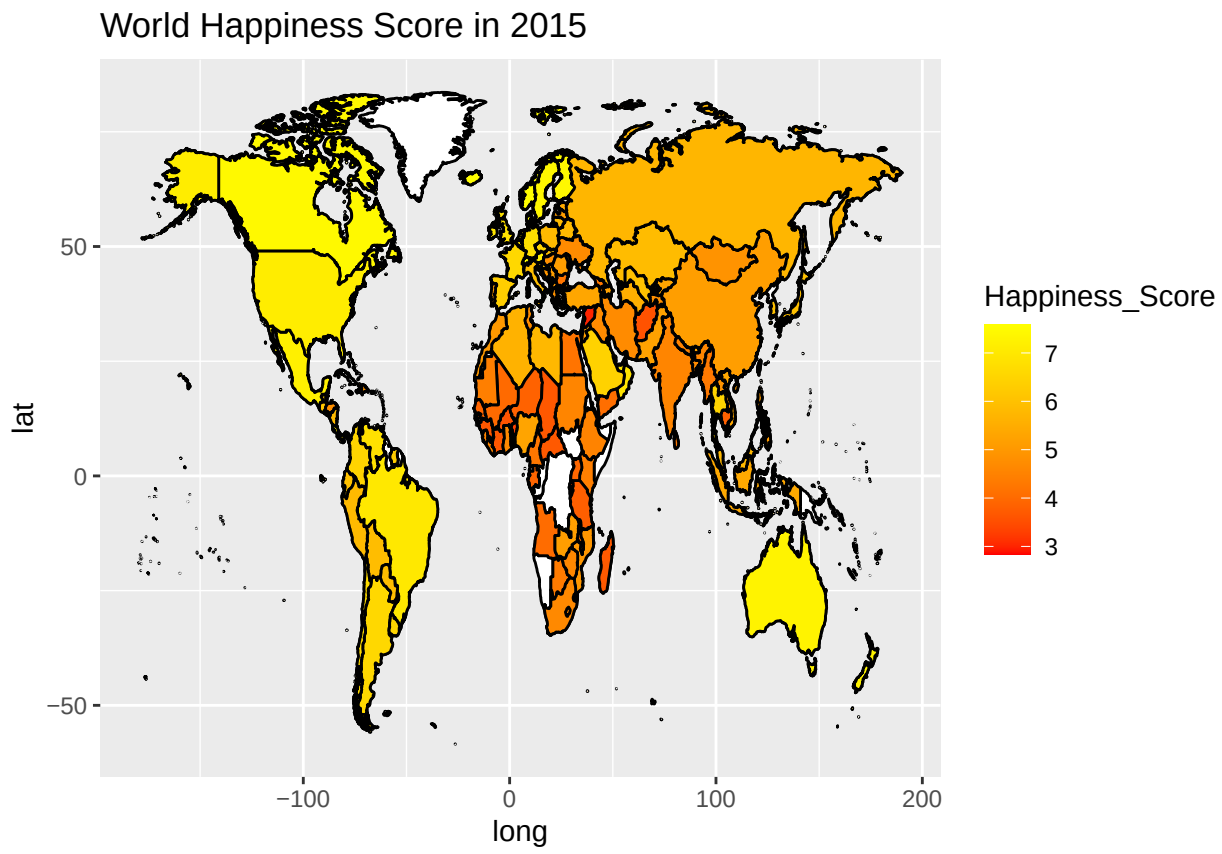
```
summary(wh15_17) # no NA values
```

```
##    Country             Region          Happiness_Rank   Happiness_Score
##  Length:464         Length:464         Min.   :  1.00   Min.   :2.693
##  Class :character   Class :character   1st Qu.: 39.75   1st Qu.:4.508
##  Mode  :character   Mode  :character   Median : 79.00   Median :5.282
##                                        Mean   : 78.76   Mean   :5.373
##                                        3rd Qu.:118.00   3rd Qu.:6.247
##                                        Max.   :158.00   Max.   :7.587
##   Economy_GDP         Family           Health          Freedom
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.6098   1st Qu.:0.7935   1st Qu.:0.4042   1st Qu.:0.2965
##  Median :0.9958   Median :1.0234   Median :0.6306   Median :0.4176
##  Mean   :0.9285   Mean   :0.9899   Mean   :0.5814   Mean   :0.4023
##  3rd Qu.:1.2518   3rd Qu.:1.2282   3rd Qu.:0.7671   3rd Qu.:0.5151
##  Max.   :1.8708   Max.   :1.6106   Max.   :1.0252   Max.   :0.6697
##    Gov_Trust         Generosity     Dystopia_Residue      year
##  Min.   :0.00000   Min.   :0.0000   Min.   :0.3286   Min.   :2015
##  1st Qu.:0.05925   1st Qu.:0.1526   1st Qu.:1.7380   1st Qu.:2015
##  Median :0.09950   Median :0.2225   Median :2.0946   Median :2016
##  Mean   :0.13449   Mean   :0.2421   Mean   :2.0944   Mean   :2016
##  3rd Qu.:0.17272   3rd Qu.:0.3161   3rd Qu.:2.4543   3rd Qu.:2017
##  Max.   :0.55191   Max.   :0.8381   Max.   :3.8377   Max.   :2017
```

```
world <- map_data('world')
world <- world %>% filter(region != "Antarctica")
world <- fortify(world)
```

```r
library(wildcard)

# we need to change name to USA as in tha world dataset which will help us in plotting on map the name
happiness.score15 <- wh15_17 %>% select(Country, Happiness_Score, year) %>% filter(year == 2015)
happiness.score15 <- wildcard(df = happiness.score15, wildcard = "United States", values = "USA",

                              expand = TRUE, rules = NULL)

happiness.score15 <- wildcard(df = happiness.score15, wildcard = "United Kingdom", values = "UK",

                              expand = TRUE, rules = NULL)



happiness.score15 <- wildcard(df = happiness.score15, wildcard = "Democratic Republic of the Congo", va

                              expand = TRUE, rules = NULL)
# now finally plotting on world map

ggplot(data=world)+
  geom_map(map=world, aes( x=long, y= lat, group= group, map_id= region), fill="white", colour="black")+
  geom_map(data=happiness.score15, map=world,

           aes(fill=Happiness_Score, map_id=Country),

           colour="black") +

  scale_fill_continuous(low="red", high="yellow",

                        guide="colorbar") +

  labs(title = "World Happiness Score in 2015")
```

```
## Warning: Ignoring unknown aesthetics: x, y
```

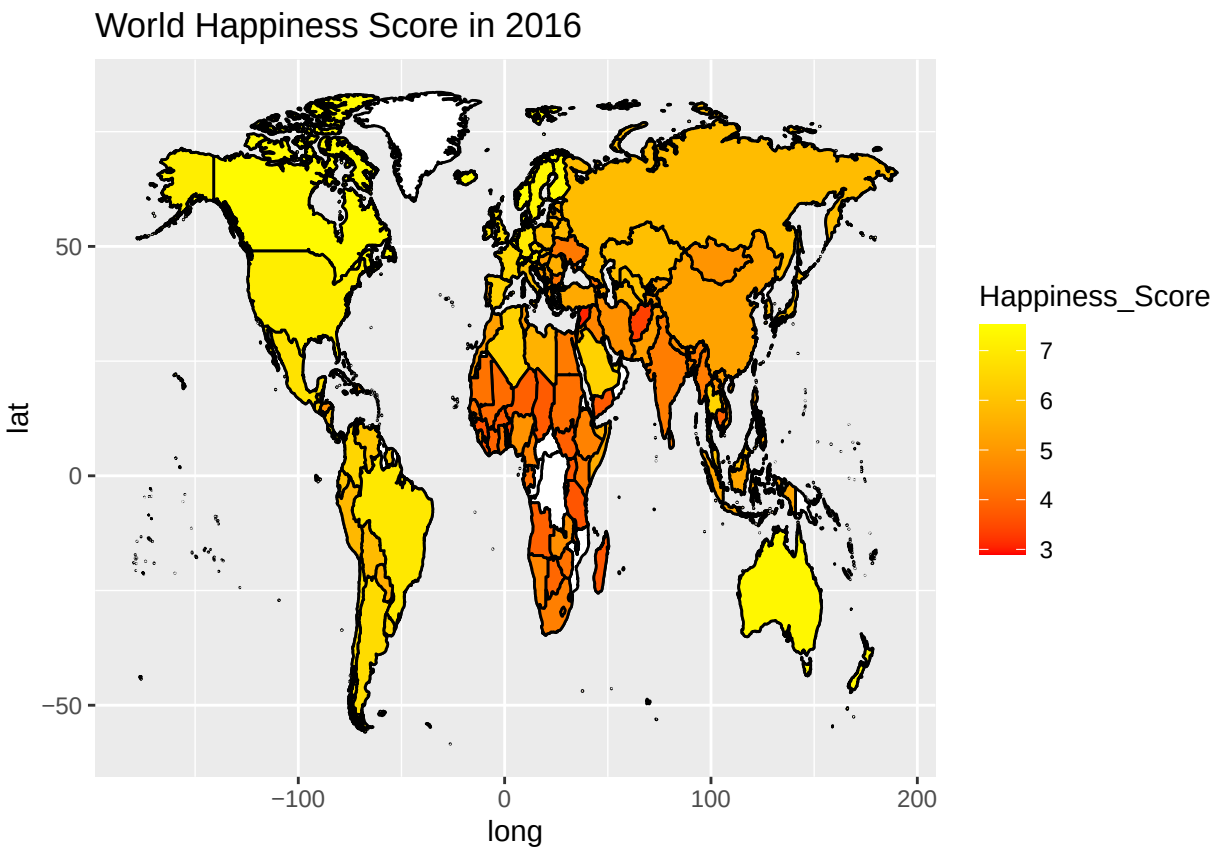


Above is a heatmap which marks the happiness score of all countries on a world map in 2015

```
happiness.score16 <- wh15_17 %>% select(Country, Happiness_Score, year) %>% filter(year == 2016)
happiness.score16 <- wildcard(df = happiness.score16, wildcard = "United States", values = "USA",

                              expand = TRUE, rules = NULL)

happiness.score16 <- wildcard(df = happiness.score16, wildcard = "United Kingdom", values = "UK",

                              expand = TRUE, rules = NULL)



happiness.score16 <- wildcard(df = happiness.score16, wildcard = "Democratic Republic of the Congo", val

                              expand = TRUE, rules = NULL)
# now finally plotting on world map

ggplot(data=world)+
  geom_map(map=world, aes( x=long, y= lat, group= group, map_id= region), fill="white", colour="black")+
  geom_map(data=happiness.score16, map=world,

           aes(fill=Happiness_Score, map_id=Country),

           colour="black") +
```

```
  scale_fill_continuous(low="red", high="yellow",
                        guide="colorbar") +
  labs(title = "World Happiness Score in 2016")
```

```
## Warning: Ignoring unknown aesthetics: x, y
```

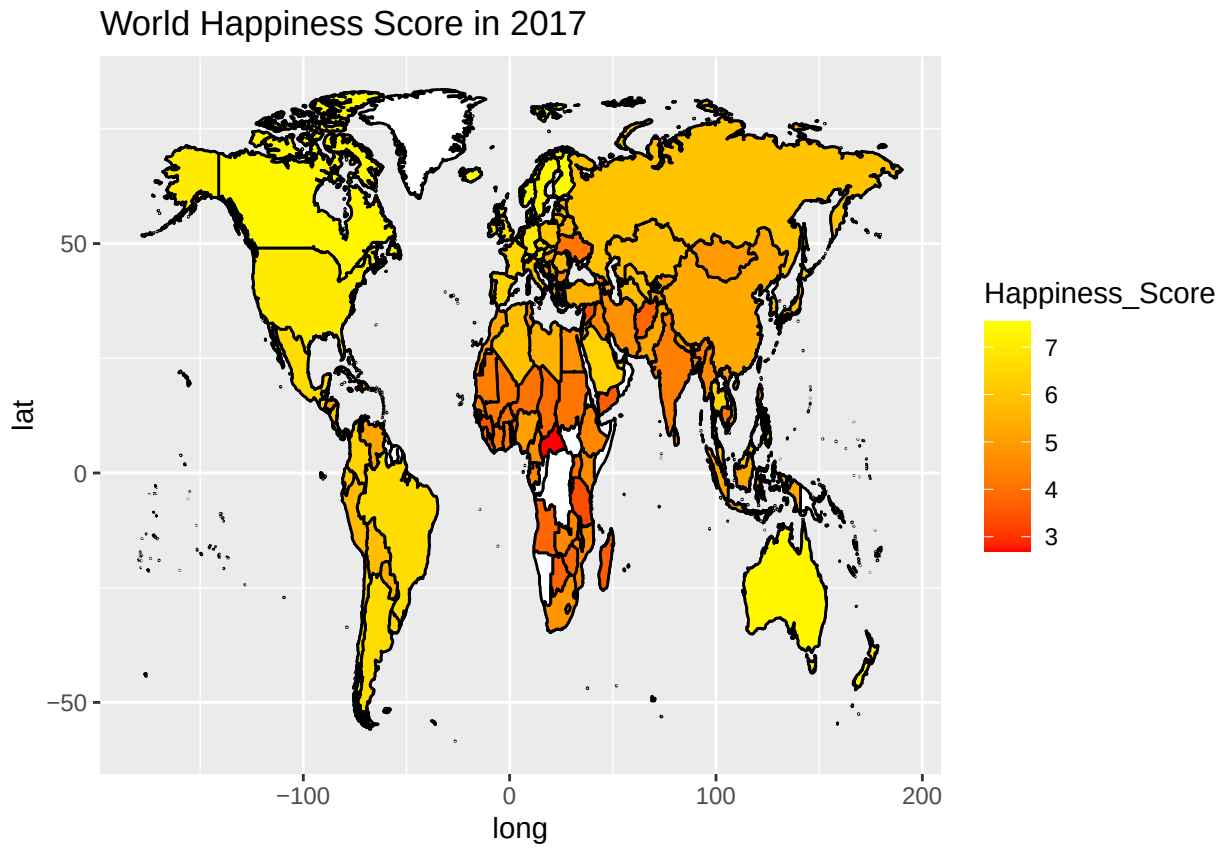


Above is a heatmap which marks the happiness score of all countries on a world map in 2016

```
happiness.score17 <- wh15_17 %>% select(Country, Happiness_Score, year) %>% filter(year == 2017)
happiness.score17 <- wildcard(df = happiness.score17, wildcard = "United States", values = "USA",
                              expand = TRUE, rules = NULL)
happiness.score17 <- wildcard(df = happiness.score17, wildcard = "United Kingdom", values = "UK",
                              expand = TRUE, rules = NULL)


happiness.score17 <- wildcard(df = happiness.score17, wildcard = "Democratic Republic of the Congo", va
                              expand = TRUE, rules = NULL)
# now finally plotting on world map
```

```
ggplot(data=world)+
  geom_map(map=world, aes( x=long, y= lat, group= group, map_id= region), fill="white", colour="black")+
  geom_map(data=happiness.score17, map=world,

           aes(fill=Happiness_Score, map_id=Country),

           colour="black") +

  scale_fill_continuous(low="red", high="yellow",

                        guide="colorbar") +

  labs(title = "World Happiness Score in 2017")
```

```
## Warning: Ignoring unknown aesthetics: x, y
```



Analysis of all 3 year Combined

```
combine_country <- wh15_17 %>%
                    group_by(Country) %>%
                    summarize( happiness_score = mean(Happiness_Score),
                               econ = mean(Economy_GDP),
```

```
                              fam= mean(Family),
                              health = mean(Health),
                              freedom = mean(Freedom),
                              gov_trust = mean(Gov_Trust))

ggplot(combine_country, aes(x=happiness_score, y= econ,fill="orange"))+
  geom_point(size= 1.5) +
  geom_smooth(method ="loess")+
  labs(title="Correlation Between Economy Growth And Happiness Score", x="happiness score", y="economy")
    theme_minimal()
```

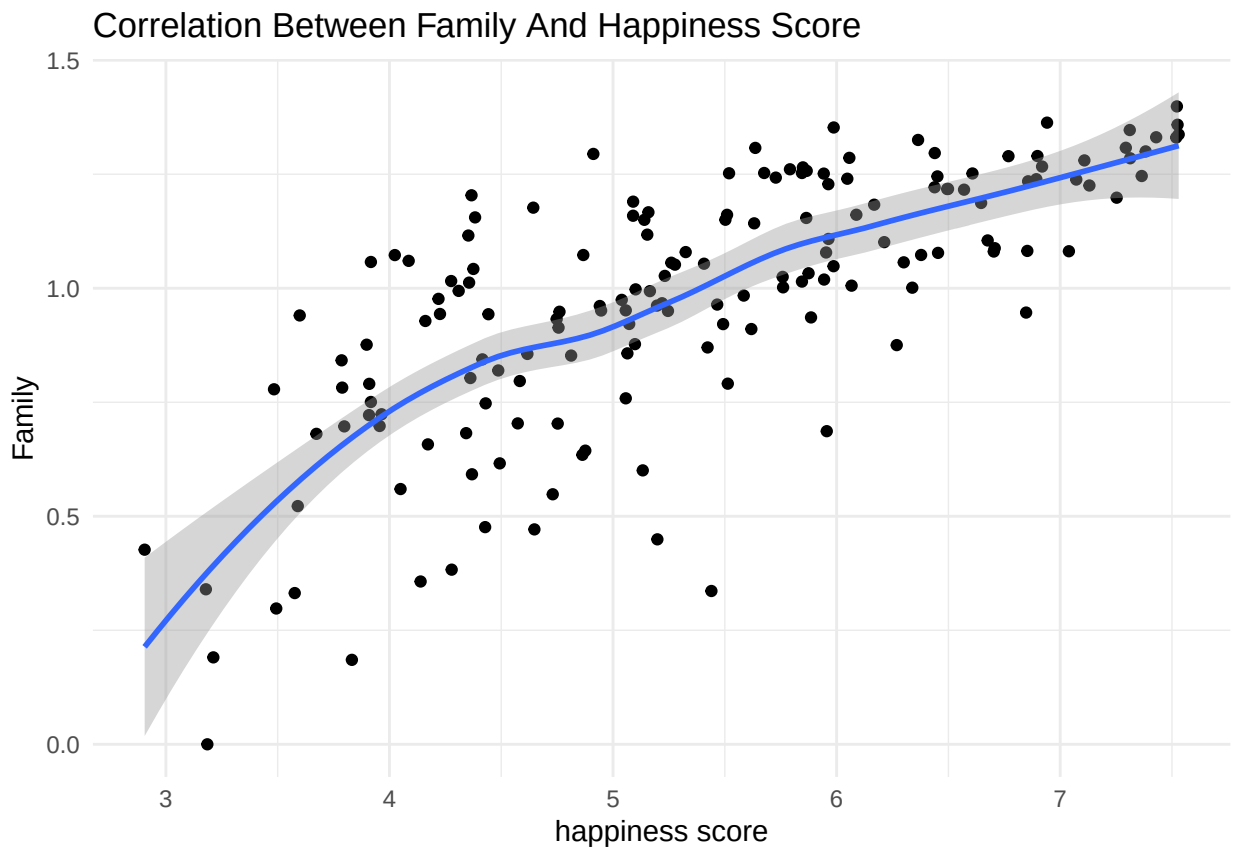
```
## `geom_smooth()` using formula 'y ~ x'
```


```
cor(combine_country$happiness_score, combine_country$econ)
```

```
## [1] 0.7943887
```

```
ggplot(combine_country, aes(x=happiness_score, y= fam))+
  geom_point(size= 1.5) +
  geom_smooth(method ="loess")+
  labs(title="Correlation Between Family And Happiness Score", x="happiness score", y="Family")+
  theme_minimal()
```

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

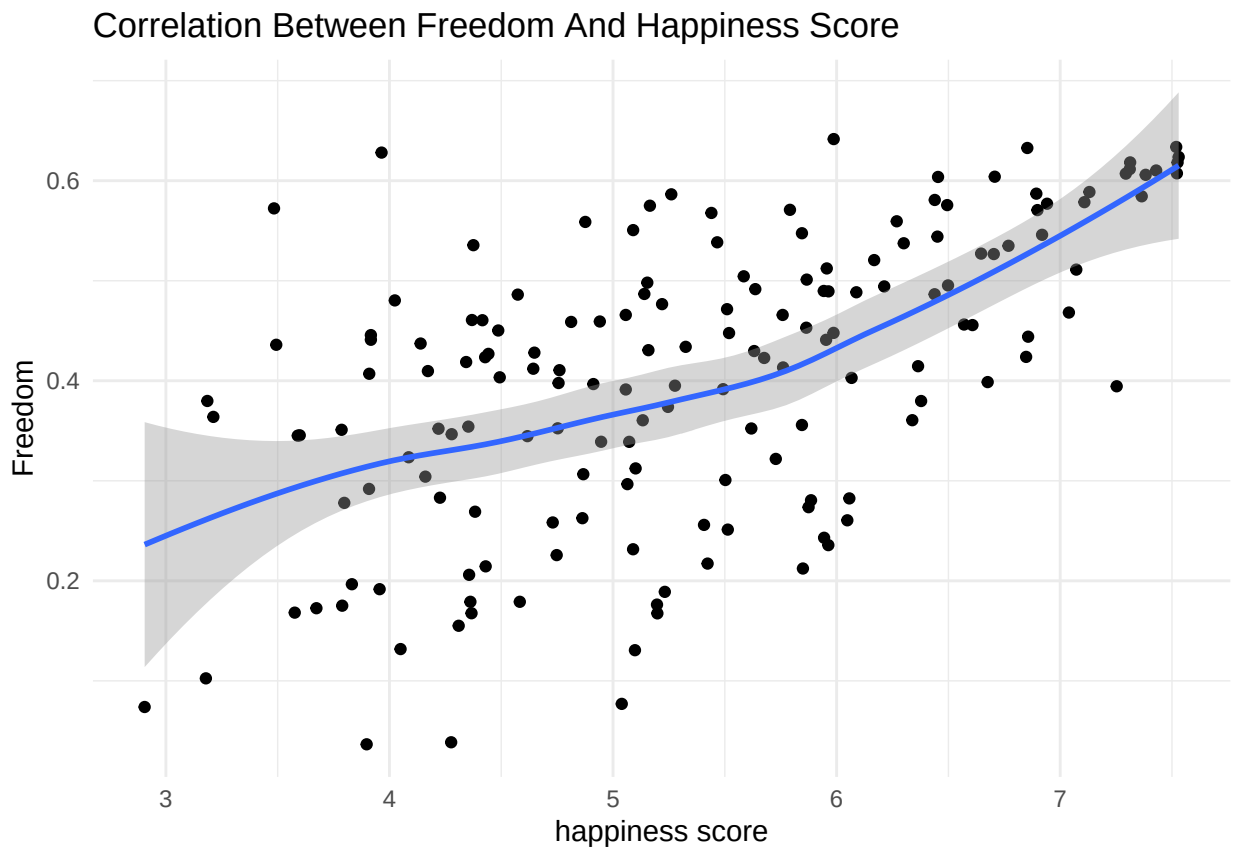



```
cor(combine_country$happiness_score, combine_country$fam)
```

```
## [1] 0.7280598
```

```
ggplot(combine_country, aes(x=happiness_score, y= freedom))+
  geom_point(size= 1.5) +
  geom_smooth(method ="loess")+
  labs(title="Correlation Between Freedom And Happiness Score", x="happiness score", y="Freedom")+
  theme_minimal()
```

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

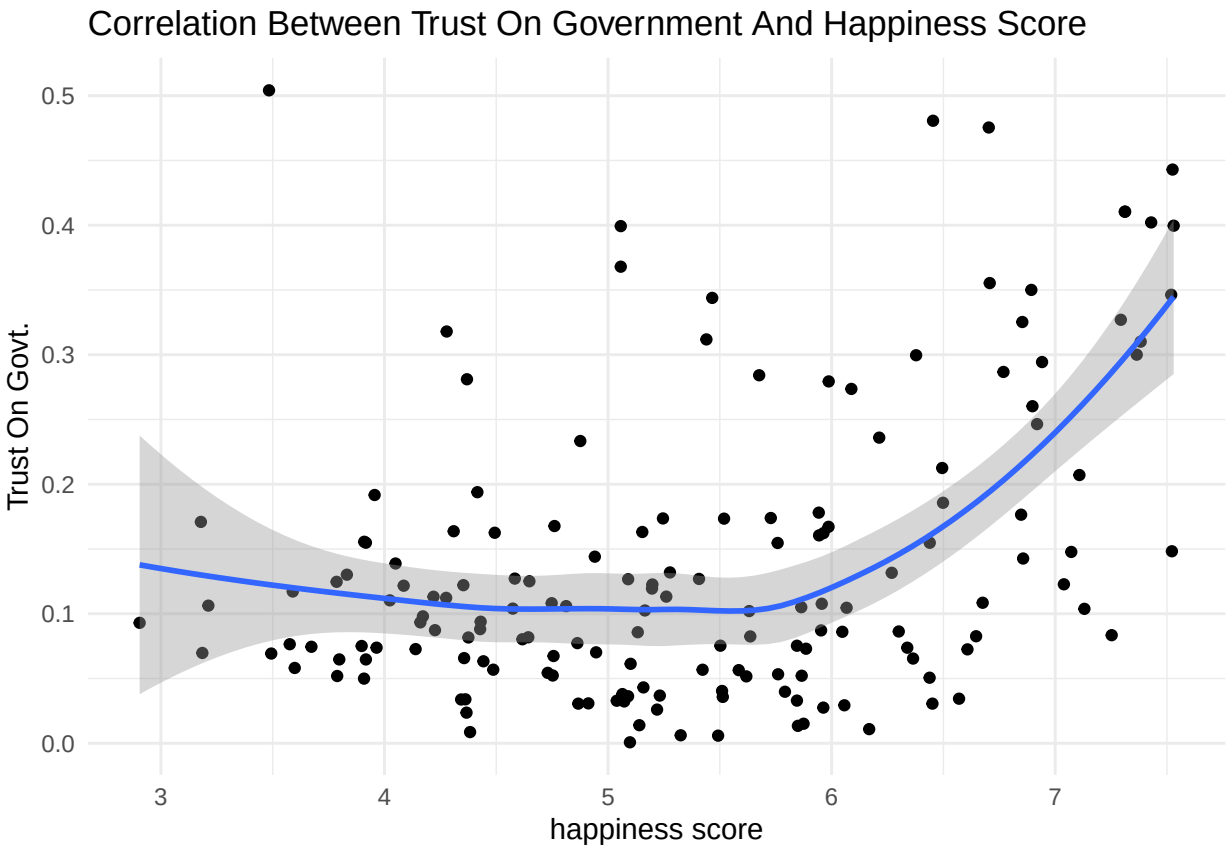



```
cor(combine_country$happiness_score, combine_country$freedom)
```

```
## [1] 0.5797765
```

```
ggplot(combine_country, aes(x=happiness_score, y= gov_trust))+
  geom_point(size= 1.5) +
  geom_smooth(method ="loess")+
  labs(title="Correlation Between Trust On Government And Happiness Score", x="happiness score", y="Trus
  theme_minimal()
```

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

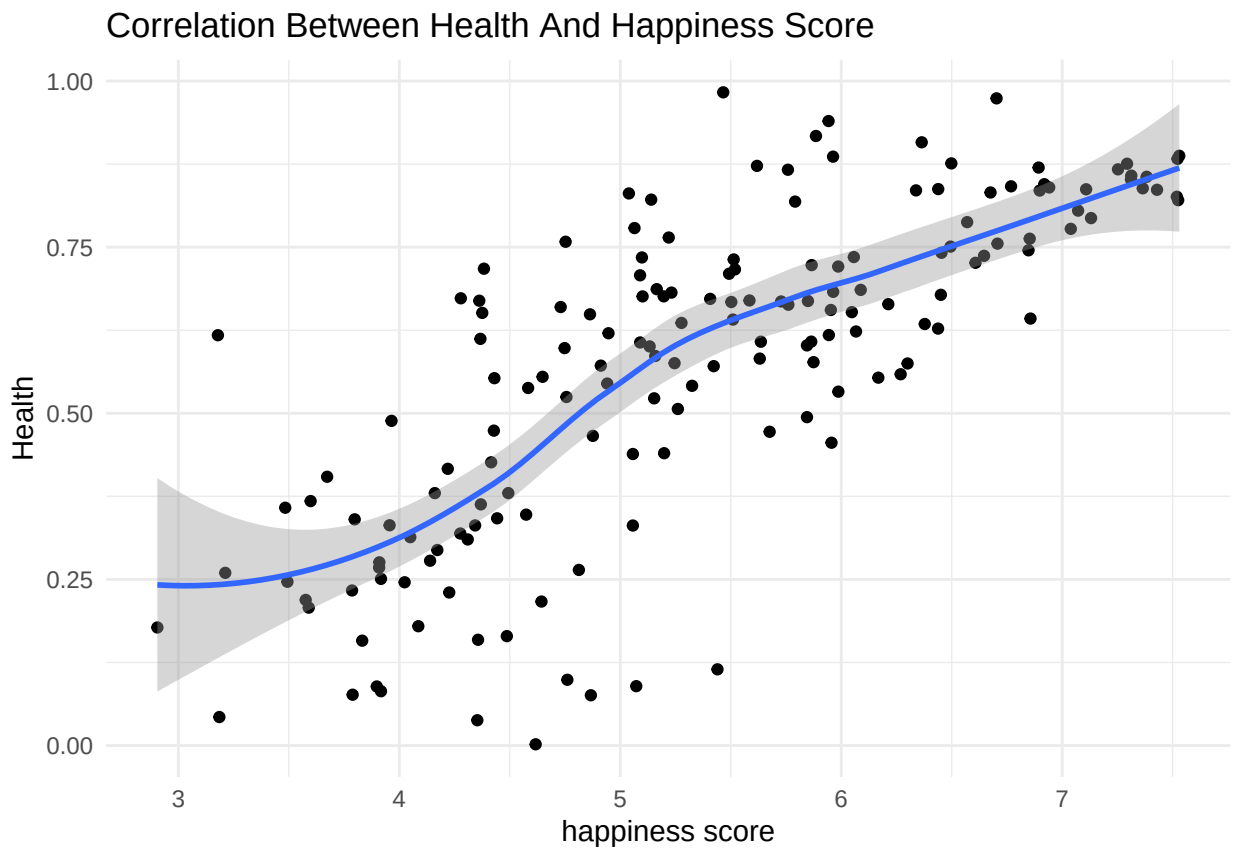



```
cor(combine_country$happiness_score, combine_country$gov_trust)
```

```
## [1] 0.3890825
```

```
ggplot(combine_country, aes(x=happiness_score, y= health))+
  geom_point(size= 1.5) +
  geom_smooth(method ="loess")+
  labs(title="Correlation Between Health And Happiness Score", x="happiness score", y="Health")+
  theme_minimal()
```

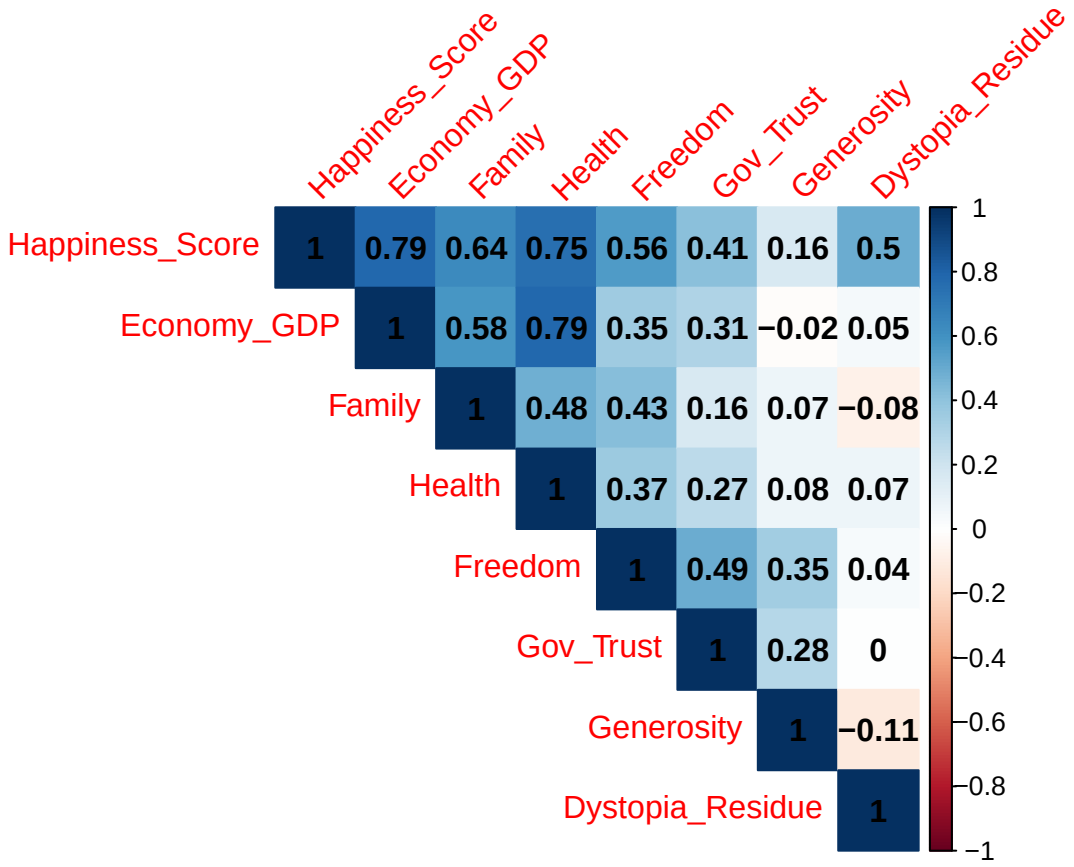
```
## `geom_smooth()` using formula 'y ~ x'
```



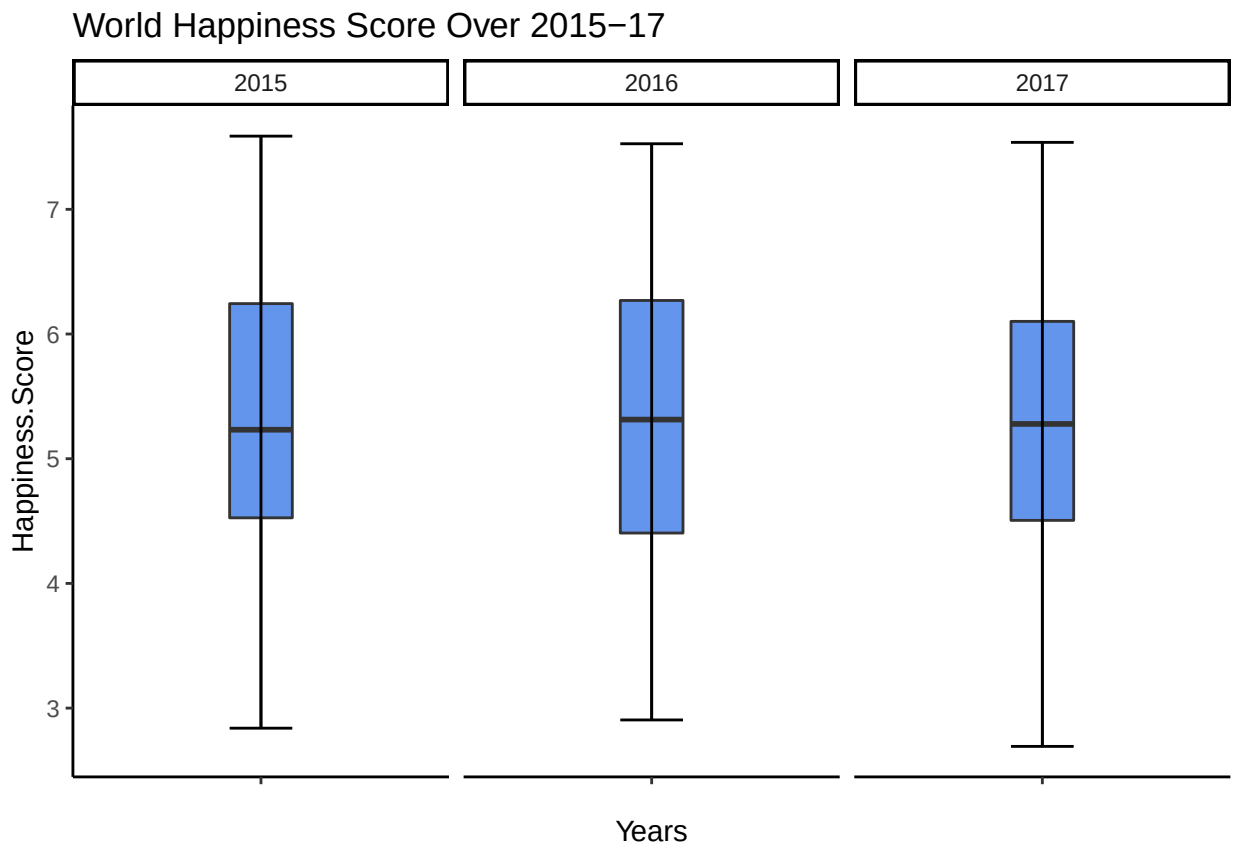

```
cor(combine_country$happiness_score, combine_country$health)
```

```
## [1] 0.7607928
```

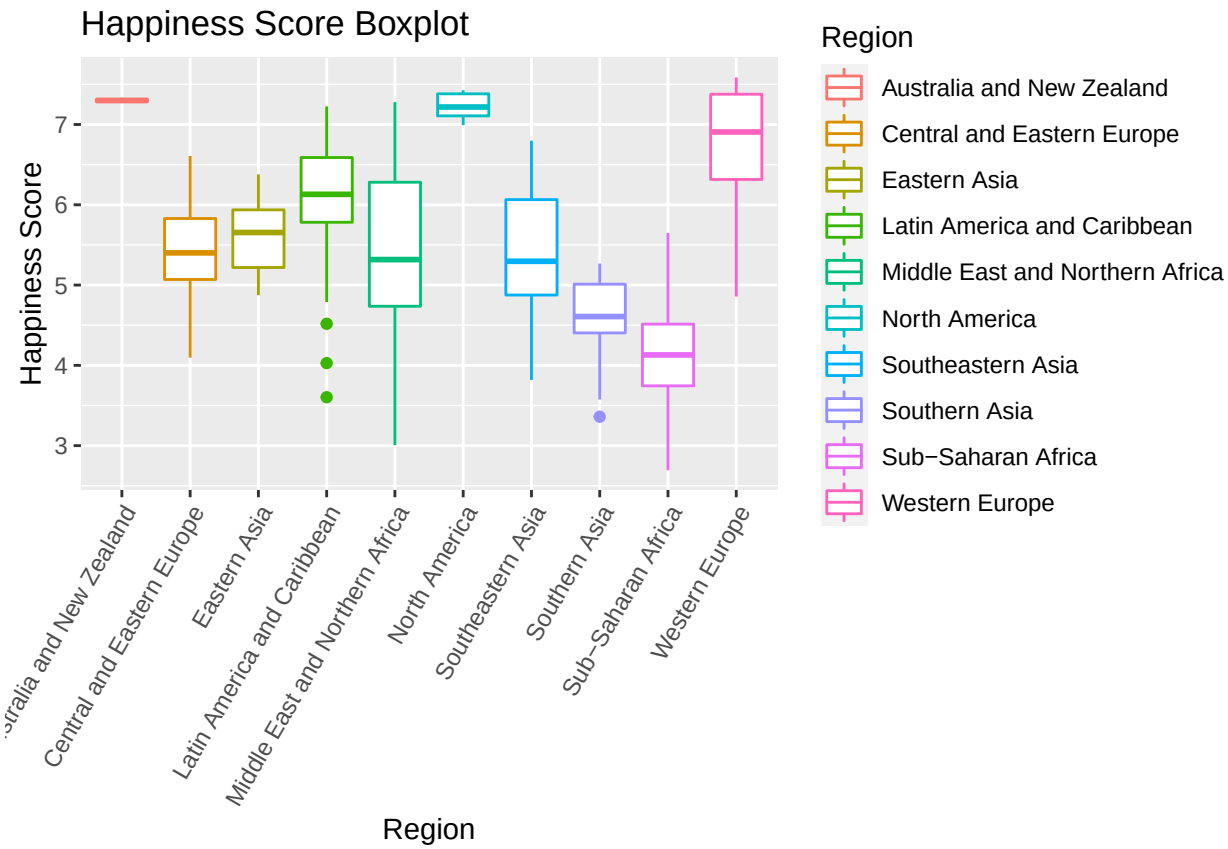
```
corrplot(cor(wh15_17 %>%

               select(Happiness_Score,Economy_GDP,
                      Family,Health,Freedom, Gov_Trust,
                      Generosity, Dystopia_Residue)),

         method="color",

         sig.level = 0.01, insig = "blank",

         addCoef.col = "black",

         tl.srt=45,

         type="upper"

)
```




```
happiness_score_15<- select(whr_2015,Country,Happiness.Score,year)
happiness_score_16<- select(whr_2016,Country,Happiness.Score,year)
happiness_score_17<- select(whr_2017,Country,Happiness.Score,year)
happiness_score_df <-bind_rows(happiness_score_15,happiness_score_16,happiness_score_17)

med= median(happiness_score_df$Happiness.Score)
ggplot(happiness_score_df, aes(x="", y= Happiness.Score, fill=""))+
  geom_boxplot(width= 0.2, fill= "Cornflowerblue")+
  facet_wrap(~year)+ theme_classic()+
  labs(title="World Happiness Score Over 2015-17", x="Years")+
  stat_boxplot(geom="errorbar", width= 0.2)
```




```
# boxplot to show happiness score of combines 3 years
ggplot(wh15_17, aes(x=Region, y= Happiness_Score, colour = Region)) +

  geom_boxplot() +

  theme(axis.text.x = element_text(angle = 60, hjust = 1)) +

  labs(title = "Happiness Score Boxplot",

       x = "Region",

       y = "Happiness Score")
```



The Above boxplot displays the mean happiness score of all regions in all 3 years combined.