WORLD HAPPINESS ANALYSIS

Overview: The project aims at measuring the happiness index of the world by performing data analysis. The data set being used is published by the United Nations and was collected by conducting life evaluation questions asked in a poll. It includes data from across the world and we are going to analyse multiple factors that might influence happiness amongst people.

The data set considered,currently has 5 years worth of data from the year 2015 to 2019. Data cleaning,wrangling,structuring,enriching and manipulation will be done on all the data sets based on requirement for forming meaningful insights.

Installation of the required packages for analysis:

```
#install.packages("ggcorrplot")
#install.packages( Rtools ) #install.packages( plotly
)
#install.packages("heatmaply")
#install.packages("ggcorrplot")
#install.packages( fuzzyjoin )
#install.packages( hrbrthemes )
#install.packages( zoo )
#install.packages( ggplotly )
#library( ggplot2 ) library( tidyverse
)
```

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

## v ggplot2 3.3.5 v purrr 0.3.4
## v tibble 3.1.4 v dplyr 1.0.7 ##
v tidyr 1.1.3 v stringr 1.4.0 ##
v readr 2.0.1 v forcats 0.5.1

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

```
library( dplyr )
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( ggcorrplot )
library( stringr )
library( fuzzyjoin )
#ggplotly( corr.plot
) library( plotly )
library( heatmaply )
```

```
##
## ======================
## Welcome to heatmaply version
1.3.0 ##
## Type citation('heatmaply') for how to cite the package.
## Type ?heatmaply for the main documentation.
##
## The github page is: https://github.com/talgalili/heatmaply/
## Please submit your suggestions and bug-reports at:
https://github.com/talgalili/heatmaply/issues ## You may ask questions at
stackoverflow,
use the r and heatmaply tags:
##https://stackoverflow.com/questions/tagged/heatmaply
## ======================
```

```
library( hrbrthemes )
```

```
## NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these
themes.

## Please use hrbrthemes::import_roboto_condensed() to install Roboto Condensed and

## if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow
```

```
library(reshape2)
```

```
##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
## smiths
#hrbrthemes::import_roboto_condensed() library(
zoo )
```

```
##
## Attaching package: 'zoo' ## The following objects are

masked from 'package:base':

##
## as.Date, as.Date.numeric
# The required packages are installed along with their corresponding libraries.
```

The Data sets present in the csv files are read to obtain the data required for Analysis

```
getwd()
```

```
## [1] "/Users//Downloads"
```

```
setwd( /Users//Documents/Foundations of Data Analytics /Project 1 /Data ) getwd()
```

```
## [1] "/Users//Documents/Foundations of Data Analytics /Project 1 /Data"
happiness_2015 <- read.csv( 2015.csv ) #
```

```
2015happiness_2016 <- read.csv( 2016.csv ) # 2019 is read using the read.csv
happiness_2017 <- read.csv( 2017.csv ) #
function happiness_2018 <- read.csv(
2018.csv ) happiness_2019 <- read.csv(
2019.csv ) country_data <- read.csv( Country.csv )
```

Data Enriching and Manipulation:

Data enriching involves adding value to the data already collected to enhance it for the analysis to be done at hand.In this case,we add the column year to give the data more context.

```
happiness_2015$Year <- # Here we are adding a column "year" for each of the
2015 happiness_2016$Year <- # 5 data sets.Each of these data sets contains
2016   happiness_2017$Year <- # the happiness data for that year,but the year
       is
2017   happiness_2018$Year <- # not mentioned in them.
2018   happiness_2019$Year
<- 2019
```

World Happiness Data for the years:

Data Manipulation and cleaning is done to ensure that there is consistency in the data.For each of the years mentioned below the data required is filtered,and manipulated to bring it a standardized form. 2015- `health = heal ,"Happiness`

Country	Region	Happiness Rank	Happiness Score	Economy or GDP	Health	Freedom	Perception Of Government Corruption	Family or Social Support	Year
Switzerland	Western Europe	1	7.587	1.39651	0.94143	0.66557	0.41978	1.34951	2015
Iceland	Western Europe	2	7.561	1.30232	0.94784	0.62877	0.14145	1.40223	2015
Denmark	Western Europe	3	7.527	1.32548	0.87464	0.64938	0.48357	1.36058	2015
Norway	Western Europe	4	7.522	1.45900	0.88521	0.66973	0.36503	1.33095	2015
Canada	North America	5	7.427	1.32629	0.90563	0.63297	0.32957	1.32261	2015

```
df_2015_filtered <- happiness_2015 %>% select
  (Country , Region, Happiness.Rank,
  Happiness.Score,
  Economy..GDP.per.Capita.,Health..Life.Expectancy.,
         Freedom, Trust..Government.Corruption., Family, Year )

# The required data is selected from the data set of the world happiness #
in the year 2015.The factors that could possibly affect the
# happiness score in different parts of the world are considered and placed
in #the df_2015_filtered variable.

df_2015_filtered <- rename_with(df_2015_filtered,
                            ~ tolower(gsub(".", "_", .x, fixed = TRUE)))

#  The  columns  considered  are  edited  to  fit  a  standardized  form.

df_2015_filtered  <-  df_2015_filtered  %>%  rename(economy_or_gdp  =

economy__gdp_per_capita_,                      #head(df_2015_filtered,5)

knitr::kable(head(df_2015_filtered,5),"pipe",col.names

=c("Country","Region","Happiness Rank"

                                                              TRUE)))
```

```
# The columns are renamed to give appropriate names which are common to all the
# years having the same column names,each corresponding to the relevant data #
pertaining to that year.
```

2016-

```
# happiness score in different parts of the world are considered and placed
in #the df_2016_filtered variable.


df_2016_filtered <- rename_with(df_2016_filtered, ~ tolower(gsub(".", "_",
.x, fixed = # The columns considered are edited to fit a standardized form.
df_2016_filtered <- df_2016_filtered %>% rename(economy_or_gdp =
economy__gdp_per_capita_, perception_of_govt_corruption=
trust__government_corruption_, family_or_social_support = family
)
#head(df_2016_filtered,5)
knitr::kable(head(df_2016_filtered,5),"pipe",col.names
=c("Country","Region","Happiness Rank"

df_2016_filtered <- happiness_2016 %>% select
  (Country , Region, Happiness.Rank,
  Happiness.Score,
  Economy..GDP.per.Capita.,Health..Life.Expectancy.,
         Freedom, Trust..Government.Corruption.,Family, Year )

# The required data is selected form the data set of the world happiness # in
the year 2016.The factors that could possibly affect the
```

Country	Region	Happiness Rank	Happiness Score	Economy or GDP	Health	Freedom	Perception Of Government Corruption	Family or Social Support	Year
Denmark	Western Europe	1	7.526	1.44178	0.79504	0.57941	0.44453	1.16374	2016
Switzerland	Western Europe	2	7.509	1.52733	0.86303	0.58557	0.41203	1.14524	2016
Iceland	Western Europe	3	7.501	1.42666	0.86733	0.56624	0.14975	1.18326	2016
Norway	Western Europe	4	7.498	1.57744	0.79579	0.59609	0.35776	1.12690	2016
Finland	Western Europe	5	7.413	1.40598	0.81091	0.57104	0.41004	1.13464	2016

```
# years having the same column names,each corresponding to the relevant data #
pertaining to that year.
```

2017-

```
df_2017_filtered <- happiness_2017 %>% select
  (Country , Happiness.Rank, Happiness.Score,
         Economy..GDP.per.Capita.,Health..Life.Expectancy.,
         Freedom, Trust..Government.Corruption.,Family, Year)

# The required data is selected form the data set of the world happiness
# in the year 2017.The factors that could possibly affect the #
happiness score in different part of the world are considered and placed
in #the df_2017_filtered variable

#region missing in the 2017 table df_2017_filtered <-
rename_with(df_2017_filtered, ~ tolower(gsub(".", "_",  .x,
fixed =
```

```
TRUE))) health =
```

```
healt
```

```
df_2015_filtered , country, region),
```

```
for few ye
```

```
##                     country             happiness_rank
##  0       0 ## happiness_score economy_or_gdp ##
0 0 ## health freedom
##  0  0 ## perception_of_govt_corruption  family_or_social_support
## 0 0 ## year region
##                           0                          0
```

```
#validating and checking for null values

#head(df_2017_filtered,5) knitr::kable(head(df_2017_filtered,5),"pipe",col.name
=c("Country","Region","Happiness Rank"
```

```
,"Happines
```

```
# The columns considered are edited to fit a standardized form.


df_2017_filtered <- df_2017_filtered %>% rename(economy_or_gdp =
economy__gdp_per_capita_, perception_of_govt_corruption=
trust__government_corruption_, family_or_social_support = family )

# The columns are renamed to give appropriate names which are common to
all the # years having the same column names,each corresponding to the
relevant data # pertaining to that year.


#df_2017_filtered  <-  df_2017_filtered  %>%  left_join(select(df_2016_filtered

region), by= "co # The region column is not present in the 2017 world data ind

that needs to be added. # df_2017_filtered

df_2017_filtered <- df_2017_filtered %>% regex_inner_join(select(
df_2017_filtered <- select(df_2017_filtered, -country.y) %>% rename(country=
country.x)

# Here we are adding the region for the 2017 data Using regex inner join instea
join to help w
# For example the Country Cyprus has the name as Cyprus for a few years and
North Cyprus colSums(is.na(df_2017_filtered))
```

Country	Region	Happiness Rank	Happiness Score	Economy or GDP	Health	Freedom	Perception Of Government Corruption	Family or Social Support	Year
Norway	Western Europe	1	7.537	1.616463	0.796665	0.635422	0.315963	1.533524	2017

Country	Region	HappinessRank	HappinessScore	Economy or GDP	Health	Freedom	Perception Of Government Corruption	Family or Social Support	Year

Denmark	2	7.522	1.482383	0.79256550.62600607.4007701	1.551122	2017	Western Europe
Iceland	3	7.504	1.480633	0.83355210.62716206.1535266	1.610574	2017	Western Europe
Switzerland	4	7.494	1.564980	0.85813130.6200706.3670073	1.516912	2017	Western Europe
Finland	5	7.469	1.443572	0.80915770.6179509.3826115	1.540247	2017	Western Europe

2018-

```
df_2018_filtered <- happiness_2018 %>% select (Country.or.region
  , Overall.rank,
  Score,
     GDP.per.capita,Healthy.life.expectancy, Freedom.to.make.life.choices
          , Perceptions.of.corruption, Social.support, Year)

# The required data is selected form the data set of the world happiness
# in the year 2018.The factors that could possibly affect the
# happiness score in different parts of the world are considered and placed in
#the df_2018_filtered variable df_2018_filtered <-

rename_with(df_2018_filtered, ~ tolower(gsub(".", "_", .x, fixed = # The

columns considered are edited to fit a standardized form.

df_2018_filtered <- df_2018_filtered %>% rename(country =
 country_or_region,economy_or_gdp = health =
 healthy_life_expectancy,perception_of_govt_corruption=
 perceptions_of_corruption , )
# The columns are renamed to give appropriate names which are common to
all the # years having the same column names,each corresponding to the
relevant data # pertaining to that year.

#df_2018_filtered  <-  df_2018_filtered  %>%  left_join(select(df_2016_filtered

region), by= "co # The region column is not present in the 2017 world data ind

that needs to be added.

#adding region to 2018 data by joining on country

df_2018_filtered <- df_2018_filtered %>% regex_inner_join(select(
df_2018_filtered <- select(df_2018_filtered, -country.y) %>% rename(country=
country.x)

## Here we are adding the region for the 2018 data Using regex inner join inste
inner join to help


#head(df_2018_filtered,5) knitr::kable(head(df_2018_filtered,5),"pipe",col.name
=c("Country","Region","Happiness Rank"
```

ap
family_or_
so

df_2015_f
iltered ,
country,
region),

,"Happines
ppiness Hap

HealthFreedom Pe Family or Year Rank Score or GDP Government Social

Corruption Support

Finland	1	7.632	1.305	0.874	0.681	0.393	1.592	2018	Western Europe
Norway	2	7.594	1.456	0.861	0.686	0.340	1.582	2018	Western Europe
Denmark	3	7.555	1.351	0.868	0.683	0.408	1.590	2018	Western Europe
Iceland	4	7.495	1.343	0.914	0.677	0.138	1.644	2018	Western Europe
Switzerland	5	7.487	1.420	0.927	0.660	0.357	1.549	2018	Western Europe

```
df_2019_filtered <- happiness_2019 %>% select (Country.or.region
  , Overall.rank,
  Score,
     GDP.per.capita,Healthy.life.expectancy, Freedom.to.make.life.choices ,
          Perceptions.of.corruption, Social.support, Year)


# The required data is selected form the data set of the world
happinessknitr::kable(head(df_2019_filtered,5),"pipe",col.names    # in
the year 2019.The factors that could possibly affect
th=c("Country","Region","Happiness Rank"  e

 # happiness score in different parts of the world are considered and placed
in#the df_2019_filtered variable df_2019_filtered <-
```

rename_with(df_2019_filtered, ~ tolower(gsub(Country RegionHappiness Hap	piness Eco nomy							
HealthFreedom P"."erception Of , "_", .x, fixed = Family or# The								Year
	Rank	Score	or GDP		Government		Social	
columns considered are edited to fit a standardized form.Corruption							Support	
Finland 1 7.769 1.340 0.986 0.596 0.393 1.587 2019 Western df_2019_filtered <- df_2019_filtered								
%>% rename(country = Europe								
country_or_region,freedom_to_make_life_choices 7.600 economy_or_gdp =								
happiness_rank = 1.383 0.996 0.592 0.410 1.573overall_rank, 2019freedom								
= Western								
Denmark 2								
)								Europe
Norway 3 7.554 1.488 1.028 0.603 0.341 1.582 2019 Western								
# The columns are renamed to give appropriate names which are common								Europe
Iceland to all the # years having the same column names,each corresponding to								4
7.494 1.380 1.026 0.591 0.118 1.624 2019 Western the relevant data #								
pertaining to that year. Europe								
#df_2019_filtered <Netherland5s 7.488 - 1.396df_2019_filtered %>%								
left_join(select(df_2016_filtered 0.999 0.557 0.298 1.522 2019								
Western								
Europe								

```
region), by= "co # The region column is not present in the 2017 world data ind

that needs to be added.

#adding region to 2018 data by joining on country

df_2019_filtered <- df_2019_filtered %>% regex_inner_join(select(
df_2019_filtered <- select(df_2019_filtered, -country.y) %>% rename(country=
country.x)

## Here we are adding the region for the 2019 data Using regex inner join inste
inner join to help #df_2019_filtered
```

```
df_2015_filtered , country, region),
,"Happiness
```

Wrangling:

```
# The data type is converted into double in order to maintain uniformity.


df_15_16_17_18 <- union_all(df_15_16_17,  df_2018_filtered) df_final <-
union_all(df_15_16_17_18, df_2019_filtered)

#df_final

knitr::kable(head(df_final,5),"pipe",col.names
=c("Country","Region","Happiness Rank",
```

Data wrangling involves unifying messy raw data into a form which is simpler to access and handle.In this case all the di erent data sets of the years are combined to create a single master data set,containing all the data that is required for analysis.

```
#Union all has been identified as the best approach while combining data. If we
to combine the

df_15_16 <- union_all(df_2015_filtered, df_2016_filtered) df_15_16_17
<- union_all(df_15_16,df_2017_filtered)
df_2018_filtered$perception_of_govt_corruption <- as.double(df_2018_filtered$

                                                    perception_of_govt_corrupt


## Warning: NAs introduced by coercion
"Happiness Score"
```

Country	Region	Happiness Rank	Happiness Score	HealthFreedom or GDP	Perception Of Corruption	Economy	Family	Government Social Support	Year
SwitzerlanWdestern Europe		1	7.587	1.39651	0.94143	0.66557		0.41978	1.349512015
Iceland	Western Europe	2	7.561	1.30232	0.94784	0.62877		0.14145	1.402232015
DenmarkWestern Europe		3	7.527	1.32548	0.87464	0.64938		0.48357	1.360582015
Norway	Western Europe	4	7.522	1.45900	0.88521	0.66973		0.36503	1.330952015
Canada	North America	5	7.427	1.32629	0.90563	0.63297		0.32957	1.322612015

```
# The data frames for all the years are combined together using the
union all # function.As we can see the the df_15_16 variable contains
the union or
```

```
and # finally the union of all of them are stored in the df_final
variable.
```

Data cleaning also involves taking care of the null values by removing them so they do not a ect our analysis.Here we check if there are any NA values in the final data frame and also remove the temporary variable which are not required any more.

```
remove(df_15_16) remove(df_15_16_17) remove(df_15_16_17_18)

# the temporary variables created in order to perform the union function are #
later deleted as we have the final data frame which contains all the unions.
 #validating and checking for null vales  colSums(is.na(df_final))
```

```
##                     country                      region
##                           0                           0
##              happiness_rank           happiness_score
##                           0                           0
##              economy_or_gdp                      health
## 0        0 ## freedom perception_of_govt_corruption ##
0 1 ## family_or_social_support year
##                           0                           0
```

```
NA_df <- df_final[rowSums(is.na(df_final)) > 0,] head(NA_df,5)
```

```
##  country  region happiness_rank ## 489 United Arab Emirates Middle
East and Northern Africa 20
##happiness_score economy_or_gdp health freedom perception_of_govt_corruption
## 489         6.774         2.096 0.67 0.284                           NA
## family_or_social_support year
## 489                 0.776 2018
```

```
#checking for NA values in the df_final which contains all the unions.
```

Data Validation:

Data validation is checking if the data has undergone the cleansing it requires to be used.However in this case we find that a NA value in the UAE gives us an error ,hence we take the average value of other values on the same column instead of discarding it to prevent data loss.

```
#Finding the records with NA values

NA_df <- df_final[rowSums(is.na(df_final)) > 0,]


#There is an NA value for perception of govt_corruption for United Arab emirates

#replacing the missing NA value of perception of govt corruption with the average

percep govt cor df_final$perception_of_govt_corruption = as.numeric(as.factor(df_final$

#changing datatype for calculating the mean

# Due to the presence of an NA value for perception of govt_corruption for United Arab
error # In this case we have taken an average of the perception of govt corruption mea
UAE the valu
```

```
df_final$perception_of_govt_corruption[is.na(df_final$perception_of_govt_corruption)]<-

#validating the replaced values and checking for NAs colSums(is.na(df_final))

                                                                perception_of_govt_corruption))

                                                                                mean(df_UAE$per
```

```
##                     country                      region
##                           0                           0
##             happiness_rank           happiness_score
##                           0                           0
##             economy_or_gdp                      health
##                           0                           0
##                     freedom perception_of_govt_corruption

## 0 0 ## family_or_social_support year
##                           0                           0
NA_df <- df_final[rowSums(is.na(df_final)) > 0,]

NA_df
## [1] country                    region
## [3] happiness_rank             happiness_score
## [5] economy_or_gdp             health
## [7] freedom                    perception_of_govt_corruption
## [9] family_or_social_support year
## <0 rows> (or 0-length row.names)
```

Business Question 1: To find the happiest countries all over the world based on the happiness score

```
top_country_list <- df_final %>% group_by(country)
  %>%
  summarise(happiness_score= sum(happiness_score)) %>%
arrange(desc(happiness_score)) %>% slice(1:5) head(top_country_list,5)
```

```
## # A tibble: 5 x 2
## country happiness_score
##  <chr> <dbl> ## 1
```

```
ye
 37.7 ## 3 Norway 37.7
## 4 Finland             37.7
## 5 Switzerland         37.6
```

```
knitr::kable(top_country_list,"pipe",col.names =c("Country","Happiness Score"),align
=c("c","c"))
```

Country	Happiness Score
Nigeria	41.131
Denmark	37.730
Norway	37.705
Finland	37.689
Switzerland	37.557

```
# Based on the master data set created df_final we group the
countries based # on the happiness score and arrange them
in descending order to obtain the top 5 countries with
highest
```

Conclusion: When the data across all the years have been considered. The above table shows us that Nigeria is the country with the highest happiness score of 41.131,while Switzerland comes in at fifth position with a score of 37.557. This is an interesting finding as Nigeria doesn't make it to the top 5 list for any of the years even though it has the highest happiness score across all the years.

Business Question 2: To find out the countries that take up the top spot on the happiness score over the years

```
countries_rank_1 <- df_final %>% dplyr::filter(happiness_rank == 1) %>%
select(                                                                  country,
                                                                        year,
                                                                        happiness_
#head(countries_rank_1, 5,) knitr::kable(countries_rank_1,"pipe",col.names
=c("Country","Year","Happiness
Score"),                                                                      align
                                                                        =c("c","c",
```

Country	Year	Happiness Score
Switzerland	2015	7.587
Denmark	2016	7.526 Norway
2017	7.537 Finland	2018
7.632		
Finland	2019	7.769

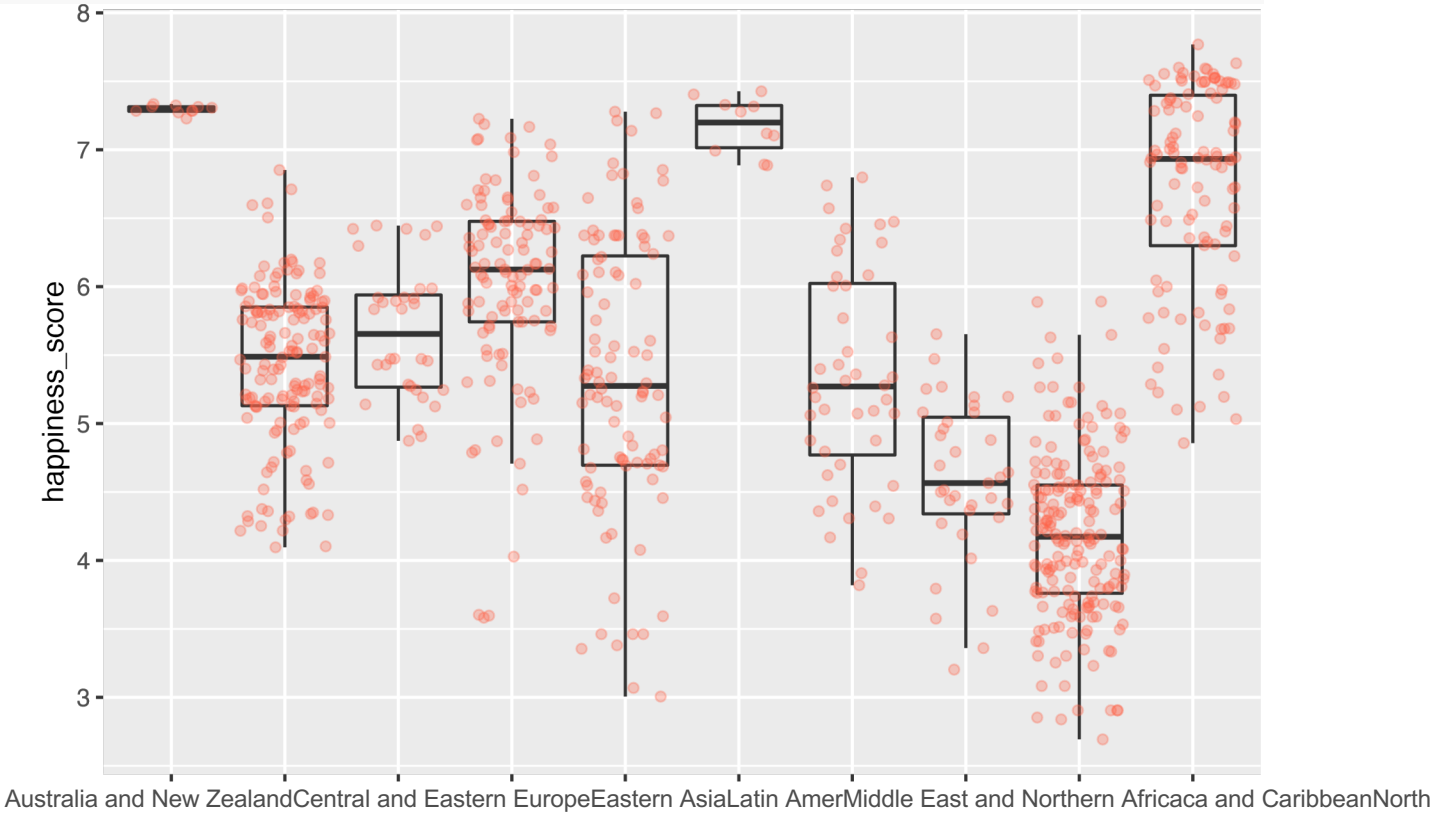
```
# From the master data set df_final we filter based on the top happiness score over the
  ars,displaying
```

As we can see from the table above we have obtained the list of all the countries that have held the top spot between the years 2016-2019.We can see that all of the countries in the list belong the European region.

To compare the happiness scores of di erent regions based on the happiness score

```
ggplot(data = df_final, aes(x = region, y = happiness_score)) +
  geom_boxplot(alpha = 0) +
  geom_jitter(alpha = 0.3, color = "tomato")
```

```
countries_top_5 <- df_final %>% dplyr::filter(happiness_rank == 1 |
                                              happiness_rank == 2 | happiness_rank
                                              == 3 | happiness_rank == 4 |
                                              happiness_rank == 5)
#select(country, year, happiness_score)
#Checking the variation of scores over the years for the top 5 countries in
all years ggplot(countries_top_5, aes(x = factor(year), y = happiness_score,
colour = country, geom_line() +ggtitle("Happiness score over the years")
```



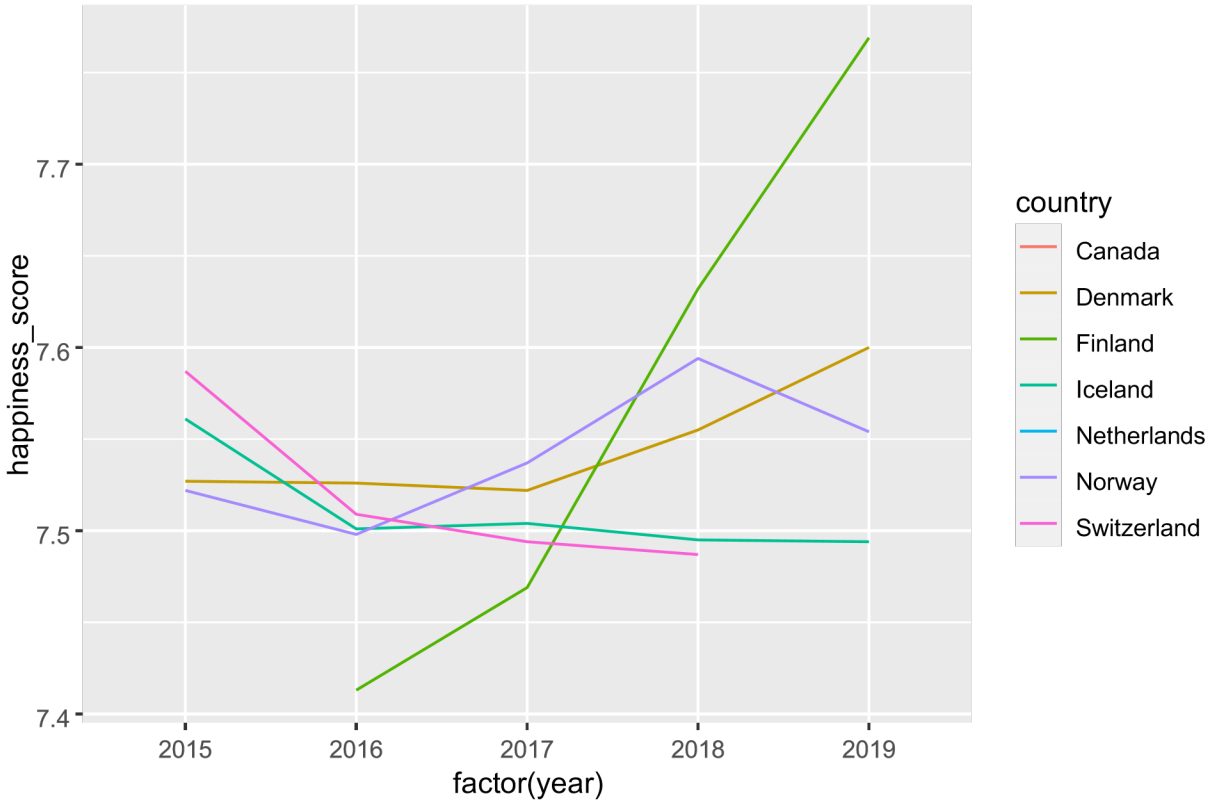

```
#We create a box plot which shows us the happiness score of different regions.As #we can
see in this box
```

Conclusion: Australia and New Zeal and is seen to have the highest average happiness score without many deviations,while the lowest is seen in the sub Saharan African region. The European region doesn't rank as high even though it has some of the happiest countries like Finland and Switzerland there are many countries which rank very low in the the happiness index.As seen in the box plot we can see that a lot of countries are shown scattered below the average happiness index line. The outliers and countries with very low happiness score tend to pull down the average for Europe

Business Question 4: To find the country with a significant hike in happiness over the years

country))

score over the years




```
## Here we are comparing the happiness scores of the top 5 countries over the years.With the x axis repr
```

Conclusion: The plot shows us that there is a significant hike in the happiness score for Finland over the years,while there is dip in the score for Switzerland over the years.

Canada and Netherlands pop up in the but not on the graph as they've appeared in the top five list only once in the 5 years

into Finland-

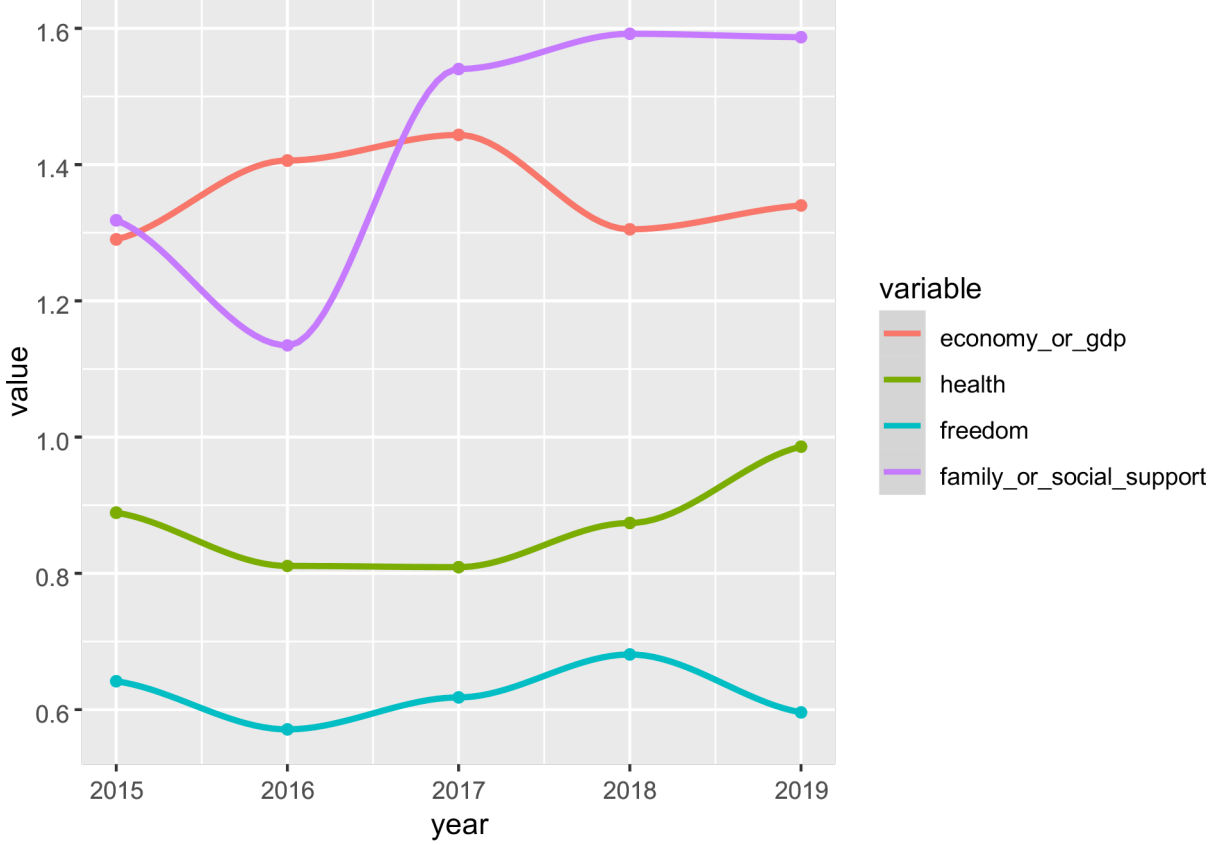
```
library(reshape2) knitr::opts_chunk$set(warning = FALSE,
message = FALSE)

#filtering the data for Finland from the final dataframe df_finland
<- df_final %>% dplyr::filter(country == Finland
)

#Excluding the categorical data columns along with the less important factors
df_finland <- select(df_finland, -region, -happiness_rank, country,
-

#reshaping the data to be suitable for making yearly graph for
d <- melt(df_finland, id.vars="year")

# plots ggplot(d, aes(year,value,
col=variable))
  + geom_point() + geom_smooth()
```

Business questions for Finland - 1) What are the factors causing the spike in Finland's happiness score from 2016 to 109? 2) Is there a specific factor that shows a similar trend to the happiness score?

Conclusion: The family or social support has seen a significant rise in Finland in the years starting from 2016 to 2019. It can be observed that a similar spike has been noticed in the happiness score for Finland in the previous graphs displaying the happiness score variation for top countries. The family_or_social_support factor could be the influencing the spike in the overall score

```
perception_of_govt_corruption, -h
```

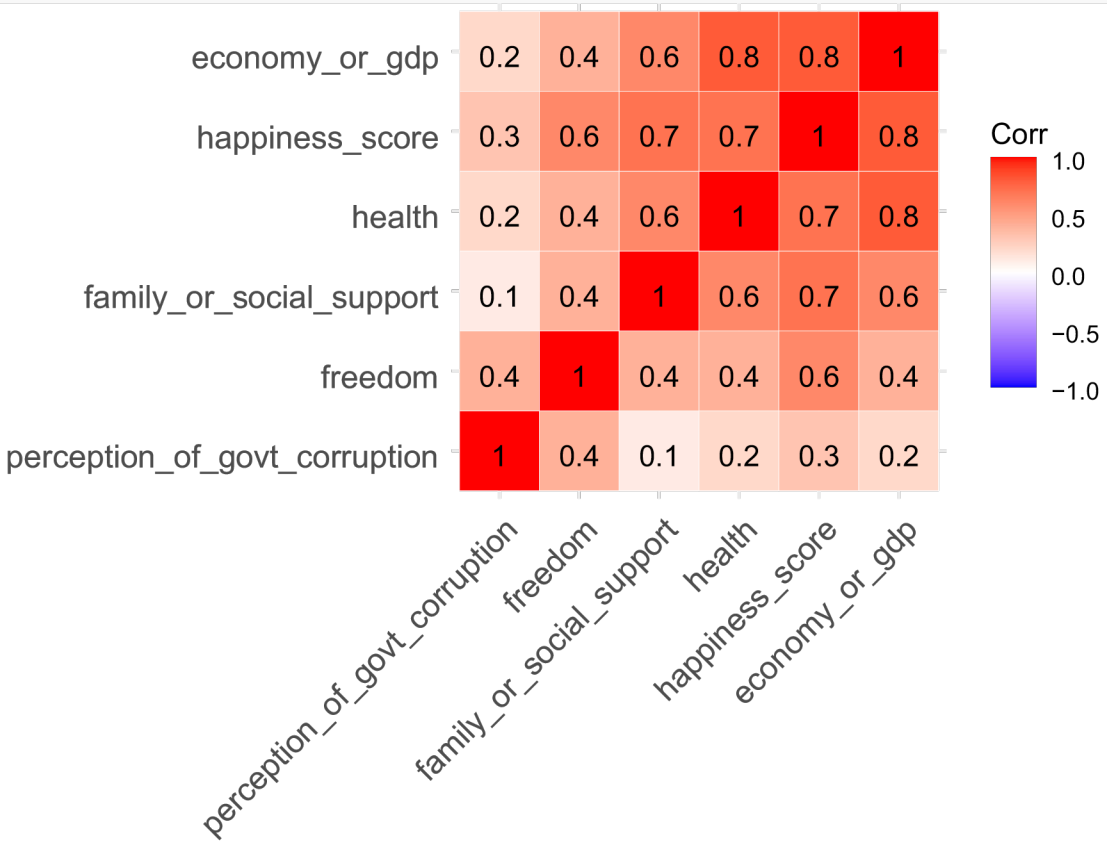
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```




```
#Deep-diving into the factors for Finland as it had the most significant rise of happiness_score amongns #Plotting the variation of the most important factors in Finland using a line plot
```

Business Question 5:To find the factors that significantly a ect the happiness ## factor.

```
correlation_df <- df_final %>% select(happiness_score,
economy_or_gdp,health ,freedom, perception_of_govt_corruption,
family_or_social_support) # Selecting the factors to be included in
the analysis.

NA_df_corr <- correlation_df[rowSums(is.na(correlation_df))

> 0,] corr <- round(cor(correlation_df), 1)

# Rounding the score up to the first decimal point.

#corr

#correlation plot

ggcorrplot(corr, hc.order = TRUE, outline.col = "white", lab = TRUE)
```




```
# We create a correlation plot to see the correlation between various factors like
GDP,health,freedom,pe
```

Conclusion: As we can see from the plot that economy_or_gdp is the most contributing factor towards the happiness score with a correlation coe cient of 0.8 followed by health and social_support/family with a coe cient of 0.7. The least contributing factor seems to be perception of government corruption with a Pearson's correlation coe cient of 0.3.

Business Question 6: To visually represent the data based on color to show the intensity of the relation between the
Conclusion:As we can see higher the value (+1)the darker the color(red).Lower the value(-1) darker the

various factors.

```
heatmaply_cor(
  cor(corr), xlab
  = "Features",
  ylab =
  "Features",
  k_col = 2,
  k_row = 2
)
# Here we create a heat map which helps us to get a visual representation of the data by
using color int
```

```
 #Joining the existing data frame with an external data source outside the cho

to get extra a df_country <- df_final %>% inner_join(select(country_data Incom


#validating the joined data colSums(is.na(df_country))
```

color(blue),which indicates how each factor is related to the other.Happiness score is strongly related to family

support and GDP hence it's more towards the red tinge,while the perception of government corruption is not as

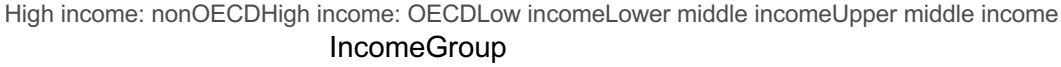
strongly related and hence it's more towards the blue tinge. Business Question 7:To compare the di erent income

groups to check if it a ects the happiness score.

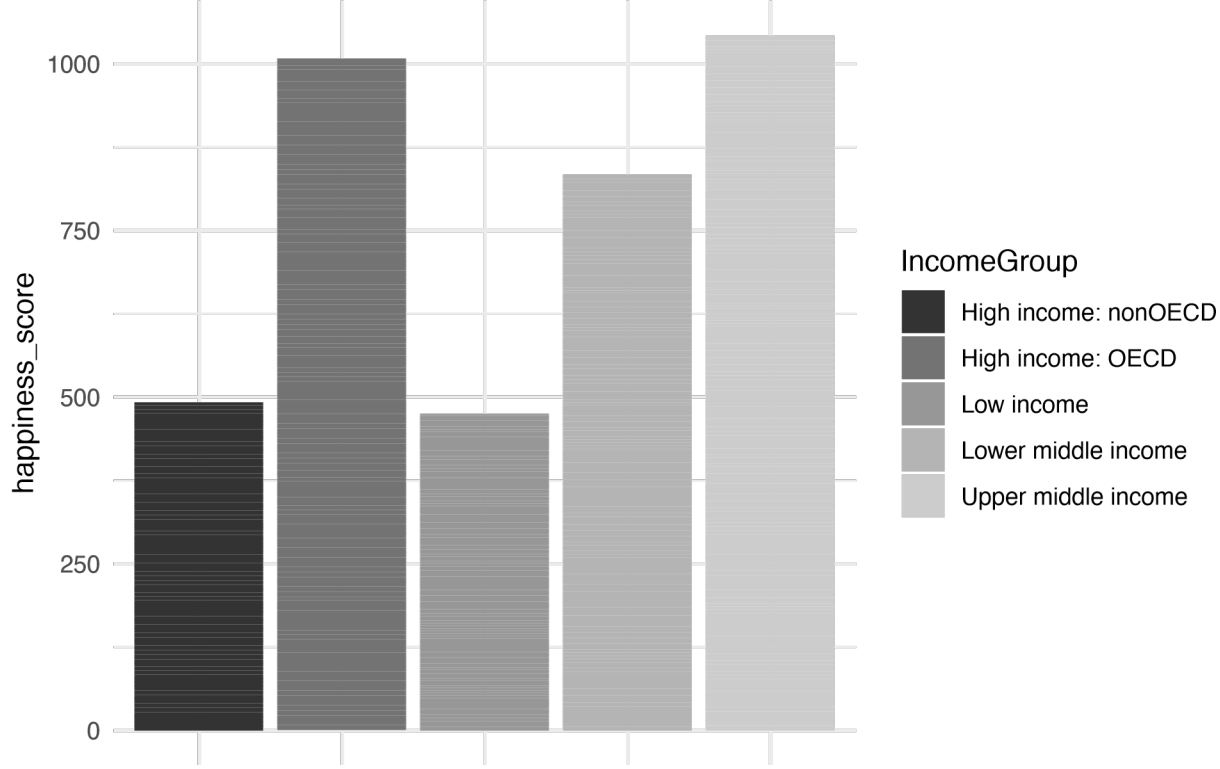
```
                                                                        c("country"="
```

```
##                     country                      region
##                           0                           0
##              happiness_rank           happiness_score
##                           0                           0
##              economy_or_gdp                      health
##                           0                           0
##                     freedom perception_of_govt_corruption

## 0 0 ## family_or_social_support year
##                           0                           0
##                 IncomeGroup
##                           0
```

```
q<-ggplot(df_country, aes(x=IncomeGroup, y=happiness_score,
fill=IncomeGroup)) + geom_bar(stat="identity")+theme_minimal() q<-
q + scale_fill_grey() q
```




```
# A bar chart is created to compare the values
```



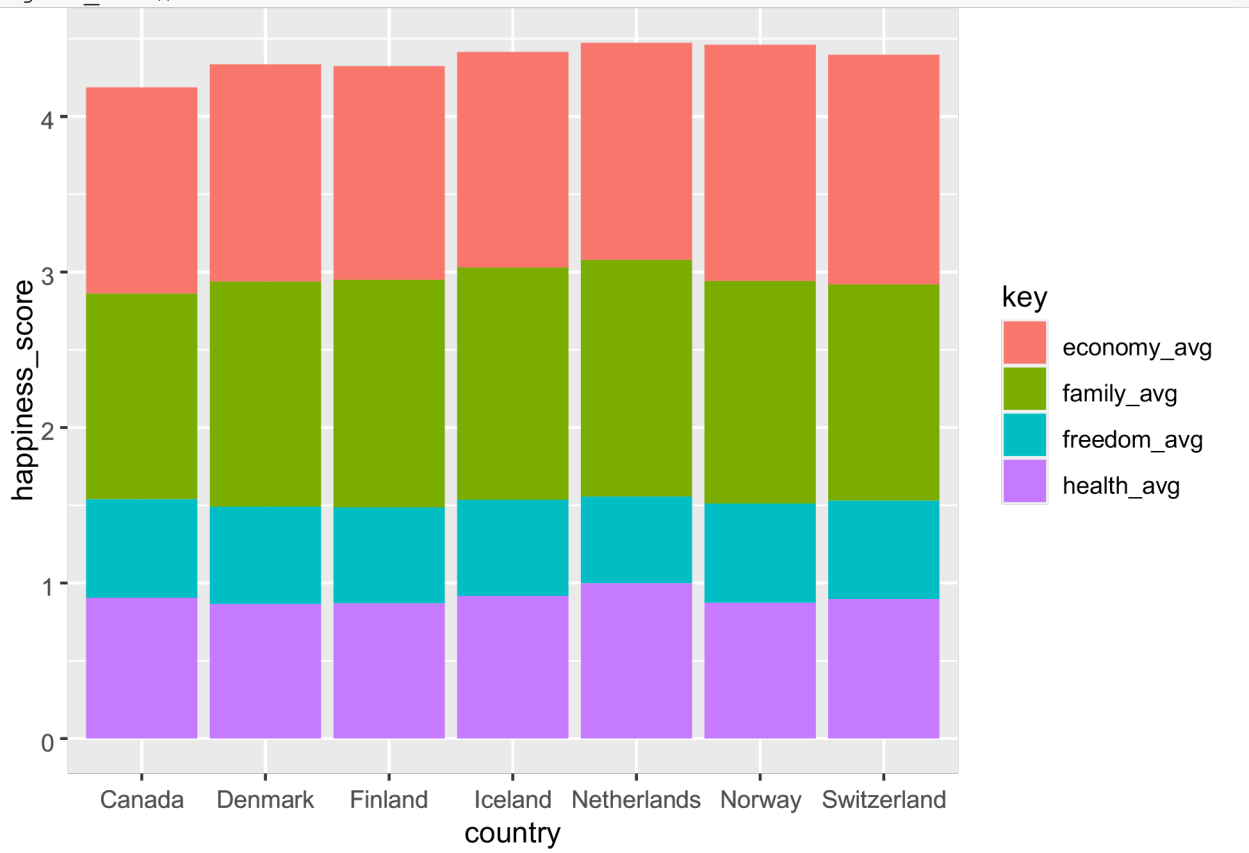

```
# In this case we consider another data set called country obtained from a government
website that conta
```

Conclusion: As we can see from the graph higher income countries don't really influence the happiness score,even though from our observations earlier it may seem that richer people are happier,but that is not the case.In fact middle and income group of countries seem to be much happier in comparison. Whereas, countries with Low income still have a low happiness score. This might imply that income or economical level matters to an extent, but after you have enough spendable income it might not matter how much extra money you have with respect to the score Note- OECD is Organisation for Economic Co-operation and Development with a group of countries. OECD has a lot of the developed countries like USA, UK, Australia, New Zealand, Canada, Italy, Finland, Iceland etc

Business Question 8:

```
# Here we create a stacked bar graph to compare the extent to which each factor
affects
the happiness sc
```

```
countries_top_5 %>% group_by(country)
  %>% summarise(n=n(), economy_avg =
  mean(economy_or_gdp), health_avg =
  mean(health),
           family_avg = mean(family_or_social_support),
           freedom_avg = mean(freedom),) %>%
  gather("key", "happiness_score", - c(country, n)) %>%
  ggplot(aes(x = country, y = happiness_score , group = key, fill = key)) +
  geom_col()
```



Conclusion: As we can see from the stacked bar graph that the family average occupies a huge chunk in the stack along with the economy followed by the health average while freedom comes in last for the counties considered.We can see that the health average and family is slightly more of an important factor in Netherlands compared to the other countries considered.

Inference: The Happiness data set helps us to get important insights into factors that seemingly a ect the happiness of the world.With di erent factors like poverty,low health systems,strained relationships and lack of freedom in many nations of the world is causing an widespread unhappiness,our attempt to take a closer look into the world's happiest and unhappiest countries and dividing the causes into basic factors to break down the possible causes,so that improvements in di erent fields may lead to happier nations and in turn a happier world. **References:**

https://www.kaggle.com/unsdsn/world-happiness

External data set- https://www.kaggle.com/kaggle/world-development-indicators