MVA_WorldHappinessAnalysis.R

Ruchi

Tue Feb 26 20:06:40 2019

```
#Ruchi Moondra
#Assignment: World Happiness Analysis
#Loading the data
worldh <- read.csv("C:/Users/Ruchi/Desktop/Ruchi/Rutgers/Multivariate/Dataset/WH_2017.cs
v")

#Loading packages required for the analysis
library(plyr)
library(plotly)
```

```
## Loading required package: ggplot2
```

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

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

```
## The following objects are masked from 'package:plyr':
##
##     arrange, mutate, rename, summarise
```

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

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

```
library(dplyr)
```

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

```
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

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

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

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------
---------------- tidyverse 1.2.1 --
```

```
## v tibble  2.0.1     v purrr   0.3.0
## v tidyr   0.8.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.3.0
```

```
## -- Conflicts ------------------------------------------------------------------
---------- tidyverse_conflicts() --
## x dplyr::arrange()   masks plotly::arrange(), plyr::arrange()
## x purrr::compact()   masks plyr::compact()
## x dplyr::count()     masks plyr::count()
## x dplyr::failwith()  masks plyr::failwith()
## x dplyr::filter()    masks plotly::filter(), stats::filter()
## x dplyr::id()        masks plyr::id()
## x dplyr::lag()       masks stats::lag()
## x dplyr::mutate()    masks plotly::mutate(), plyr::mutate()
## x dplyr::rename()    masks plotly::rename(), plyr::rename()
## x dplyr::summarise() masks plotly::summarise(), plyr::summarise()
## x dplyr::summarize() masks plyr::summarize()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:plyr':
##
##     here
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
library(caTools)
library(ggplot2)
library(ggthemes)
library(reshape2)
```

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

```
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:reshape2':
##
##     dcast, melt
```

```
## The following objects are masked from 'package:lubridate':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday,
##     week, yday, year
```

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

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

```
library(tidyr)
library(corrgram)
```

```
##
## Attaching package: 'corrgram'
```

```
## The following object is masked from 'package:plyr':
##
##     baseball
```

```
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
library(formattable)
```

```
##
## Attaching package: 'formattable'
```

```
## The following object is masked from 'package:plotly':
##
##     style
```

```
library(cowplot)
```

```
##
## Attaching package: 'cowplot'
```

```
## The following object is masked from 'package:ggthemes':
##
##     theme_map
```

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

```
library(ggpubr)
```

```
## Loading required package: magrittr
```

```
##
## Attaching package: 'magrittr'
```

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

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
##
## Attaching package: 'ggpubr'
```

```
## The following object is masked from 'package:cowplot':
##
##     get_legend
```

```
## The following object is masked from 'package:plyr':
##
##     mutate
```

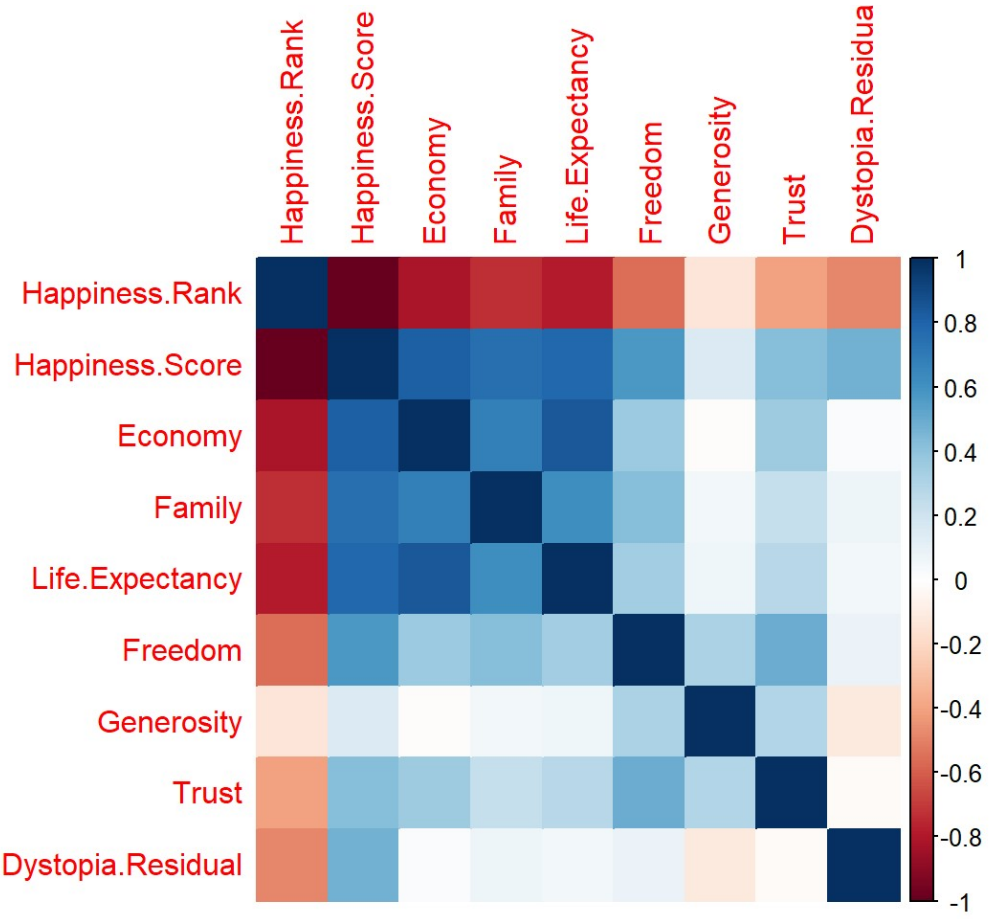
```
library(plot3D)

#View the data
View(worldh)
#Displays the first few rows of the dataset
head(worldh)
```

```
##       Country Happiness.Rank Happiness.Score Whisker.high Whisker.low
## 1      Norway              1           7.537     7.594445    7.479556
## 2     Denmark              2           7.522     7.581728    7.462272
## 3     Iceland              3           7.504     7.622030    7.385970
## 4 Switzerland              4           7.494     7.561772    7.426227
## 5     Finland              5           7.469     7.527542    7.410458
## 6 Netherlands              6           7.377     7.427426    7.326574
##   Economy..GDP.per.Capita.   Family Health..Life.Expectancy.   Freedom
## 1                1.616463 1.533524               0.7966665 0.6354226
## 2                1.482383 1.551122               0.7925655 0.6260067
## 3                1.480633 1.610574               0.8335521 0.6271626
## 4                1.564980 1.516912               0.8581313 0.6200706
## 5                1.443572 1.540247               0.8091577 0.6179509
## 6                1.503945 1.428939               0.8106961 0.5853845
##   Generosity Trust..Government.Corruption. Dystopia.Residual
## 1  0.3620122                     0.3159638          2.277027
## 2  0.3552805                     0.4007701          2.313707
## 3  0.4755402                     0.1535266          2.322715
## 4  0.2905493                     0.3670073          2.276716
## 5  0.2454828                     0.3826115          2.430182
## 6  0.4704898                     0.2826618          2.294804
```

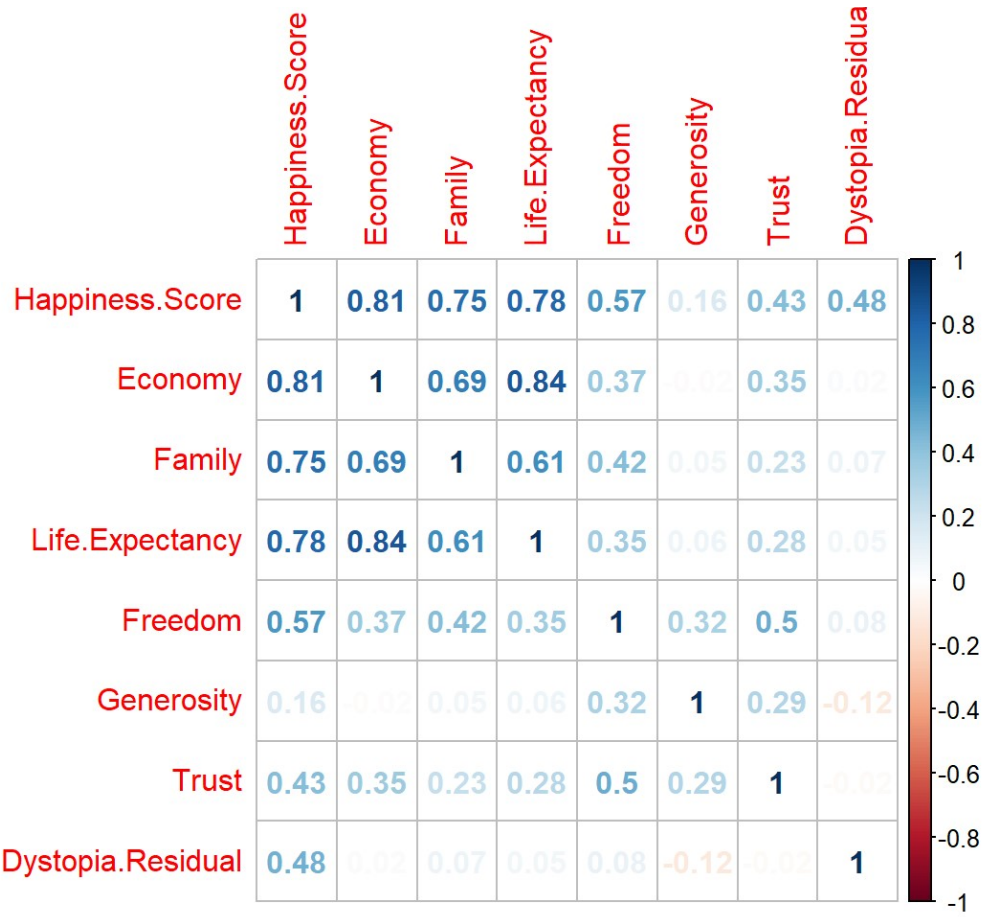
```
#Display the structure of the attributes
str(worldh)
```

```
## 'data.frame':    155 obs. of  12 variables:
##  $ Country                      : Factor w/ 155 levels "Afghanistan",..: 105 38 58 13
3 45 99 26 100 132 7 ...
##  $ Happiness.Rank               : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Happiness.Score              : num  7.54 7.52 7.5 7.49 7.47 ...
##  $ Whisker.high                 : num  7.59 7.58 7.62 7.56 7.53 ...
##  $ Whisker.low                  : num  7.48 7.46 7.39 7.43 7.41 ...
##  $ Economy..GDP.per.Capita.     : num  1.62 1.48 1.48 1.56 1.44 ...
##  $ Family                       : num  1.53 1.55 1.61 1.52 1.54 ...
##  $ Health..Life.Expectancy.     : num  0.797 0.793 0.834 0.858 0.809 ...
##  $ Freedom                      : num  0.635 0.626 0.627 0.62 0.618 ...
##  $ Generosity                   : num  0.362 0.355 0.476 0.291 0.245 ...
##  $ Trust..Government.Corruption.: num  0.316 0.401 0.154 0.367 0.383 ...
##  $ Dystopia.Residual            : num  2.28 2.31 2.32 2.28 2.43 ...
```

```
# Changing the name of columns
colnames (worldh) <- c("Country", "Happiness.Rank", "Happiness.Score",
                       "Whisker.High", "Whisker.Low", "Economy", "Family",
                       "Life.Expectancy", "Freedom", "Generosity",
                       "Trust", "Dystopia.Residual")

# Deleting unnecessary columns (Whisker.high and Whisker.low)
worldh <- worldh[, -c(4,5)]

# Finding the correlation between numerical columns
Num.cols <- sapply(worldh, is.numeric)
Cor.data <- cor(worldh[, Num.cols])
corrplot(Cor.data, method = 'color')
```

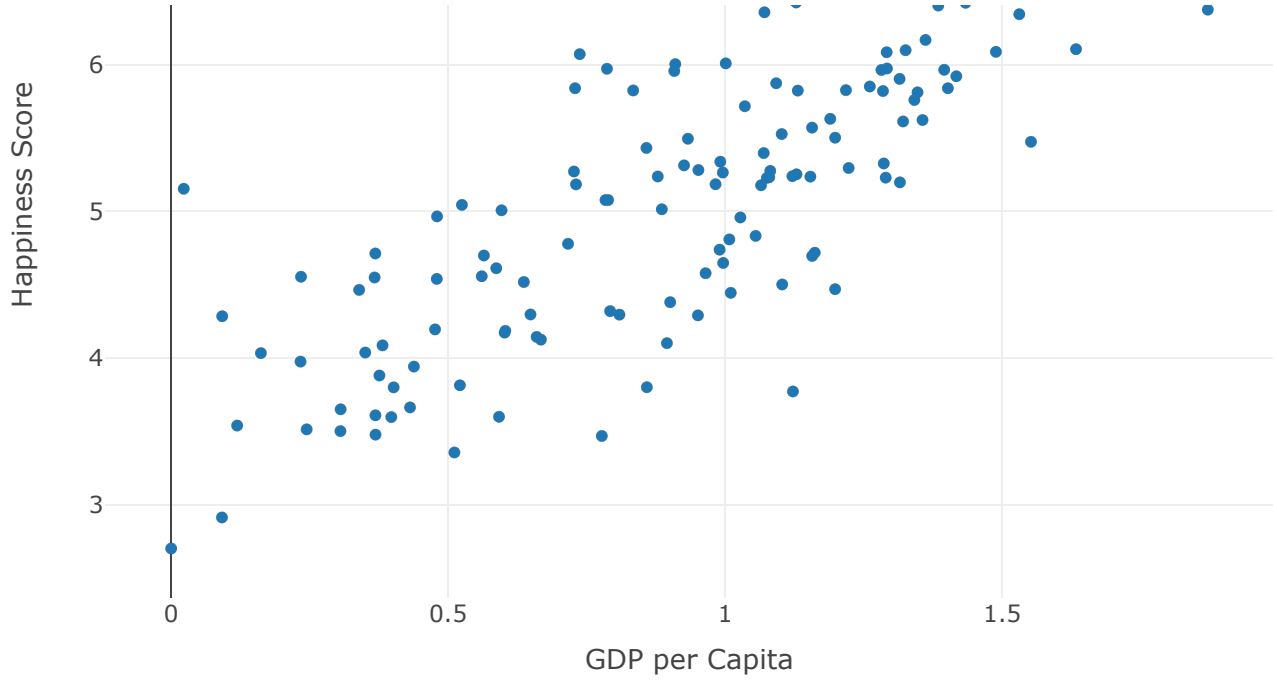



```
#Analysis: We can see there is an inverse correlation between "Happiness Rank" and all th
e other numerical variables. In other words, the lower the happiness rank, the higher th
e happiness score, and the higher the other seven factors that contribute to happiness. S
o let's remove the happiness rank, and see the correlation again.

# Create a correlation plot
newdatacor = cor(worldh[c(3:10)])
corrplot(newdatacor, method = "number")
```




```
#Analysis: In the above cor plot, Economy, life expectancy, and family play the most sign
ificant role in contributing to happiness.
#Trust and generosity have the lowest impact on the happiness score.

#Plotting ScatterPLot
plot_ly(data = worldh,
        x=~Economy, y=~Happiness.Score, type = "scatter",
        text = ~paste("Country:", Country)) %>%
  layout(title = "Happiness and GDP",
         xaxis = list(title = "GDP per Capita"),
         yaxis = list(title = "Happiness Score"))
```

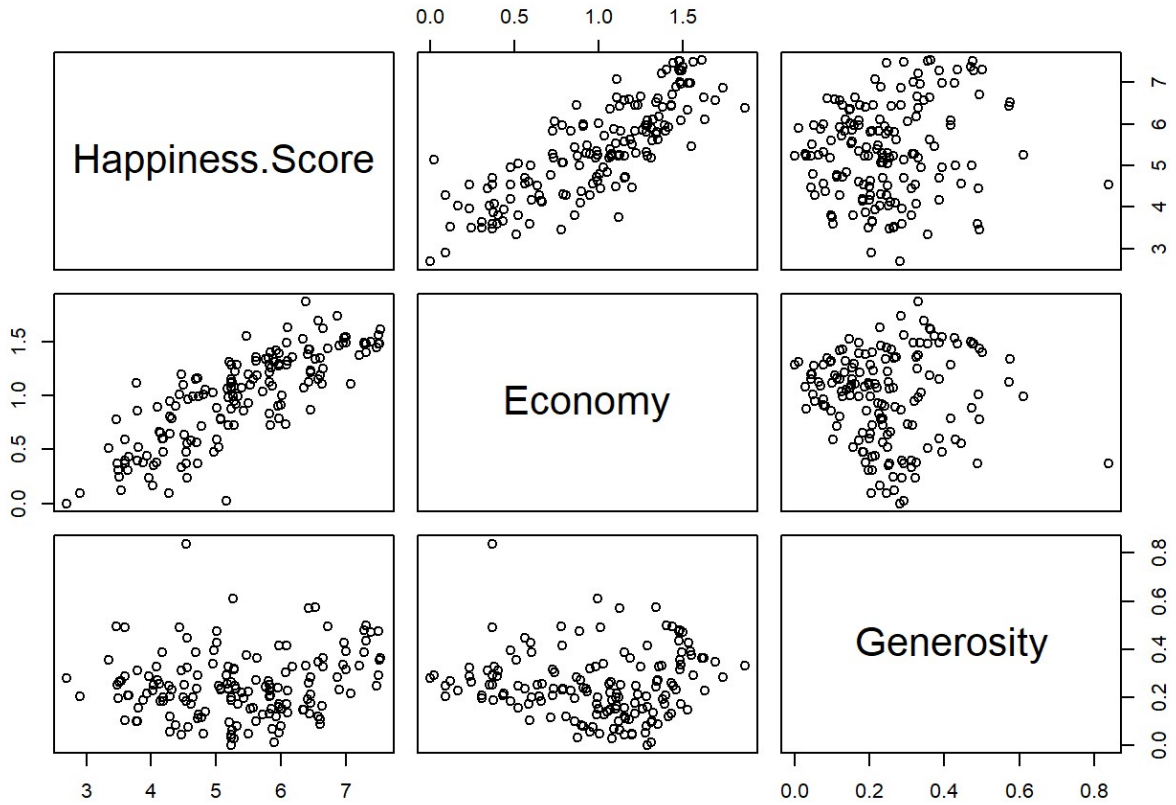
```
## No scatter mode specifed:
##   Setting the mode to markers
##   Read more about this attribute -> https://plot.ly/r/reference/#scatter-mode
```



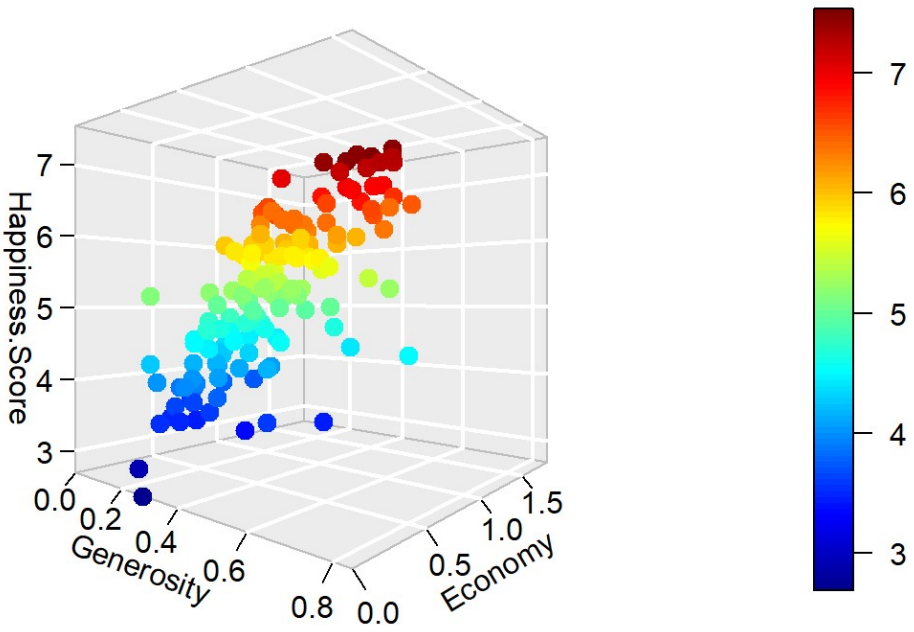

```
#Analysis: This interactive scatterplot shows that there is a strong positive correlation
between GDP and Happiness.

#Let's do multiple Regression
dat <- worldh[c("Happiness.Score","Economy","Generosity")]
head(dat)
```

```
##   Happiness.Score  Economy Generosity
## 1           7.537 1.616463  0.3620122
## 2           7.522 1.482383  0.3552805
## 3           7.504 1.480633  0.4755402
## 4           7.494 1.564980  0.2905493
## 5           7.469 1.443572  0.2454828
## 6           7.377 1.503945  0.4704898
```

```
plot(dat)
```




```
#It seems like there is a positive correlation between economy and happiness score but th
is is not true between happiness score
#and generosity.

#3D plot of same
scatter3D(dat$Generosity, dat$Economy, dat$Happiness.Score, phi = 0, bty = "g",
          pch = 20, cex = 2, ticktype = "detailed",
          main = "Happiness data", xlab = "Generosity",
          ylab ="Economy", zlab = "Happiness.Score")
```




```
#From the scatter plot we cannot determine that combination of high economy and generosit
y leads to greater happiness score.
#This is something we have to conclude after analyzing the effect of these 2 taken togeth
er.
```