eda-la1

Author: K Arshitha USN: 1NT20IS068

2023-05-17

## R markdown

dataset <-read.csv("C:/Users/Dataset/archive/2015.csv")  
library(readxl)

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

grouped\_data <- dataset %>% group\_by(Region) %>% summarise(avg\_happiness = mean(Happiness.Score))

## to view summary of the dataset

head(dataset)

## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 1 Switzerland Western Europe 1 7.587 0.03411  
## 2 Iceland Western Europe 2 7.561 0.04884  
## 3 Denmark Western Europe 3 7.527 0.03328  
## 4 Norway Western Europe 4 7.522 0.03880  
## 5 Canada North America 5 7.427 0.03553  
## 6 Finland Western Europe 6 7.406 0.03140  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 1 1.39651 1.34951 0.94143 0.66557  
## 2 1.30232 1.40223 0.94784 0.62877  
## 3 1.32548 1.36058 0.87464 0.64938  
## 4 1.45900 1.33095 0.88521 0.66973  
## 5 1.32629 1.32261 0.90563 0.63297  
## 6 1.29025 1.31826 0.88911 0.64169  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 1 0.41978 0.29678 2.51738  
## 2 0.14145 0.43630 2.70201  
## 3 0.48357 0.34139 2.49204  
## 4 0.36503 0.34699 2.46531  
## 5 0.32957 0.45811 2.45176  
## 6 0.41372 0.23351 2.61955

# row name of dataset

rownames(dataset)

## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"   
## [13] "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23" "24"   
## [25] "25" "26" "27" "28" "29" "30" "31" "32" "33" "34" "35" "36"   
## [37] "37" "38" "39" "40" "41" "42" "43" "44" "45" "46" "47" "48"   
## [49] "49" "50" "51" "52" "53" "54" "55" "56" "57" "58" "59" "60"   
## [61] "61" "62" "63" "64" "65" "66" "67" "68" "69" "70" "71" "72"   
## [73] "73" "74" "75" "76" "77" "78" "79" "80" "81" "82" "83" "84"   
## [85] "85" "86" "87" "88" "89" "90" "91" "92" "93" "94" "95" "96"   
## [97] "97" "98" "99" "100" "101" "102" "103" "104" "105" "106" "107" "108"  
## [109] "109" "110" "111" "112" "113" "114" "115" "116" "117" "118" "119" "120"  
## [121] "121" "122" "123" "124" "125" "126" "127" "128" "129" "130" "131" "132"  
## [133] "133" "134" "135" "136" "137" "138" "139" "140" "141" "142" "143" "144"  
## [145] "145" "146" "147" "148" "149" "150" "151" "152" "153" "154" "155" "156"  
## [157] "157" "158"

## to view colonames in a dataset

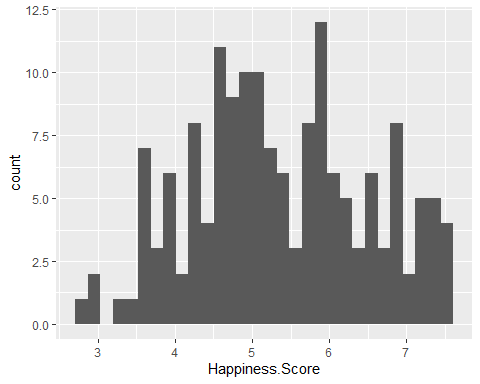
colnames(dataset)

## [1] "Country" "Region"   
## [3] "Happiness.Rank" "Happiness.Score"   
## [5] "Standard.Error" "Economy..GDP.per.Capita."   
## [7] "Family" "Health..Life.Expectancy."   
## [9] "Freedom" "Trust..Government.Corruption."  
## [11] "Generosity" "Dystopia.Residual"

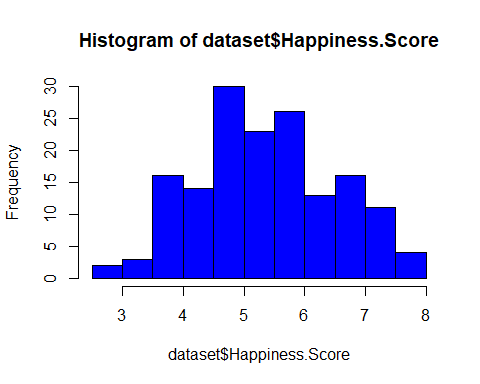
#to plot histogram

library(ggplot2)  
ggplot(dataset,aes(x=Happiness.Score))+geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

 # to plot histogram with a color

hist(dataset$Happiness.Score,col="blue")

 # to determine the length of a vector

length(vector())

## [1] 0

# to find the mean of coloumn

mean(dataset$Happiness.Score)

## [1] 5.375734

# to find median of coloumn

median(dataset$Happiness.Rank)

## [1] 79.5

#By changing the options(“max.print”) value, you can set the default number of rows to display.

options(max.print = 30)

#to find the dimensions of the dataset

dim(dataset)

## [1] 158 12

# to find the number of rows in the dataset

nrow(dataset)

## [1] 158

# to find number of coloumns in a datset

ncol(dataset)

## [1] 12

# to show the coloumn names in the dataset

names(dataset)

## [1] "Country" "Region"   
## [3] "Happiness.Rank" "Happiness.Score"   
## [5] "Standard.Error" "Economy..GDP.per.Capita."   
## [7] "Family" "Health..Life.Expectancy."   
## [9] "Freedom" "Trust..Government.Corruption."  
## [11] "Generosity" "Dystopia.Residual"

# to view the last few rows ofa dataset

tail(dataset)

## Country Region Happiness.Rank Happiness.Score  
## 153 Afghanistan Southern Asia 153 3.575  
## 154 Rwanda Sub-Saharan Africa 154 3.465  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 153 0.03084 0.31982 0.30285 0.30335  
## 154 0.03464 0.22208 0.77370 0.42864  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 153 0.23414 0.09719 0.36510 1.95210  
## 154 0.59201 0.55191 0.22628 0.67042  
## [ reached 'max' / getOption("max.print") -- omitted 4 rows ]

# to extract a subset of a datframe based on a condition

subset(dataset)

## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 1 Switzerland Western Europe 1 7.587 0.03411  
## 2 Iceland Western Europe 2 7.561 0.04884  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 1 1.39651 1.34951 0.94143 0.66557  
## 2 1.30232 1.40223 0.94784 0.62877  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 1 0.41978 0.29678 2.51738  
## 2 0.14145 0.43630 2.70201  
## [ reached 'max' / getOption("max.print") -- omitted 156 rows ]

# to select only specific coloumn from a dataframe

library(dplyr)  
dataset %>% select(2,3)

## Region Happiness.Rank  
## 1 Western Europe 1  
## 2 Western Europe 2  
## 3 Western Europe 3  
## 4 Western Europe 4  
## 5 North America 5  
## 6 Western Europe 6  
## 7 Western Europe 7  
## 8 Western Europe 8  
## 9 Australia and New Zealand 9  
## 10 Australia and New Zealand 10  
## 11 Middle East and Northern Africa 11  
## 12 Latin America and Caribbean 12  
## 13 Western Europe 13  
## 14 Latin America and Caribbean 14  
## 15 North America 15  
## [ reached 'max' / getOption("max.print") -- omitted 143 rows ]

# to Select columns by list of index or position

dataset%>% select(c(2,3))

## Region Happiness.Rank  
## 1 Western Europe 1  
## 2 Western Europe 2  
## 3 Western Europe 3  
## 4 Western Europe 4  
## 5 North America 5  
## 6 Western Europe 6  
## 7 Western Europe 7  
## 8 Western Europe 8  
## 9 Australia and New Zealand 9  
## 10 Australia and New Zealand 10  
## 11 Middle East and Northern Africa 11  
## 12 Latin America and Caribbean 12  
## 13 Western Europe 13  
## 14 Latin America and Caribbean 14  
## 15 North America 15  
## [ reached 'max' / getOption("max.print") -- omitted 143 rows ]

# to select coloumn by index range

dataset %>% select(2:3)

## Region Happiness.Rank  
## 1 Western Europe 1  
## 2 Western Europe 2  
## 3 Western Europe 3  
## 4 Western Europe 4  
## 5 North America 5  
## 6 Western Europe 6  
## 7 Western Europe 7  
## 8 Western Europe 8  
## 9 Australia and New Zealand 9  
## 10 Australia and New Zealand 10  
## 11 Middle East and Northern Africa 11  
## 12 Latin America and Caribbean 12  
## 13 Western Europe 13  
## 14 Latin America and Caribbean 14  
## 15 North America 15  
## [ reached 'max' / getOption("max.print") -- omitted 143 rows ]

# to select coloums by specific name

dataset %>% select(Country,Region)

## Country Region  
## 1 Switzerland Western Europe  
## 2 Iceland Western Europe  
## 3 Denmark Western Europe  
## 4 Norway Western Europe  
## 5 Canada North America  
## 6 Finland Western Europe  
## 7 Netherlands Western Europe  
## 8 Sweden Western Europe  
## 9 New Zealand Australia and New Zealand  
## 10 Australia Australia and New Zealand  
## 11 Israel Middle East and Northern Africa  
## 12 Costa Rica Latin America and Caribbean  
## 13 Austria Western Europe  
## 14 Mexico Latin America and Caribbean  
## 15 United States North America  
## [ reached 'max' / getOption("max.print") -- omitted 143 rows ]

# to select coloumns except the mentioned coloumns

dataset %>% select(-c('Country','Region'))

## Happiness.Rank Happiness.Score Standard.Error Economy..GDP.per.Capita.  
## 1 1 7.587 0.03411 1.39651  
## 2 2 7.561 0.04884 1.30232  
## 3 3 7.527 0.03328 1.32548  
## Family Health..Life.Expectancy. Freedom Trust..Government.Corruption.  
## 1 1.34951 0.94143 0.66557 0.41978  
## 2 1.40223 0.94784 0.62877 0.14145  
## 3 1.36058 0.87464 0.64938 0.48357  
## Generosity Dystopia.Residual  
## 1 0.29678 2.51738  
## 2 0.43630 2.70201  
## 3 0.34139 2.49204  
## [ reached 'max' / getOption("max.print") -- omitted 155 rows ]

# to Select columns between specfied coloums

dataset %>% select('Happiness.Rank':'Family')

## Happiness.Rank Happiness.Score Standard.Error Economy..GDP.per.Capita.  
## 1 1 7.587 0.03411 1.39651  
## 2 2 7.561 0.04884 1.30232  
## 3 3 7.527 0.03328 1.32548  
## 4 4 7.522 0.03880 1.45900  
## 5 5 7.427 0.03553 1.32629  
## 6 6 7.406 0.03140 1.29025  
## Family  
## 1 1.34951  
## 2 1.40223  
## 3 1.36058  
## 4 1.33095  
## 5 1.32261  
## 6 1.31826  
## [ reached 'max' / getOption("max.print") -- omitted 152 rows ]

# to Select columns starts with a string

dataset %>% select(starts\_with('l'))

## data frame with 0 columns and 158 rows

# to Select columns that ends with a string

dataset %>% select(starts\_with('l'))

## data frame with 0 columns and 158 rows

# to Select columns that contains

dataset %>% select(contains('p'))

## Happiness.Rank Happiness.Score Economy..GDP.per.Capita.  
## 1 1 7.587 1.39651  
## 2 2 7.561 1.30232  
## 3 3 7.527 1.32548  
## 4 4 7.522 1.45900  
## 5 5 7.427 1.32629  
## Health..Life.Expectancy. Trust..Government.Corruption. Dystopia.Residual  
## 1 0.94143 0.41978 2.51738  
## 2 0.94784 0.14145 2.70201  
## 3 0.87464 0.48357 2.49204  
## 4 0.88521 0.36503 2.46531  
## 5 0.90563 0.32957 2.45176  
## [ reached 'max' / getOption("max.print") -- omitted 153 rows ]

# Select all numeric columns

dataset %>% select\_if(is.numeric)

## Happiness.Rank Happiness.Score Standard.Error Economy..GDP.per.Capita.  
## 1 1 7.587 0.03411 1.39651  
## 2 2 7.561 0.04884 1.30232  
## 3 3 7.527 0.03328 1.32548  
## Family Health..Life.Expectancy. Freedom Trust..Government.Corruption.  
## 1 1.34951 0.94143 0.66557 0.41978  
## 2 1.40223 0.94784 0.62877 0.14145  
## 3 1.36058 0.87464 0.64938 0.48357  
## Generosity Dystopia.Residual  
## 1 0.29678 2.51738  
## 2 0.43630 2.70201  
## 3 0.34139 2.49204  
## [ reached 'max' / getOption("max.print") -- omitted 155 rows ]

# filter by a row name

filter(dataset, Country %in% c("21", "39", "33"))

## [1] Country Region   
## [3] Happiness.Rank Happiness.Score   
## [5] Standard.Error Economy..GDP.per.Capita.   
## [7] Family Health..Life.Expectancy.   
## [9] Freedom Trust..Government.Corruption.  
## [11] Generosity Dystopia.Residual   
## <0 rows> (or 0-length row.names)

# filter() by row number

slice(dataset,5)

## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 1 Canada North America 5 7.427 0.03553  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 1 1.32629 1.32261 0.90563 0.63297  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 1 0.32957 0.45811 2.45176

## Using arrange in descending order

temp <- dataset %>% arrange(desc(Happiness.Rank))  
 temp

## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 1 Togo Sub-Saharan Africa 158 2.839 0.06727  
## 2 Burundi Sub-Saharan Africa 157 2.905 0.08658  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 1 0.20868 0.13995 0.28443 0.36453  
## 2 0.01530 0.41587 0.22396 0.11850  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 1 0.10731 0.16681 1.56726  
## 2 0.10062 0.19727 1.83302  
## [ reached 'max' / getOption("max.print") -- omitted 156 rows ]

# Using arrange by multiple columns

temp <- dataset %>% arrange(Happiness.Rank,desc(Country))  
 temp

## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 1 Switzerland Western Europe 1 7.587 0.03411  
## 2 Iceland Western Europe 2 7.561 0.04884  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 1 1.39651 1.34951 0.94143 0.66557  
## 2 1.30232 1.40223 0.94784 0.62877  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 1 0.41978 0.29678 2.51738  
## 2 0.14145 0.43630 2.70201  
## [ reached 'max' / getOption("max.print") -- omitted 156 rows ]

# Load dplyr

library(dplyr)

#group\_by() on department

grp\_tbl <- dataset %>% group\_by(Country)  
grp\_tbl

## # A tibble: 158 × 12  
## # Groups: Country [158]  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## <chr> <chr> <int> <dbl> <dbl>  
## 1 Switzerland Western Europe 1 7.59 0.0341  
## 2 Iceland Western Europe 2 7.56 0.0488  
## 3 Denmark Western Europe 3 7.53 0.0333  
## 4 Norway Western Europe 4 7.52 0.0388  
## 5 Canada North America 5 7.43 0.0355  
## 6 Finland Western Europe 6 7.41 0.0314  
## 7 Netherlands Western Europe 7 7.38 0.0280  
## 8 Sweden Western Europe 8 7.36 0.0316  
## 9 New Zealand Australia and New … 9 7.29 0.0337  
## 10 Australia Australia and New … 10 7.28 0.0408  
## # ℹ 148 more rows  
## # ℹ 7 more variables: Economy..GDP.per.Capita. <dbl>, Family <dbl>,  
## # Health..Life.Expectancy. <dbl>, Freedom <dbl>,  
## # Trust..Government.Corruption. <dbl>, Generosity <dbl>,  
## # Dystopia.Residual <dbl>

## Convert tibble to DataFrame

df2 <- dataset %>% as.data.frame()  
 class(df2)

## [1] "data.frame"

#Group by on multiple column

agg\_tbl <- dataset %>% group\_by(Country) %>%   
 summarise(Happiness.Rank=sum(Happiness.Score))  
agg\_tbl

## # A tibble: 158 × 2  
## Country Happiness.Rank  
## <chr> <dbl>  
## 1 Afghanistan 3.58  
## 2 Albania 4.96  
## 3 Algeria 5.60  
## 4 Angola 4.03  
## 5 Argentina 6.57  
## 6 Armenia 4.35  
## 7 Australia 7.28  
## 8 Austria 7.2   
## 9 Azerbaijan 5.21  
## 10 Bahrain 5.96  
## # ℹ 148 more rows

## is.na(): Checks for missing values in a data frame

is.na(dataset)

## Country Region Happiness.Rank Happiness.Score Standard.Error  
## [1,] FALSE FALSE FALSE FALSE FALSE  
## [2,] FALSE FALSE FALSE FALSE FALSE  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## [1,] FALSE FALSE FALSE FALSE  
## [2,] FALSE FALSE FALSE FALSE  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## [1,] FALSE FALSE FALSE  
## [2,] FALSE FALSE FALSE  
## [ reached getOption("max.print") -- omitted 156 rows ]

## complete.cases(): Removes rows with missing values

complete.cases(dataset)

## [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE  
## [16] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE  
## [ reached getOption("max.print") -- omitted 128 entries ]

## mean(): Calculates the mean of a numeric vector

mean(dataset$Happiness.Score)

## [1] 5.375734

# median(): Calculates the median of a numeric vector

median(dataset$Happiness.Rank)

## [1] 79.5

#sd():Calculates the sd of a numeric data

sd(dataset$Economy..GDP.per.Capita.)

## [1] 0.4031208

# var(): Calculates the variance of a numeric vector

var(dataset$Freedom)

## [1] 0.02270832

## max(): Finds the maximum value in a numeric vector

max(dataset$Happiness.Rank)

## [1] 158

#min(): finds the minimum value in a numeric vector

min(dataset$Happiness.Rank)

## [1] 1

## quantile(): Calculates the quantiles of a numeric vector

quantile(dataset$Happiness.Rank)

## 0% 25% 50% 75% 100%   
## 1.00 40.25 79.50 118.75 158.00

## unique(): Finds the unique values of a vector or column in a data frame

unique(dataset$Counry)

## NULL

# duplicated(): Identifies duplicated values in a vector or column in a data frame

duplicated(dataset$Country)

## [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
## [13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
## [25] FALSE FALSE FALSE FALSE FALSE FALSE  
## [ reached getOption("max.print") -- omitted 128 entries ]

# To divide a vector into groups, use the split() function.

split(dataset, f = dataset$Country)

## $Afghanistan  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 153 Afghanistan Southern Asia 153 3.575 0.03084  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 153 0.31982 0.30285 0.30335 0.23414  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 153 0.09719 0.3651 1.9521  
##   
## $Albania  
## Country Region Happiness.Rank Happiness.Score  
## 95 Albania Central and Eastern Europe 95 4.959  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 95 0.05013 0.87867 0.80434 0.81325  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 95 0.35733 0.06413 0.14272 1.89894  
##   
## $Algeria  
## Country Region Happiness.Rank Happiness.Score  
## 68 Algeria Middle East and Northern Africa 68 5.605  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 68 0.05099 0.93929 1.07772 0.61766  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 68 0.28579 0.17383 0.07822 2.43209  
##   
## $Angola  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 137 Angola Sub-Saharan Africa 137 4.033 0.04758  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 137 0.75778 0.8604 0.16683 0.10384  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 137 0.07122 0.12344 1.94939  
##   
## $Argentina  
## Country Region Happiness.Rank Happiness.Score  
## 30 Argentina Latin America and Caribbean 30 6.574  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 30 0.04612 1.05351 1.24823 0.78723  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 30 0.44974 0.08484 0.11451 2.836  
##   
## $Armenia  
## Country Region Happiness.Rank Happiness.Score  
## 127 Armenia Central and Eastern Europe 127 4.35  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 127 0.04763 0.76821 0.77711 0.7299  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 127 0.19847 0.039 0.07855 1.75873  
##   
## $Australia  
## Country Region Happiness.Rank Happiness.Score  
## 10 Australia Australia and New Zealand 10 7.284  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 10 0.04083 1.33358 1.30923 0.93156  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 10 0.65124 0.35637 0.43562 2.26646  
##   
## $Austria  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 13 Austria Western Europe 13 7.2 0.03751  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 13 1.33723 1.29704 0.89042 0.62433  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 13 0.18676 0.33088 2.5332  
##   
## $Azerbaijan  
## Country Region Happiness.Rank Happiness.Score  
## 80 Azerbaijan Central and Eastern Europe 80 5.212  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 80 0.03363 1.02389 0.93793 0.64045  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 80 0.3703 0.16065 0.07799 2.00073  
##   
## $Bahrain  
## Country Region Happiness.Rank Happiness.Score  
## 49 Bahrain Middle East and Northern Africa 49 5.96  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 49 0.05412 1.32376 1.21624 0.74716  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 49 0.45492 0.306 0.17362 1.73797  
##   
## $Bangladesh  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 109 Bangladesh Southern Asia 109 4.694 0.03077  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 109 0.39753 0.43106 0.60164 0.4082  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 109 0.12569 0.21222 2.51767  
##   
## $Belarus  
## Country Region Happiness.Rank Happiness.Score  
## 59 Belarus Central and Eastern Europe 59 5.813  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 59 0.03938 1.03192 1.23289 0.73608  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 59 0.37938 0.1909 0.11046 2.1309  
##   
## $Belgium  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 19 Belgium Western Europe 19 6.937 0.03595  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 19 1.30782 1.28566 0.89667 0.5845  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 19 0.2254 0.2225 2.41484  
##   
## $Benin  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 155 Benin Sub-Saharan Africa 155 3.34 0.03656  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 155 0.28665 0.35386 0.3191 0.4845  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 155 0.0801 0.1826 1.63328  
##   
## $Bhutan  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 79 Bhutan Southern Asia 79 5.253 0.03225  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 79 0.77042 1.10395 0.57407 0.53206  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 79 0.15445 0.47998 1.63794  
##   
## $Bolivia  
## Country Region Happiness.Rank Happiness.Score  
## 51 Bolivia Latin America and Caribbean 51 5.89  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 51 0.05642 0.68133 0.97841 0.5392  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 51 0.57414 0.088 0.20536 2.82334  
##   
## $`Bosnia and Herzegovina`  
## Country Region Happiness.Rank  
## 96 Bosnia and Herzegovina Central and Eastern Europe 96  
## Happiness.Score Standard.Error Economy..GDP.per.Capita. Family  
## 96 4.949 0.06913 0.83223 0.91916  
## Health..Life.Expectancy. Freedom Trust..Government.Corruption. Generosity  
## 96 0.79081 0.09245 0.00227 0.24808  
## Dystopia.Residual  
## 96 2.06367  
##   
## $Botswana  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 128 Botswana Sub-Saharan Africa 128 4.332 0.04934  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 128 0.99355 1.10464 0.04776 0.49495  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 128 0.12474 0.10461 1.46181  
##   
## $Brazil  
## Country Region Happiness.Rank Happiness.Score  
## 16 Brazil Latin America and Caribbean 16 6.983  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 16 0.04076 0.98124 1.23287 0.69702  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 16 0.49049 0.17521 0.14574 3.26001  
##   
## $Bulgaria  
## Country Region Happiness.Rank Happiness.Score  
## 134 Bulgaria Central and Eastern Europe 134 4.218  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 134 0.04828 1.01216 1.10614 0.76649  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 134 0.30587 0.00872 0.11921 0.89991  
##   
## $`Burkina Faso`  
## Country Region Happiness.Rank Happiness.Score  
## 152 Burkina Faso Sub-Saharan Africa 152 3.587  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 152 0.04324 0.25812 0.85188 0.27125  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 152 0.39493 0.12832 0.21747 1.46494  
##   
## $Burundi  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 157 Burundi Sub-Saharan Africa 157 2.905 0.08658  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 157 0.0153 0.41587 0.22396 0.1185  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 157 0.10062 0.19727 1.83302  
##   
## $Cambodia  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 145 Cambodia Southeastern Asia 145 3.819 0.05069  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 145 0.46038 0.62736 0.61114 0.66246  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 145 0.07247 0.40359 0.98195  
##   
## $Cameroon  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 133 Cameroon Sub-Saharan Africa 133 4.252 0.04678  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 133 0.4225 0.88767 0.23402 0.49309  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 133 0.05786 0.20618 1.95071  
##   
## $Canada  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 5 Canada North America 5 7.427 0.03553  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 5 1.32629 1.32261 0.90563 0.63297  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 5 0.32957 0.45811 2.45176  
##   
## $`Central African Republic`  
## Country Region Happiness.Rank Happiness.Score  
## 148 Central African Republic Sub-Saharan Africa 148 3.678  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 148 0.06112 0.0785 0 0.06699  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 148 0.48879 0.08289 0.23835 2.7223  
##   
## $Chad  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 149 Chad Sub-Saharan Africa 149 3.667 0.0383  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 149 0.34193 0.76062 0.1501 0.23501  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 149 0.05269 0.18386 1.94296  
##   
## $Chile  
## Country Region Happiness.Rank Happiness.Score  
## 27 Chile Latin America and Caribbean 27 6.67  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 27 0.058 1.10715 1.12447 0.85857  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 27 0.44132 0.12869 0.33363 2.67585  
##   
## $China  
## Country Region Happiness.Rank Happiness.Score Standard.Error  
## 84 China Eastern Asia 84 5.14 0.02424  
## Economy..GDP.per.Capita. Family Health..Life.Expectancy. Freedom  
## 84 0.89012 0.94675 0.81658 0.51697  
## Trust..Government.Corruption. Generosity Dystopia.Residual  
## 84 0.02781 0.08185 1.8604  
##   
## $Colombia  
## Country Region Happiness.Rank Happiness.Score  
## 33 Colombia Latin America and Caribbean 33 6.477  
## Standard.Error Economy..GDP.per.Capita. Family Health..Life.Expectancy.  
## 33 0.05051 0.91861 1.24018 0.69077  
## Freedom Trust..Government.Corruption. Generosity Dystopia.Residual  
## 33 0.53466 0.0512 0.18401 2.85737  
##   
## [ reached getOption("max.print") -- omitted 128 entries ]

# length() function: returns the number of elements in a vector

length(dataset)

## [1] 12

# Counting the number of unique values in a column:

unique\_count <- length(unique(dataset$Country))  
unique\_count

## [1] 158

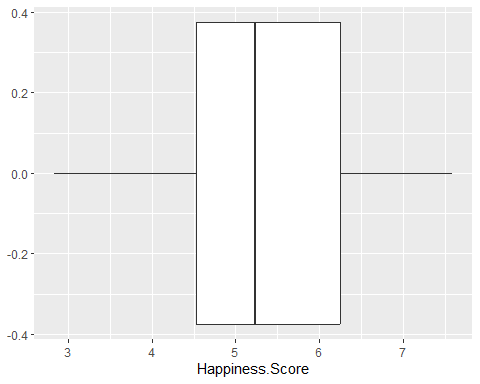
# Calculate the percentage of missing values in each column:

missing\_percentage <- colMeans(is.na(dataset)) \* 100  
missing\_percentage

## Country Region   
## 0 0   
## Happiness.Rank Happiness.Score   
## 0 0   
## Standard.Error Economy..GDP.per.Capita.   
## 0 0   
## Family Health..Life.Expectancy.   
## 0 0   
## Freedom Trust..Government.Corruption.   
## 0 0   
## Generosity Dystopia.Residual   
## 0 0

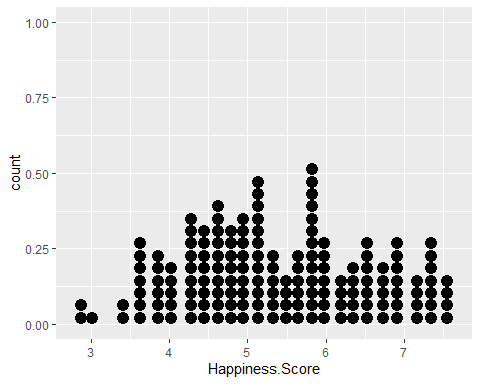
# to draw a boxplot

ggplot(dataset,aes(x=Happiness.Score))+geom\_boxplot()

 # to draw a dot plot

ggplot(dataset,aes(x=Happiness.Score))+geom\_dotplot()

## Bin width defaults to 1/30 of the range of the data. Pick better value with  
## `binwidth`.

 # to draw density graph

ggplot(dataset,aes(x=Happiness.Score))+geom\_density()

