

R Notebook

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

```
date() # Current system time and date.
```

```
## [1] "Fri May 31 13:16:54 2024"
```

```
Sys.time() # Current system time and date (redundant).
```

```
## [1] "2024-05-31 13:16:54 +07"
```

```
R.version.string # R version and version release date.
```

```
## [1] "R version 4.2.3 (2023-03-15 ucrt)"
```

```
options(digits=6) # Confirm default digits.  
options(scipen=999) # Suppress scientific notation.  
options(width=60) # Confirm output width.  
ls() # List all objects in the working # directory.
```

```
## character(0)
```

```
rm(list = ls()) # CAUTION: Remove all files in the #working directory. If this action is not desired, u  
ls.str() # List all objects with finite detail.  
getwd() # Identify the current working directory
```

```
## [1] "C:/Users/linan/Documents/GitHub/project/R-project/using-r-for-biostatistics"
```

```
setwd("C:/Users/linan/Documents/GitHub/project/R-project/using-r-for-biostatistics") # Set to a new wor  
getwd()# Confirm the working directory.
```

```
## [1] "C:/Users/linan/Documents/GitHub/project/R-project/using-r-for-biostatistics"
```

```
list.files()# List files at the PC directory
```

```

## [1] "assignment.csv"
## [2] "BreedMilkLb365.txt"
## [3] "Chapter-1-Import Files.Rmd"
## [4] "Chapter-2-exploration-visualization.Rmd"
## [5] "Chapter-3-distribution-correlation.nb.html"
## [6] "Chapter-3-distribution-correlation.Rmd"
## [7] "Chapter-3-distribution-correlation_files"
## [8] "datasets"
## [9] "EmpWellAge21to79.csv"
## [10] "GenderEndurance.csv"
## [11] "nhanes_2015_2016.csv"
## [12] "Rplot-Frequency of variables.png"
## [13] "Rplot - Distribution of factor-type object variables gender, RaceEthnicity, BMIStatus, and obe
## [14] "Rplot - Distribution of numeric-type object variable age.png"
## [15] "SmartWayVehicleListMY2008.csv"
## [16] "Sorghum2012to2016.xlsx"
## [17] "YearSoilTypeCropRainYieldBushelsPerAcreNoHeader.txt"

.libPaths()# Library pathname

## [1] "C:/Users/linan/AppData/Local/R/win-library/4.2"
## [2] "C:/Program Files/R/R-4.2.3/library"

.Library# Library pathname.

## [1] "C:/PROGRA~1/R/R-42~1.3/library"

sessionInfo() # R version, locale, and packages.

## R version 4.2.3 (2023-03-15 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 22631)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_Indonesia.utf8
## [2] LC_CTYPE=English_Indonesia.utf8
## [3] LC_MONETARY=English_Indonesia.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_Indonesia.utf8
##
## attached base packages:
## [1] stats      graphics   grDevices  utils      datasets
## [6] methods    base
##
## loaded via a namespace (and not attached):
## [1] compiler_4.2.3    fastmap_1.1.1    cli_3.6.1
## [4] tools_4.2.3       htmltools_0.5.8   rstudioapi_0.16.0
## [7] yaml_2.3.8        rmarkdown_2.26    knitr_1.45
## [10] xfun_0.40       digest_0.6.31    rlang_1.1.1
## [13] evaluate_0.23

```

```
search() # Attached packages and objects.
```

```
## [1] ".GlobalEnv"      "package:stats"  
## [3] "package:graphics" "package:grDevices"  
## [5] "package:utils"     "package:datasets"  
## [7] "package:methods"   "Autoloads"  
## [9] "package:base"
```

```
searchpaths() # Attached packages and objects.
```

```
## [1] ".GlobalEnv"  
## [2] "C:/Program Files/R/R-4.2.3/library/stats"  
## [3] "C:/Program Files/R/R-4.2.3/library/graphics"  
## [4] "C:/Program Files/R/R-4.2.3/library/grDevices"  
## [5] "C:/Program Files/R/R-4.2.3/library/utils"  
## [6] "C:/Program Files/R/R-4.2.3/library/datasets"  
## [7] "C:/Program Files/R/R-4.2.3/library/methods"  
## [8] "Autoloads"  
## [9] "C:/PROGRA~1/R/R-42~1.3/library/base"
```

```
#####
```

```
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(Hmisc)
```

```
##  
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:dplyr':  
##  
##     src, summarize
```

```
## The following objects are masked from 'package:base':  
##  
##     format.pval, units
```

```

library(gmodels)
library(ggplot2)
library(tidyverse)

## -- Attaching core tidyverse packages ---- tidyverse 2.0.0 --
## vforcats    1.0.0      vstringr    1.5.1
## vlubridate  1.9.3      vtibble     3.2.1
## vpurrr      1.0.2      vtidyrm    1.3.1
## vreadr      2.1.5

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter()    masks stats::filter()
## x dplyr::lag()       masks stats::lag()
## x Hmisc::src()       masks dplyr::src()
## x Hmisc::summarize() masks dplyr::summarize()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(beeswarm)

setwd("C:/Users/linan/Documents/GitHub/project/R-project/using-r-for-biostatistics")

EmployeeBiometric.df <- read.table(file="EmpWellAge21to79.csv", header=TRUE, dec=".\"", sep = ",") 

str(EmployeeBiometric.df)

## 'data.frame': 3945 obs. of 10 variables:
## $ EmployeeID : chr "ID10005" "ID10010" "ID10016" "ID10028" ...
## $ Gender      : int 1 1 1 2 1 1 1 2 1 1 ...
## $ AgeYears    : int 34 67 41 57 35 40 40 34 40 50 ...
## $ RaceEthnicity: int 5 5 5 1 5 3 2 5 2 2 ...
## $ TotalCholesterolmgdL: int 221 175 224 217 139 198 NA 205 87 149 ...
## $ SBPmmHg     : int 102 NA NA 90 86 114 NA 92 120 126 ...
## $ DBPmmHg     : int 44 NA NA 42 42 74 NA 48 80 72 ...
## $ BMIMetric   : num 18.4 18.7 21.6 20.6 18.7 ...
## $ BMISstatus  : int 1 2 2 2 2 2 2 1 2 ...
## $ Obesity     : int 1 1 1 1 1 1 1 1 1 1 ...

dim(EmployeeBiometric.df)

## [1] 3945 10

head(EmployeeBiometric.df)

##   EmployeeID Gender AgeYears RaceEthnicity
## 1 ID10005     1      34          5
## 2 ID10010     1      67          5
## 3 ID10016     1      41          5
## 4 ID10028     2      57          1
## 5 ID10041     1      35          5

```

```

## 6     ID10051      1      40      3
##   TotalCholesterolmgdL SBPmmHg DBPmmHg BMIMetric BMIStatus
## 1             221       102      44    18.35        1
## 2             175        NA       NA    18.66        2
## 3             224        NA       NA    21.61        2
## 4             217        90      42    20.59        2
## 5             139        86      42    18.66        2
## 6             198       114      74    19.73        2
##   Obesity
## 1     1
## 2     1
## 3     1
## 4     1
## 5     1
## 6     1

```

Check for duplicate in individual level = no duplicate ID

```
table(duplicated(EmployeeBiometric.df$EmployeeID))
```

```

##
## FALSE
## 3945

```

```
summary(EmployeeBiometric.df)
```

```

##   EmployeeID          Gender      AgeYears
##   Length:3945      Min. :1.00    Min. :21.0
##   Class :character  1st Qu.:1.00   1st Qu.:34.0
##   Mode  :character  Median :1.00   Median :47.0
##                   Mean   :1.47   Mean   :47.7
##                   3rd Qu.:2.00  3rd Qu.:61.0
##                   Max.  :2.00   Max.  :79.0
##
##   RaceEthnicity TotalCholesterolmgdL      SBPmmHg
##   Min.   :1.0   Min.   : 78      Min.   : 74
##   1st Qu.:2.0  1st Qu.:151     1st Qu.:108
##   Median :3.0  Median :175     Median :118
##   Mean   :3.4  Mean   :180     Mean   :122
##   3rd Qu.:5.0  3rd Qu.:205     3rd Qu.:132
##   Max.   :5.0  Max.   :380     Max.   :222
##   NA's   :894   NA's   :209
##
##   DBPmmHg          BMIMetric      BMIStatus
##   Min.   :34.0   Min.   :13.4   Min.   :1.00
##   1st Qu.:60.0   1st Qu.:22.6   1st Qu.:2.00
##   Median :70.0   Median :26.4   Median :3.00
##   Mean   :69.3   Mean   :27.7   Mean   :2.86
##   3rd Qu.:78.0   3rd Qu.:31.3   3rd Qu.:4.00
##   Max.   :122.0  Max.   :70.1   Max.   :4.00
##   NA's   :217    NA's   :20    NA's   :20
##
##   Obesity
##   Min.   :1.00

```

```

## 1st Qu.:1.00
## Median :1.00
## Mean   :1.31
## 3rd Qu.:2.00
## Max.   :2.00
## NA's    :20

Change EmployeeID,Gender, Ethnic, and BMIStatus from int to factor

EmployeeBiometric.df$EmployeeID <- as.factor(EmployeeBiometric.df$EmployeeID)

EmployeeBiometric.df$Gender <- factor(EmployeeBiometric.df$Gender,
                                         labels=c("Female","Male"))
levels(EmployeeBiometric.df$Gender)

## [1] "Female" "Male"

EmployeeBiometric.df$BMIStatus<-factor(EmployeeBiometric.df$BMIStatus,
                                         labels = c("Underweight","Normal Weight", "Overweight","Obese"))
levels(EmployeeBiometric.df$BMIStatus)

## [1] "Underweight"    "Normal Weight"   "Overweight"
## [4] "Obese"

EmployeeBiometric.df$RaceEthnicity <-
  factor(EmployeeBiometric.df$RaceEthnicity,
         labels=c("Asian","Black","Hispanic","Other","White"))
levels(EmployeeBiometric.df$RaceEthnicity)

## [1] "Asian"      "Black"      "Hispanic"   "Other"      "White"

EmployeeBiometric.df$Obesity <-
  factor(EmployeeBiometric.df$Obesity,
         labels=c("Not Obese","Obese"))
levels(EmployeeBiometric.df$Obesity)

## [1] "Not Obese" "Obese"

```

Change AgeYears, SBPmmHg, DBPmmHg from int to numeric

```

EmployeeBiometric.df$AgeYears <- as.numeric(EmployeeBiometric.df$AgeYears)
EmployeeBiometric.df$SBPmmHg <- as.numeric(EmployeeBiometric.df$SBPmmHg)
EmployeeBiometric.df$DBPmmHg <- as.numeric(EmployeeBiometric.df$DBPmmHg)

```

Check data structure

```
str(EmployeeBiometric.df)
```

```

## 'data.frame':   3945 obs. of  10 variables:
## $ EmployeeID      : Factor w/ 3945 levels "ID10005","ID10010",...: 1 2 3 4 5 6 7 8 9 10 ...
## $ Gender          : Factor w/ 2 levels "Female","Male": 1 1 1 2 1 1 1 2 1 1 ...
## $ AgeYears         : num  34 67 41 57 35 40 40 34 40 50 ...
## $ RaceEthnicity    : Factor w/ 5 levels "Asian","Black",...: 5 5 5 1 5 3 2 5 2 2 ...
## $ TotalCholesterolmgdL: int  221 175 224 217 139 198 NA 205 87 149 ...
## $ SBPmmHg          : num  102 NA NA 90 86 114 NA 92 120 126 ...
## $ DBPmmHg          : num  44 NA NA 42 42 74 NA 48 80 72 ...
## $ BMIMetric         : num  18.4 18.7 21.6 20.6 18.7 ...
## $ BMISstatus        : Factor w/ 4 levels "Underweight",...: 1 2 2 2 2 2 2 2 1 2 ...
## $ Obesity           : Factor w/ 2 levels "Not Obese","Obese": 1 1 1 1 1 1 1 1 1 1 ...

getwd()

## [1] "C:/Users/linan/Documents/GitHub/project/R-project/using-r-for-biostatistics"

ls()

## [1] "EmployeeBiometric.df"

attach(EmployeeBiometric.df)
dim(EmployeeBiometric.df)

## [1] 3945   10

names(EmployeeBiometric.df)

##  [1] "EmployeeID"          "Gender"
##  [3] "AgeYears"            "RaceEthnicity"
##  [5] "TotalCholesterolmgdL" "SBPmmHg"
##  [7] "DBPmmHg"              "BMIMetric"
##  [9] "BMISstatus"           "Obesity"

colnames(EmployeeBiometric.df)

##  [1] "EmployeeID"          "Gender"
##  [3] "AgeYears"            "RaceEthnicity"
##  [5] "TotalCholesterolmgdL" "SBPmmHg"
##  [7] "DBPmmHg"              "BMIMetric"
##  [9] "BMISstatus"           "Obesity"

head(EmployeeBiometric.df)

##   EmployeeID Gender AgeYears RaceEthnicity
## 1 ID10005 Female     34      White
## 2 ID10010 Female     67      White
## 3 ID10016 Female     41      White
## 4 ID10028 Male       57      Asian
## 5 ID10041 Female     35      White

```

```

## 6 ID10051 Female 40 Hispanic
## TotalCholesterolmgdL SBPmmHg DBPmmHg BMIMetric
## 1 221 102 44 18.35
## 2 175 NA NA 18.66
## 3 224 NA NA 21.61
## 4 217 90 42 20.59
## 5 139 86 42 18.66
## 6 198 114 74 19.73
## BMIStatus Obesity
## 1 Underweight Not Obese
## 2 Normal Weight Not Obese
## 3 Normal Weight Not Obese
## 4 Normal Weight Not Obese
## 5 Normal Weight Not Obese
## 6 Normal Weight Not Obese

```

```
tail(EmployeeBiometric.df)
```

```

## EmployeeID Gender AgeYears RaceEthnicity
## 3940 ID50080 Male 57 White
## 3941 ID50087 Female 44 Asian
## 3942 ID50104 Male 45 Asian
## 3943 ID50108 Female 28 Hispanic
## 3944 ID50109 Female 57 White
## 3945 ID50126 Male 57 White
## TotalCholesterolmgdL SBPmmHg DBPmmHg BMIMetric
## 3940 152 98 62 NA
## 3941 NA 108 64 NA
## 3942 NA 138 74 NA
## 3943 186 156 88 NA
## 3944 139 NA NA NA
## 3945 170 NA NA NA
## BMIStatus Obesity
## 3940 <NA> <NA>
## 3941 <NA> <NA>
## 3942 <NA> <NA>
## 3943 <NA> <NA>
## 3944 <NA> <NA>
## 3945 <NA> <NA>

```

```
summary(EmployeeBiometric.df)
```

```

## EmployeeID Gender AgeYears
## ID10005: 1 Female:2078 Min. :21.0
## ID10010: 1 Male :1867 1st Qu.:34.0
## ID10016: 1 Median :47.0
## ID10028: 1 Mean :47.7
## ID10041: 1 3rd Qu.:61.0
## ID10051: 1 Max. :79.0
## (Other):3939
## RaceEthnicity TotalCholesterolmgdL SBPmmHg
## Asian : 482 Min. : 78 Min. : 74
## Black : 818 1st Qu.:151 1st Qu.:108

```

```

##   Hispanic: 904   Median :175           Median :118
##   Other     : 117   Mean    :180           Mean    :122
##   White     :1624   3rd Qu.:205          3rd Qu.:132
##                           Max.    :380           Max.    :222
##                           NA's    :894           NA's    :209
##   DBPmmHg      BMIMetric          BMIStatus
##   Min.    : 34.0   Min.    :13.4 Underweight  : 168
##   1st Qu.: 60.0   1st Qu.:22.6 Normal Weight:1437
##   Median  : 70.0   Median  :26.4 Overweight   :1114
##   Mean    : 69.3   Mean    :27.7 Obese       :1206
##   3rd Qu.: 78.0   3rd Qu.:31.3 NA's        :  20
##   Max.    :122.0   Max.    :70.1
##   NA's    :217     NA's    : 20
##   Obesity
##   Not Obese:2719
##   Obese     :1206
##   NA's     :  20
##
##
##
##

```

Conduct a Visual Data Check Using Graphics using epiDisplay

The epiDisplay::tab0.85() function provides syntax that generates an attractive barplot with frequency counts placed over each bar and a frequency distribution table printed to the screen. Also, for factor-type object variables with missing data, the epiDisplay::tab0.85() function incorporates the presentation of these missing data into the barplot and the frequency distribution table, providing a complete sense of the data

Create bar plots for factor variables : Gender, RaceEthnicity, BMISstatus, Obesity

```

par(ask=TRUE)
par(mfrow=c(2,2))
epiDisplay::tab1(EmployeeBiometric.df$Gender,
                 main = "Frequency of Gender",
                 ylab = "Frequency",
                 graph = TRUE,
                 missing=TRUE,
                 bar.values = c("frequency"),
                 horiz = TRUE,
                 cex = 0.8,
                 cex.names = 0.8,
                 cex.lab=0.8,
                 cex.axis=0.8,
                 col = c("black","red"))

```

```

## EmployeeBiometric.df$Gender :
##   Frequency Percent Cum. percent
## Female      2078    52.7      52.7
## Male        1867    47.3     100.0
## Total       3945   100.0     100.0

```

```

epiDisplay::tab1(EmployeeBiometric.df$RaceEthnicity,
                 main = "Frequency of RaceEthnicity",

```

```

ylab = "Frequency",
graph = TRUE,
missing=TRUE,
bar.values = c("frequency"),
horiz = TRUE,
cex = 0.8,
cex.names = 0.8,
cex.lab=0.8,
cex.axis=0.8,
col = c("black","red","blue","green","cyan"))

```

```

## EmployeeBiometric.df$RaceEthnicity :
##          Frequency Percent Cum. percent
## Asian           482    12.2      12.2
## Black           818    20.7      33.0
## Hispanic        904    22.9      55.9
## Other            117     3.0      58.8
## White          1624    41.2     100.0
## Total           3945   100.0     100.0

```

```

epiDisplay::tab1(EmployeeBiometric.df$BMIStatus,
  main = "Frequency of BMIStatus",
  ylab = "Frequency",
  graph = TRUE,
  missing=TRUE,
  bar.values = c("frequency"),
  horiz = TRUE,
  cex = 0.85,
  cex.names = 0.85,
  cex.lab=0.85,
  cex.axis=0.85,
  col = c("black","red","blue","green","cyan"))

```

```

## EmployeeBiometric.df$BMIStatus :
##          Frequency  %(NA+)  %(NA-)
## Underweight       168      4.3      4.3
## Normal Weight    1437     36.4     36.6
## Overweight       1114     28.2     28.4
## Obese            1206     30.6     30.7
## NA's              20       0.5      0.0
## Total             3945    100.0    100.0

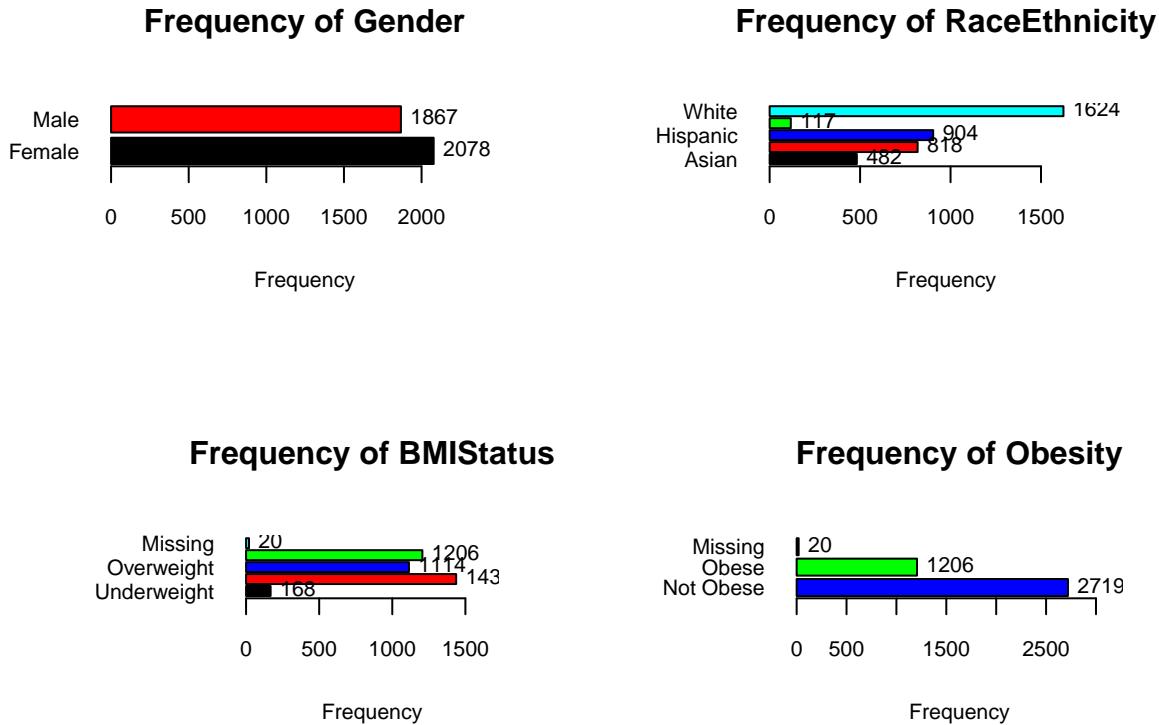
```

```

epiDisplay::tab1(EmployeeBiometric.df$Obesity,
  main = "Frequency of Obesity",
  ylab = "Frequency",
  graph = TRUE,
  missing=TRUE,
  bar.values = c("frequency"),
  horiz = TRUE,
  cex = 0.85,
  cex.names = 0.85,
  cex.lab=0.85,

```

```
cex.axis=0.85,
col = c("blue","green","red"))
```



```
## EmployeeBiometric.df$Obesity :
##          Frequency  %(NA+)  %(NA-)
## Not Obese      2719     68.9     69.3
## Obese         1206     30.6     30.7
## NA's           20      0.5      0.0
## Total         3945    100.0    100.0
```

Create bar plots for numeric variables : AgeYears, TotalCholesterolmgdL, SBPmmHg, DBPmmHg, and the calculated value for BMIMetric.

Figures for the EmployeeBiometric.df object variables with numeric values (e.g., often called interval values or scale values) will be displayed in a 2 by 2 graphical grid. Each square in the grid uses a different R-based function to display data distribution:

- Histogram — Use the hist() function.
- Density Curve — The plot() function is wrapped around the density() function.
- Beanplot — Use the beanplot::beanplot() function, by loading the external package beanplot into the active R session.
- Quantile-Quantile Curve — Use the qqnorm() function.

Variable AgeYears

```
par(ask=TRUE)
par(mfrow=c(2,2))
```

```

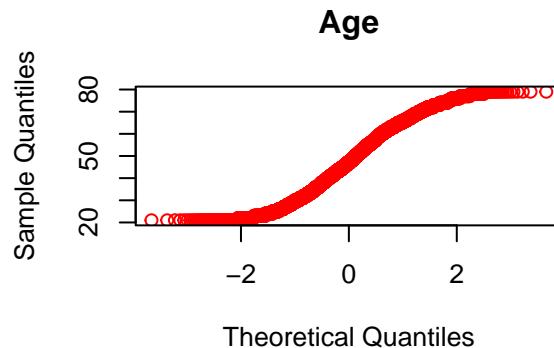
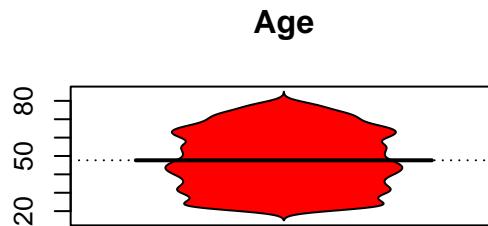
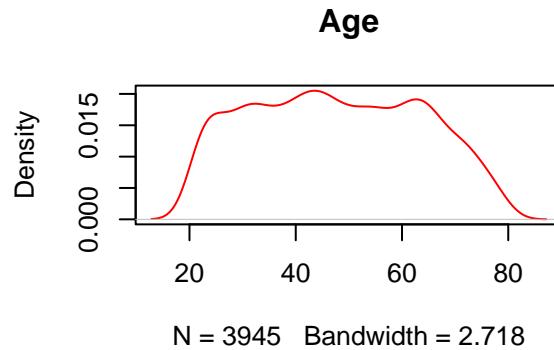
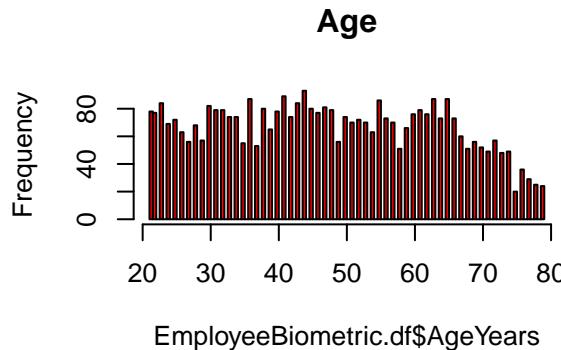
hist(EmployeeBiometric.df$AgeYears, main="Age", col="red", breaks=100)

plot(density(EmployeeBiometric.df$AgeYears, na.rm=TRUE),
     main="Age", col="red")

beanplot:::beanplot(EmployeeBiometric.df$AgeYears, main="Age", col="red", what=c(0.85,0.85,0.85,0), overall=TRUE)

qqnorm(EmployeeBiometric.df$AgeYears, main="Age", col="red")

```



Variable TotalCholesterolmgL

```

par(ask=TRUE)
par(mfrow=c(2,2))

hist(EmployeeBiometric.df$TotalCholesterolmgdL, main="Total Cholesterol mg/dL", col="blue", breaks=100)

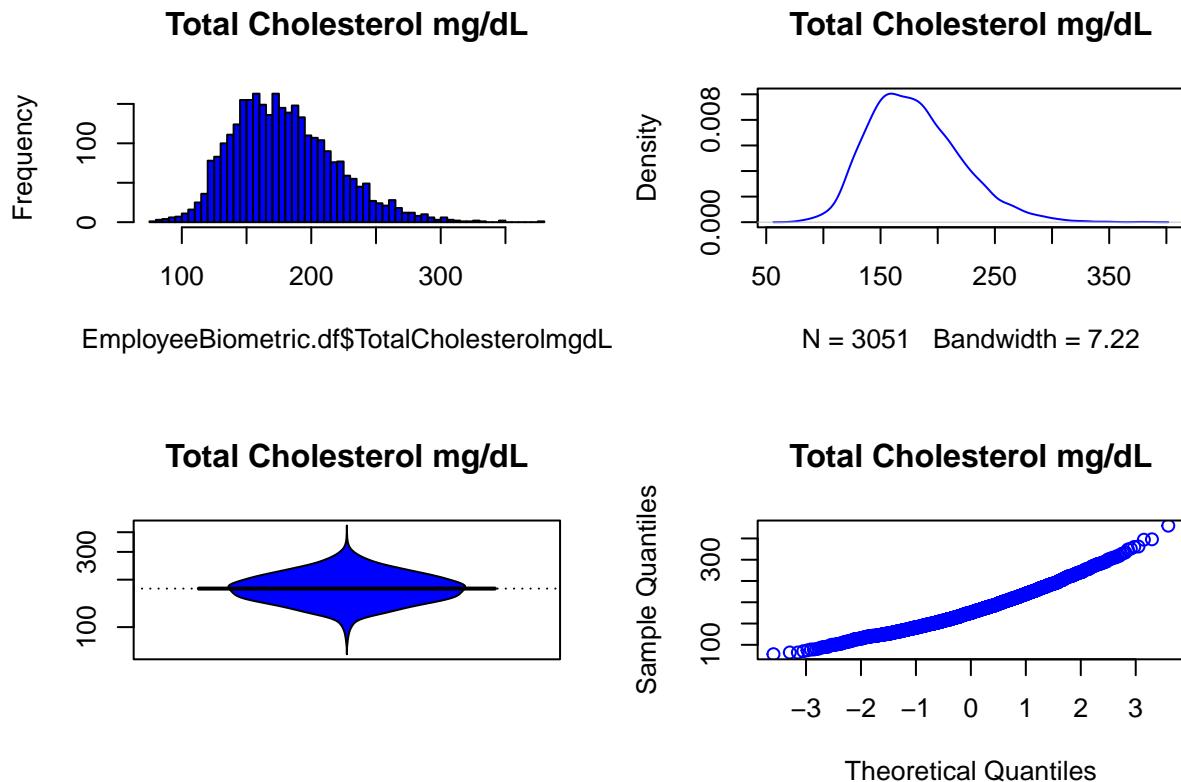
plot(density(EmployeeBiometric.df$TotalCholesterolmgdL, na.rm=TRUE),
     main="Total Cholesterol mg/dL", col="blue")

beanplot:::beanplot(EmployeeBiometric.df$TotalCholesterolmgdL, main="Total Cholesterol mg/dL", col="blue")

## log="y" selected

```

```
qqnorm(EmployeeBiometric.df$TotalCholesterolmgdL, main="Total Cholesterol mg/dL", col="blue")
```



Variable SBPmmHg

```
par(ask=TRUE)
par(mfrow=c(2,2))

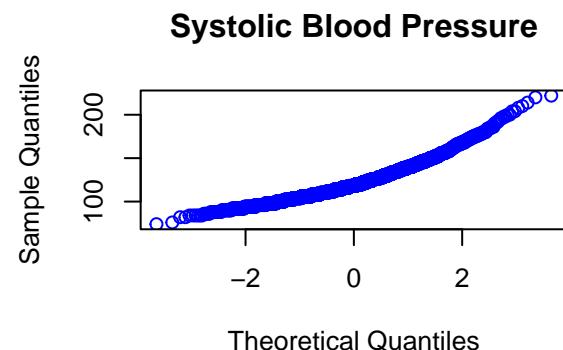
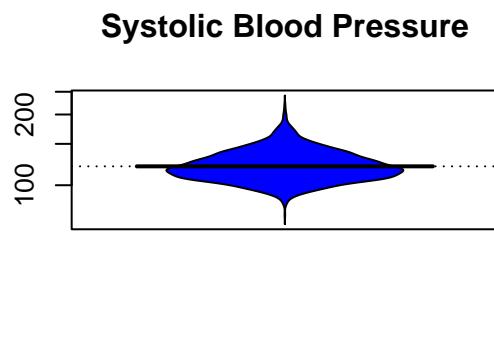
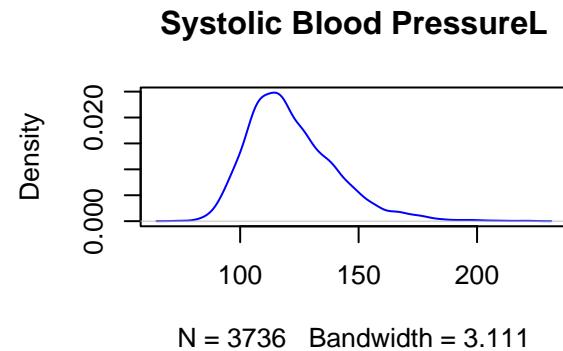
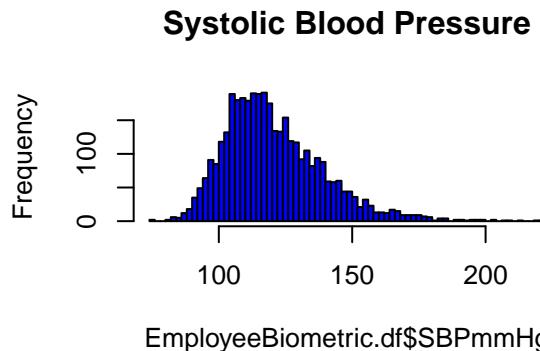
hist(EmployeeBiometric.df$SBPmmHg,
      main="Systolic Blood Pressure",
      col="blue",
      breaks=100)

plot(density(EmployeeBiometric.df$SBPmmHg,
             na.rm=TRUE),
      main="Systolic Blood PressureL",
      col="blue")

beanplot:::beanplot(EmployeeBiometric.df$SBPmmHg,
                     main="Systolic Blood Pressure",
                     col="blue",
                     what=c(1,1,1,0),
                     overallline="mean",
                     boxwex=0.75,
                     horizontal=FALSE)

## log="y" selected
```

```
qqnorm(EmployeeBiometric.df$SBPmmHg,
       main="Systolic Blood Pressure",
       col="blue")
```



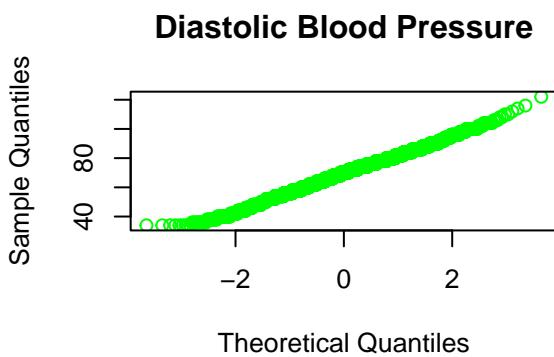
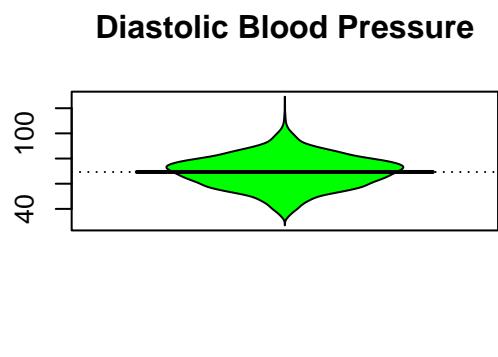
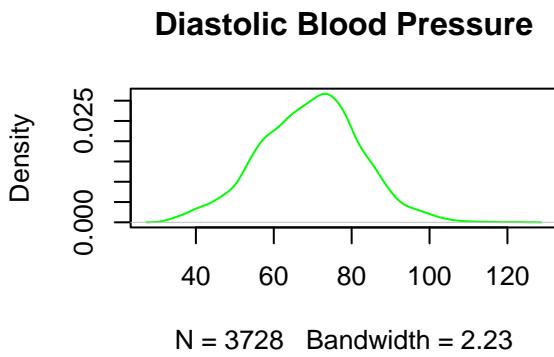
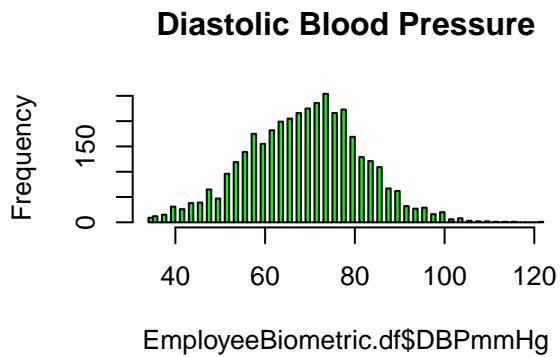
Variable DBPmmHg

```
par(ask=TRUE)
par(mfrow=c(2,2))

hist(EmployeeBiometric.df$DBPmmHg, main="Diastolic Blood Pressure", col="green", breaks=100)

plot(density(EmployeeBiometric.df$DBPmmHg, na.rm=TRUE),
      main="Diastolic Blood Pressure", col="green")

beanplot:::beanplot(EmployeeBiometric.df$DBPmmHg, main="Diastolic Blood Pressure", col="green",
                     what=c(0.85,0.85,0.85,0), overallline="mean", boxwex=0.75, horizontal=FALSE)
qqnorm(EmployeeBiometric.df$DBPmmHg, main="Diastolic Blood Pressure", col="green")
```



Variable BMIMetric

```
par(ask=TRUE)
par(mfrow=c(2,2))

hist(EmployeeBiometric.df$BMIMetric, main="Body Mass Index: BMI Metric = kg/m^2", col="purple", breaks=20)

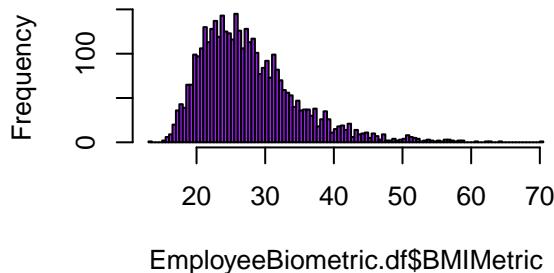
plot(density(EmployeeBiometric.df$BMIMetric, na.rm=TRUE),
     main="Body Mass Index: BMI Metric = kg/m^2", col="purple")

beanplot::beanplot(EmployeeBiometric.df$BMIMetric, main="Body Mass Index: BMI Metric = kg/m^2", col="purple")

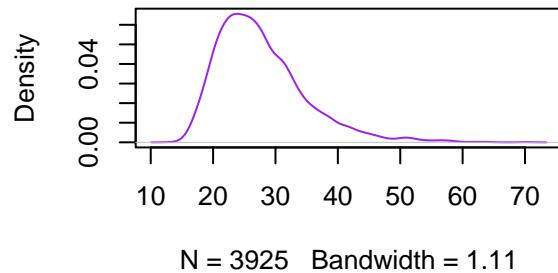
## log="y" selected

qqnorm(EmployeeBiometric.df$BMIMetric, main="Body Mass Index: BMI Metric = kg/m^2", col="purple")
```

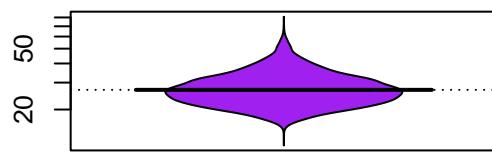
Body Mass Index: BMI Metric = kg/m²



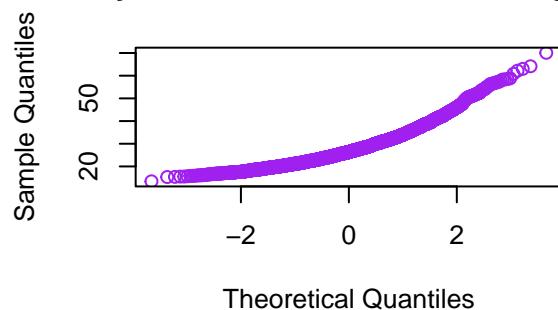
Body Mass Index: BMI Metric = kg/m²



Body Mass Index: BMI Metric = kg/m²



Body Mass Index: BMI Metric = kg/m²



Summary for variable factors

```
summary(EmployeeBiometric.df[, c(2, 4, 9, 10)])
```

```
##      Gender      RaceEthnicity      BMIStatus
##  Female:2078    Asian : 482 Underweight : 168
##  Male  :1867     Black : 818 Normal Weight:1437
## 
## Hispanic: 904   Overweight  :1114
## Other   : 117    Obese       :1206
## White   :1624    NA's        :  20
## 
##          Obesity
## Not Obese:2719
## Obese     :1206
## NA's      :  20
## 
```

Summary for numeric variable

```
summary(EmployeeBiometric.df[, c(3, 5, 7, 8)])
```

```
##      AgeYears   TotalCholesterolmgdL    DBPmmHg
##  Min.   :21.0    Min.   : 78           Min.   : 34.0
##  1st Qu.:34.0    1st Qu.:151          1st Qu.: 60.0
##  Median :47.0    Median :175          Median : 70.0
```

```

##   Mean    :47.7    Mean    :180        Mean    : 69.3
##  3rd Qu.:61.0    3rd Qu.:205        3rd Qu.: 78.0
##  Max.    :79.0    Max.    :380        Max.    :122.0
##                NA's    :894        NA's    :217
##   BMIMetric
##   Min.    :13.4
##   1st Qu.:22.6
##   Median :26.4
##   Mean    :27.7
##   3rd Qu.:31.3
##   Max.    :70.1
##   NA's    :20

```

Provide percentage of factor variable

```
epiDisplay::tab1(EmployeeBiometric.df$Gender, graph=FALSE)
```

```

## EmployeeBiometric.df$Gender :
##           Frequency Percent Cum. percent
## Female      2078     52.7       52.7
## Male        1867     47.3      100.0
## Total       3945    100.0      100.0

```

```
epiDisplay::tab1(EmployeeBiometric.df$RaceEthnicity, graph = FALSE)
```

```

## EmployeeBiometric.df$RaceEthnicity :
##           Frequency Percent Cum. percent
## Asian        482     12.2      12.2
## Black        818     20.7      33.0
## Hispanic     904     22.9      55.9
## Other         117      3.0      58.8
## White        1624     41.2      100.0
## Total       3945    100.0      100.0

```

```
epiDisplay::tab1(EmployeeBiometric.df$BMIStatus, graph=FALSE)
```

```

## EmployeeBiometric.df$BMIStatus :
##           Frequency  %(NA+)  %(NA-)
## Underweight      168      4.3      4.3
## Normal Weight   1437     36.4     36.6
## Overweight      1114     28.2     28.4
## Obese           1206     30.6     30.7
## NA's            20       0.5      0.0
## Total          3945    100.0    100.0

```

```
epiDisplay::tab1(EmployeeBiometric.df$Obesity, graph = FALSE)
```

```

## EmployeeBiometric.df$Obesity :
##           Frequency  %(NA+)  %(NA-)
## Not Obese       2719     68.9     69.3
## Obese          1206     30.6     30.7
## NA's           20       0.5      0.0
## Total          3945    100.0    100.0

```

Measures of Central Tendency for Numeric-Type Object Variables To address this concern and the need for attention to Standard Deviation as a vital descriptive statistic, consider use of the RcmdrMisc::numSummary() function, which by default provides Mean, Standard Deviation, IQR (Interquartile Range, or Q3 minus Q1) Quartiles (0% (Minimum), 25%, 50% (Median), 75%, 100% (Maximum)) N, and NAs (the number of missing values, if any).

```
RcmdrMisc::numSummary(EmployeeBiometric.df$AgeYears)
```

```
##      mean      sd  IQR 0% 25% 50% 75% 100%     n
##  47.6563 15.8221 27 21  34  47  61   79 3945
```

```
RcmdrMisc::numSummary(EmployeeBiometric.df$TotalCholesterolmgdL)
```

```
##      mean      sd  IQR 0% 25% 50% 75% 100%     n    NA
## 179.835 39.9184 54 78 151 175 205 380 3051 894
```

```
RcmdrMisc::numSummary(EmployeeBiometric.df$SBPmmHg)
```

```
##      mean      sd  IQR 0% 25% 50% 75% 100%     n    NA
## 121.646 18.6069 24 74 108 118 132 222 3736 209
```

```
RcmdrMisc::numSummary(EmployeeBiometric.df$DBPmmHg)
```

```
##      mean      sd  IQR 0% 25% 50% 75% 100%     n    NA
## 69.3192 12.836 18 34  60  70  78 122 3728 217
```

```
RcmdrMisc::numSummary(EmployeeBiometric.df$BMIMetric)
```

```
##      mean      sd  IQR 0% 25% 50% 75% 100%     n    NA
## 27.7057 7.12103 8.65 13.45 22.61 26.38 31.26 70.08 3925 20
```

Summary, group by gender

```
RcmdrMisc::numSummary(EmployeeBiometric.df[,c("AgeYears", "TotalCholesterolmgdL", "SBPmmHg", "DBPmmHg", "BM
```

```
##
## Variable: AgeYears
##      mean      sd  IQR 0% 25% 50% 75% 100%     n    NA
## Female 47.6925 15.7969 27 21  34  47  61   79 2078  0
## Male   47.6160 15.8542 27 21  34  47  61   79 1867  0
##
## Variable: TotalCholesterolmgdL
##      mean      sd  IQR 0% 25% 50% 75% 100%     n    NA
## Female 179.768 40.5027 56.0 78 150 176 206.0 380 1628 450
## Male   179.911 39.2533 52.5 85 151 175 203.5 348 1423 444
##
## Variable: SBPmmHg
##      mean      sd  IQR 0% 25% 50% 75% 100%     n    NA
## Female 119.390 18.0147 20 82 108 116 128 222 1976 102
```

```

## Male    124.178 18.9374  26 74 110 122 136  220 1760 107
##
## Variable: DBPmmHg
##      mean      sd IQR 0% 25% 50% 75% 100%   n  NA
## Female 68.2222 12.604 16 34 60 68 76 122 1971 107
## Male   70.5498 12.985 18 34 62 72 80 114 1757 110
##
## Variable: BMIMetric
##      mean      sd IQR 0% 25% 50% 75% 100%
## Female 27.6231 7.17061 8.745 15.41 22.435 26.18 31.18 64.15
## Male   27.7976 7.06625 8.540 13.45 22.750 26.56 31.29 70.08
##      n NA
## Female 2067 11
## Male   1858 9

```

Summary, group by RaceEthnicity

```
RcmdrMisc::numSummary(EmployeeBiometric.df[,c("AgeYears", "TotalCholesterolmgdL", "SBPmmHg", "DBPmmHg", "BM
```

```

##
## Variable: AgeYears
##      mean      sd IQR 0% 25% 50% 75% 100%   n  NA
## Asian   47.0353 14.7131 24 21 35 46 59 79 482 0
## Black   48.2433 16.1025 28 21 34 49 62 79 818 0
## Hispanic 46.8584 15.3455 26 21 35 46 61 79 904 0
## Other   38.6068 13.3177 20 21 28 36 48 78 117 0
## White   48.6410 16.2003 27 21 35 48 62 79 1624 0
##
## Variable: TotalCholesterolmgdL
##      mean      sd IQR 0% 25% 50% 75% 100%
## Asian   178.809 40.5190 52.00 95 150 175.0 202.00 324
## Black   181.892 40.7727 54.75 78 152 177.0 206.75 380
## Hispanic 178.557 38.1856 52.50 82 150 174.0 202.50 309
## Other   176.864 38.7806 59.00 109 147 171.5 206.00 292
## White   179.979 40.3126 54.00 82 151 176.0 205.00 347
##      n  NA
## Asian   362 120
## Black   646 172
## Hispanic 687 217
## Other   88 29
## White   1268 356
##
## Variable: SBPmmHg
##      mean      sd IQR 0% 25% 50% 75% 100%   n  NA
## Asian   119.062 18.0030 24 82 106 116 130 222 454 28
## Black   125.683 18.4081 24 90 112 122 136 200 769 49
## Hispanic 122.948 18.4694 22 88 110 120 132 214 863 41
## Other   122.000 18.0343 25 90 108 118 133 198 111 6
## White   119.635 18.6154 24 74 106 116 130 220 1539 85
##
## Variable: DBPmmHg
##      mean      sd IQR 0% 25% 50% 75% 100%   n  NA
## Asian   67.6380 12.6124 18 38 58 68 76 104 453 29

```

```

## Black    72.5312 12.9211  16 34  64  74  80 112 768 50
## Hispanic 70.5128 13.0697  18 34  62  72  80 122 862 42
## Other    68.8649 12.2121  16 34  62  70  78 100 111  6
## White    67.5698 12.3939  18 34  58  68  76 116 1534 90
##
## Variable: BMIMetric
##          mean      sd   IQR  0%  25%  50%
## Asian     27.9289 7.45770 10.0150 13.45 22.2075 26.325
## Black     27.4193 6.73504  8.0600 15.55 22.8300 26.250
## Hispanic  27.6545 6.99525  8.6675 15.53 22.6600 26.375
## Other     28.5462 6.87024  7.7500 17.57 23.8900 26.690
## White     27.7517 7.29454  8.6200 15.34 22.5100 26.490
##          75% 100% n NA
## Asian     32.2225 58.34  476  6
## Black     30.8900 64.15  813  5
## Hispanic  31.3275 62.19  902  2
## Other     31.6400 51.24  117  0
## White     31.1300 70.08 1617  7

```

Summary, group by BMIstatus

```
RcmdrMisc::numSummary(EmployeeBiometric.df[,c("AgeYears", "TotalCholesterolmgdL", "SBPmmHg", "DBPmmHg", "BM
```

```

##
## Variable: AgeYears
##          mean      sd   IQR  0%  25%  50%  75% 100%
## Underweight 45.9583 15.3289 27.25 21  32  46 59.25  77
## Normal Weight 47.7537 15.8668 27.00 21  34  47 61.00  79
## Overweight   47.9282 15.7987 26.75 21  35  48 61.75  79
## Obese        47.4677 15.8322 26.75 21  34  47 60.75  79
##          n NA
## Underweight   168  0
## Normal Weight 1437  0
## Overweight    1114  0
## Obese         1206  0
##
## Variable: TotalCholesterolmgdL
##          mean      sd   IQR  0%  25%  50%  75% 100%
## Underweight 179.354 39.8540 48 87 154.5 173.0 202.5 326
## Normal Weight 179.461 41.0036 54 78 150.0 174.0 204.0 380
## Overweight   181.183 40.0782 56 82 151.0 177.0 207.0 347
## Obese        179.254 38.6444 52 82 151.0 176.5 203.0 324
##          n NA
## Underweight   127  41
## Normal Weight 1088 349
## Overweight    871 243
## Obese         950 256
##
## Variable: SBPmmHg
##          mean      sd   IQR  0%  25%  50%  75% 100% n
## Underweight 113.045 18.6707 16 82 102 110 118 208 157
## Normal Weight 116.622 18.1826 20 76 104 114 124 220 1363
## Overweight   123.802 18.5569 24 74 110 120 134 222 1058

```

```

## Obese      126.915 17.1579  24 90 114 126 138  204 1145
##          NA
## Underweight 11
## Normal Weight 74
## Overweight   56
## Obese        61
##
## Variable: DBPmmHg
##           mean      sd IQR 0% 25% 50% 75% 100%    n
## Underweight 63.3758 12.8100 18 36 54 64 72 116 157
## Normal Weight 65.9750 12.3119 16 34 58 66 74 114 1358
## Overweight   70.6199 12.5282 16 34 62 72 78 122 1055
## Obese        72.9817 12.4347 16 34 64 74 80 108 1145
##          NA
## Underweight 11
## Normal Weight 79
## Overweight   59
## Obese        61
##
## Variable: BMIMetric
##           mean      sd IQR 0% 25% 50%
## Underweight 17.4923 0.824308 1.0325 13.45 17.0900 17.58
## Normal Weight 22.1205 1.747179 2.9300 18.60 20.7200 22.28
## Overweight   27.3041 1.395345 2.3650 25.00 26.0725 27.24
## Obese        36.1543 6.022411 6.9200 30.01 31.7400 34.19
##           75% 100%    n NA
## Underweight 18.1225 18.59 168 0
## Normal Weight 23.6500 25.00 1437 0
## Overweight   28.4375 30.00 1114 0
## Obese        38.6600 70.08 1206 0

```

Summary, group by Obesity

```
RcmdrMisc::numSummary(EmployeeBiometric.df[,c("AgeYears", "TotalCholesterolmgdL", "SBPmmHg", "DBPmmHg", "BM
```

```

##
## Variable: AgeYears
##           mean      sd IQR 0% 25% 50% 75% 100%    n
## Not Obese 47.7142 15.8071 27.00 21 34 47 61.00    79 2719
## Obese     47.4677 15.8322 26.75 21 34 47 60.75    79 1206
##          NA
## Not Obese 0
## Obese     0
##
## Variable: TotalCholesterolmgdL
##           mean      sd IQR 0% 25% 50% 75% 100%    n
## Not Obese 180.174 40.5399 54 78 151 175.0 205 380 2086
## Obese     179.254 38.6444 52 82 151 176.5 203 324 950
##          NA
## Not Obese 633
## Obese     256
##
## Variable: SBPmmHg
```

```

##               mean      sd IQR 0% 25% 50% 75% 100%     n  NA
## Not Obese  119.351 18.7503 22 74 106 116 128  222 2578 141
## Obese      126.915 17.1579 24 90 114 126 138  204 1145  61
##
## Variable: DBPmmHg
##               mean      sd IQR 0% 25% 50% 75% 100%     n  NA
## Not Obese  67.7230 12.6746 18 34 58 68 76 122 2570 149
## Obese      72.9817 12.4347 16 34 64 74 80 108 1145  61
##
## Variable: BMIMetric
##               mean      sd IQR 0% 25% 50% 75%
## Not Obese 23.9583 3.37783 5.38 13.45 21.365 24.01 26.745
## Obese     36.1543 6.02241 6.92 30.01 31.740 34.19 38.660
##               100%     n  NA
## Not Obese 30.00 2719  0
## Obese     70.08 1206  0

```

To give another view of the descriptive statistics gained from use of the summary() function and the RcmdrMisc::numSummary() function, notice below how these outcomes are presented in graphical format, applying a beeswarm plot over a boxplot.

1. Distribution of age based on categorical variables :

```

par(ask=TRUE)
par(mfrow=c(2,2))

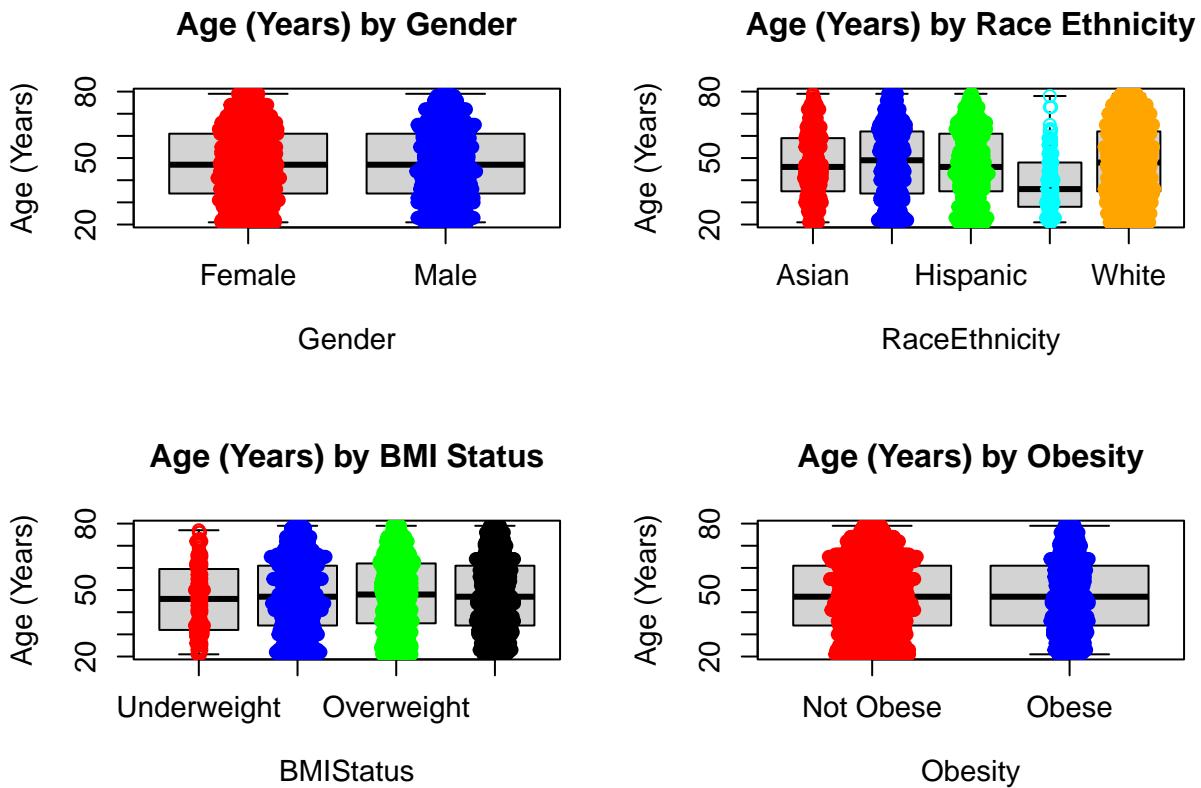
boxplot(AgeYears ~ Gender, data=EmployeeBiometric.df,
        main="Age (Years) by Gender", ylab="Age (Years)",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(AgeYears~Gender, data=EmployeeBiometric.df,
         col=c("red","blue"),add=TRUE,spacing=0.10)

boxplot(AgeYears~RaceEthnicity, data=EmployeeBiometric.df,
        main="Age (Years) by Race Ethnicity", ylab="Age (Years)",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(AgeYears~RaceEthnicity, data=EmployeeBiometric.df,
         col=c("red","blue","green","cyan","orange"),add=TRUE,spacing=0.10)

boxplot(AgeYears~BMIStatus, data=EmployeeBiometric.df,
        main="Age (Years) by BMI Status", ylab="Age (Years)",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(AgeYears~BMIStatus, data=EmployeeBiometric.df,
         col=c("red","blue","green","black"),add=TRUE,spacing=0.10)

boxplot(AgeYears~Obesity, data=EmployeeBiometric.df,
        main="Age (Years) by Obesity", ylab="Age (Years)",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(AgeYears~Obesity, data= EmployeeBiometric.df,
         col=c("red","blue"), add=TRUE, spacing=0.1)

```



2. Distribution of Total Cholesterol based on categorical variables :

```

par(ask=TRUE)
par(mfrow=c(2,2))

boxplot(TotalCholesterolmgdL ~ Gender, data=EmployeeBiometric.df,
        main="Total Cholesterol by Gender", ylab="Total Cholesterol",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(TotalCholesterolmgdL~Gender, data=EmployeeBiometric.df,
          col=c("red","blue"),add=TRUE,spacing=0.10)

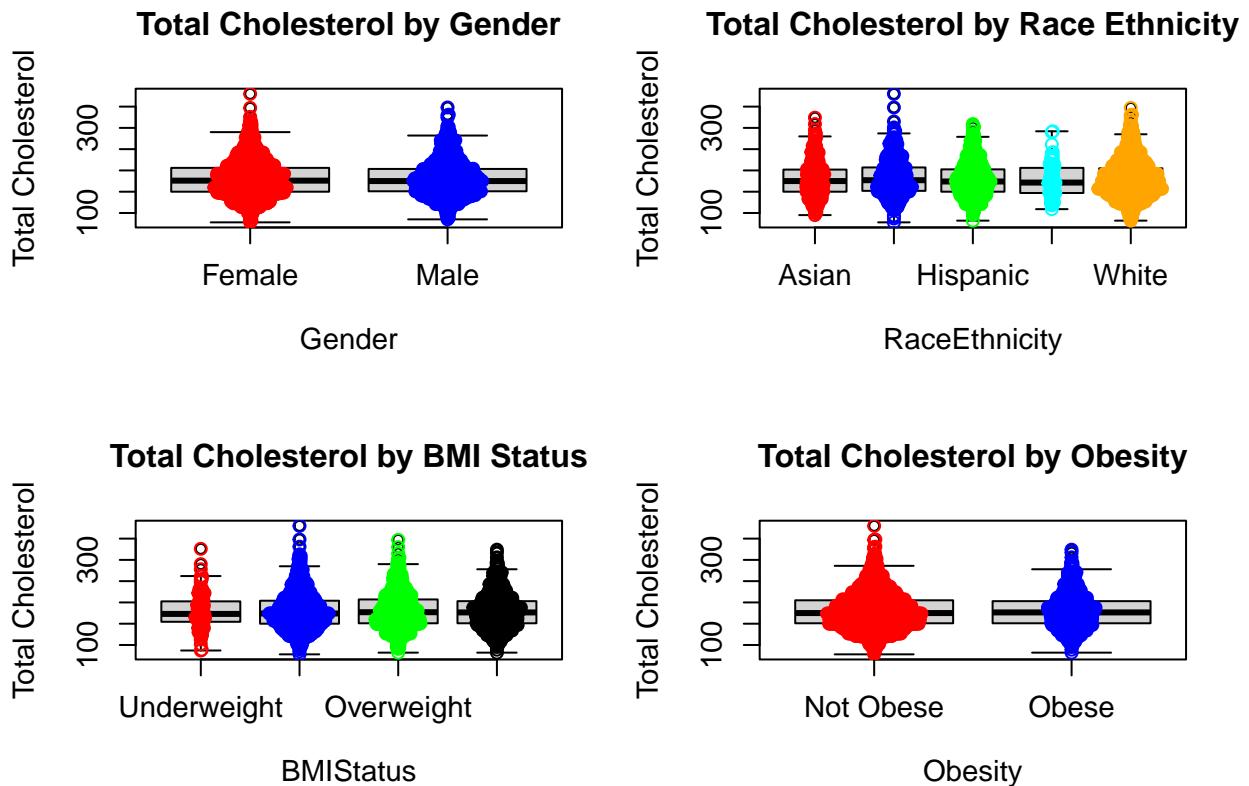
boxplot(TotalCholesterolmgdL~RaceEthnicity, data=EmployeeBiometric.df,
        main="Total Cholesterol by Race Ethnicity", ylab="Total Cholesterol",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(TotalCholesterolmgdL~RaceEthnicity, data=EmployeeBiometric.df,
          col=c("red","blue","green","cyan","orange"),add=TRUE,spacing=0.10)

boxplot(TotalCholesterolmgdL~BMISstatus, data=EmployeeBiometric.df,
        main="Total Cholesterol by BMI Status", ylab="Total Cholesterol",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(TotalCholesterolmgdL~BMISstatus, data=EmployeeBiometric.df,
          col=c("red","blue","green","black"),add=TRUE,spacing=0.10)

boxplot(TotalCholesterolmgdL~Obesity, data=EmployeeBiometric.df,
        main="Total Cholesterol by Obesity", ylab="Total Cholesterol",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(TotalCholesterolmgdL~Obesity, data= EmployeeBiometric.df,
          col=c("red","blue","green","black"),add=TRUE,spacing=0.10)

```

```
col=c("red","blue"), add=TRUE, spacing=0.1)
```



3. Distribution of SBP based on categorical variables :

```
par(ask=TRUE)
par(mfrow=c(2,2))
boxplot(SBPmmHg ~ Gender, data=EmployeeBiometric.df,
        main="Systolic Blood Pressure by Gender", ylab="Systolic Blood Pressure",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(SBPmmHg~Gender, data=EmployeeBiometric.df,
          col=c("red","blue"),add=TRUE,spacing=0.10)

boxplot(SBPmmHg~RaceEthnicity, data=EmployeeBiometric.df,
        main="Systolic Blood Pressure by Race Ethnicity", ylab="Systolic Blood Pressure",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(SBPmmHg~RaceEthnicity, data=EmployeeBiometric.df,
          col=c("red","blue","green","cyan","orange"),add=TRUE,spacing=0.10)

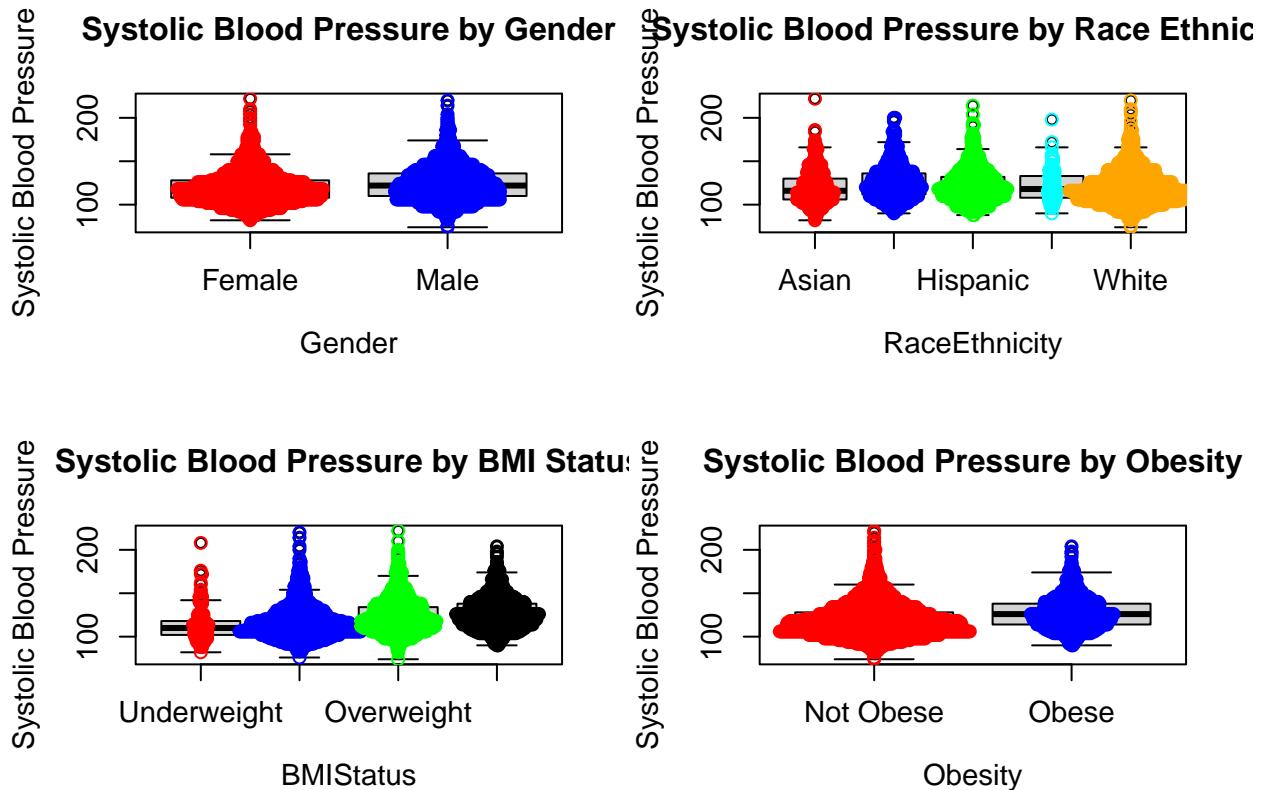
boxplot(SBPmmHg~BMIStatus, data=EmployeeBiometric.df,
        main="Systolic Blood Pressure by BMI Status", ylab="Systolic Blood Pressure",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(SBPmmHg~BMIStatus, data=EmployeeBiometric.df,
          col=c("red","blue","green","black"),add=TRUE,spacing=0.10)

boxplot(SBPmmHg~Obesity, data=EmployeeBiometric.df,
```

```

main="Systolic Blood Pressure by Obesity", ylab="Systolic Blood Pressure",
cex.axis=1.15, cex.lab=1.15)
beeswarm(SBPmmHg~Obesity, data= EmployeeBiometric.df,
col=c("red","blue"), add=TRUE, spacing=0.1)

```



4. Distribution of DBP based on categorical variables :

```

par(ask=TRUE)
par(mfrow=c(2,2))
boxplot(DBPmmHg ~ Gender, data=EmployeeBiometric.df,
       main="Diastolic Blood Pressure by Gender", ylab="Diastolic Blood Pressure",
       cex.axis=1.15, cex.lab=1.15)
beeswarm(DBPmmHg~Gender, data=EmployeeBiometric.df,
         col=c("red","blue"),add=TRUE,spacing=0.10)

boxplot(DBPmmHg~RaceEthnicity, data=EmployeeBiometric.df,
        main="Diastolic Blood Pressure by Race Ethnicity", ylab="Diastolic Blood Pressure",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(DBPmmHg~RaceEthnicity, data=EmployeeBiometric.df,
          col=c("red","blue","green","cyan","orange"),add=TRUE,spacing=0.10)

boxplot(DBPmmHg~BMISstatus, data=EmployeeBiometric.df,
        main="Diastolic Blood Pressure by BMI Status", ylab="Diastolic Blood Pressure",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(DBPmmHg~BMISstatus, data=EmployeeBiometric.df,

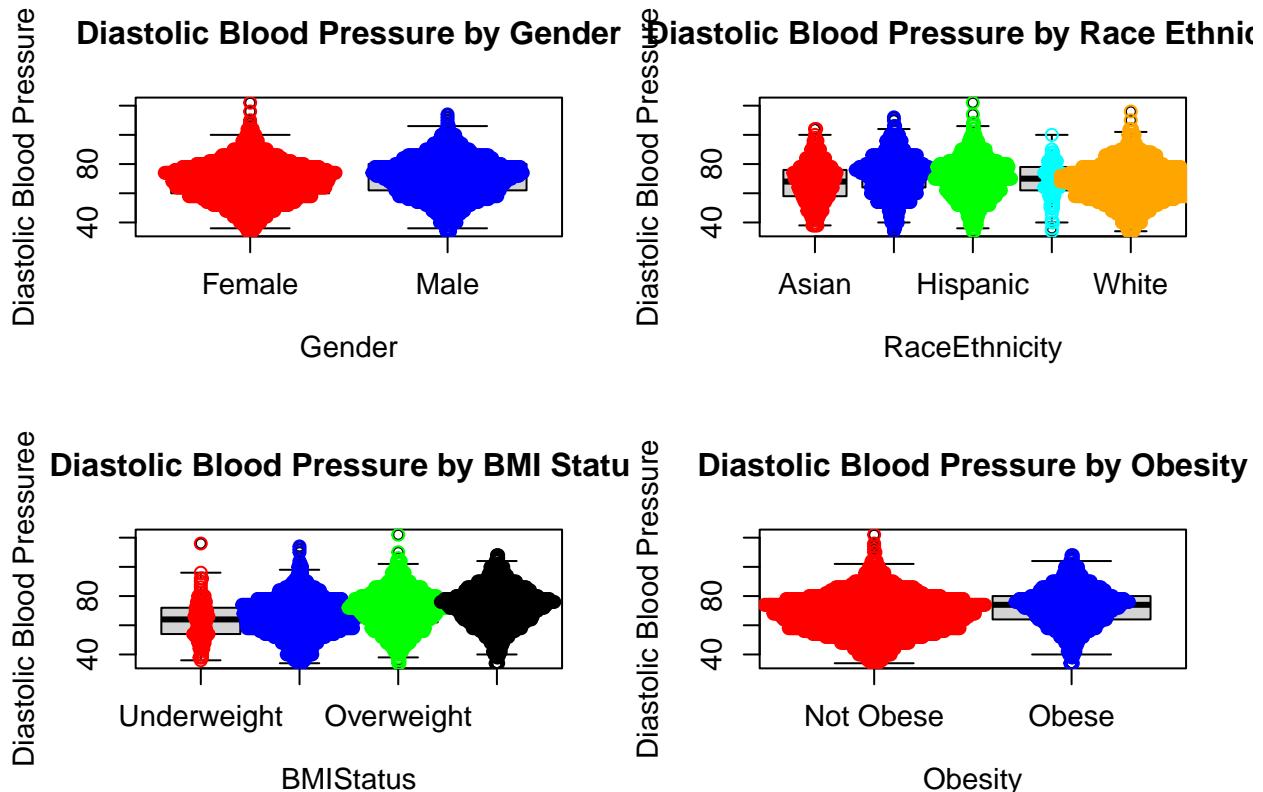
```

```

    col=c("red","blue","green","black"),add=TRUE,spacing=0.10)

boxplot(DBPmmHg~Obesity, data=EmployeeBiometric.df,
        main="Diastolic Blood Pressure by Obesity", ylab="Diastolic Blood Pressure",
        cex.axis=1.15, cex.lab=1.15)
beeswarm(DBPmmHg~Obesity, data= EmployeeBiometric.df,
          col=c("red","blue"), add=TRUE, spacing=0.1)

```



5. Distribution of BMIMetric by categorical variables

```

par(ask=TRUE)
par(mfrow=c(2,2))

boxplot(BMIMetric~Gender, data=EmployeeBiometric.df,
        main = "BMIMetric by Gender",
        ylab = "BMI Metric", cex.axis=1.15, cex.lab=1.15)
beeswarm(BMIMetric~Gender, data=EmployeeBiometric.df,
          col = c("red","blue"), add = TRUE, spacing=0.10)

boxplot(BMIMetric~RaceEthnicity, data=EmployeeBiometric.df,
        main = "BMIMetric by RaceEthnicity",
        ylab = "BMI Metric", cex.axis=1.15, cex.lab=1.15)
beeswarm(BMIMetric~RaceEthnicity, data=EmployeeBiometric.df,
          col = c("red","blue","green","yellow","cyan"), add = TRUE, spacing=0.10)

boxplot(BMIMetric~BMIStatus, data=EmployeeBiometric.df,
        main = "BMIMetric by BMISatus",
        ylab = "BMI Metric", cex.axis=1.15, cex.lab=1.15)
beeswarm(BMIMetric~BMISatus, data=EmployeeBiometric.df,
          col = c("red","blue","green","yellow","cyan"), add = TRUE, spacing=0.10)

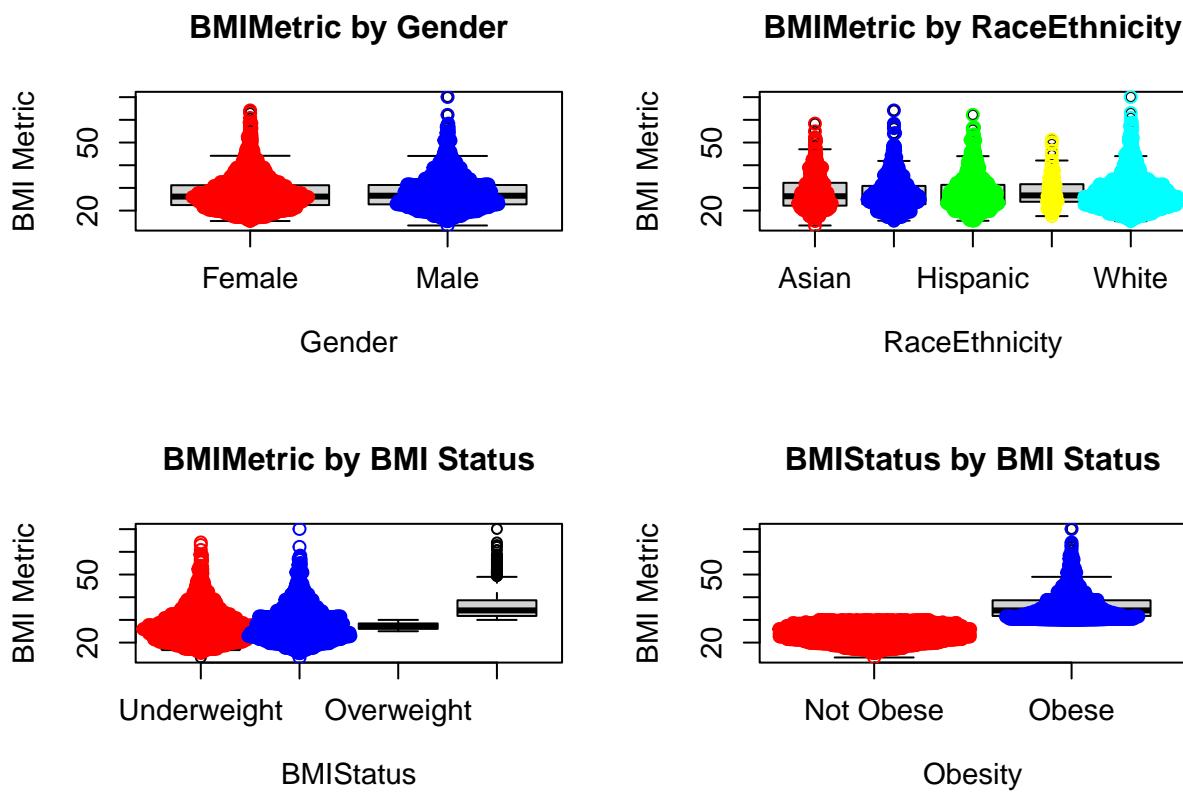
```

```

    main = "BMIMetric by BMI Status",
    ylab = "BMI Metric", cex.axis=1.15, cex.lab=1.15)
beeswarm(BMIMetric~Gender, data=EmployeeBiometric.df,
          col = c("red","blue","green","yellow"), add = TRUE, spacing=0.10)

boxplot(BMIMetric~Obesity, data=EmployeeBiometric.df,
        main = "BMIStatus by BMI Status",
        ylab = "BMI Metric", cex.axis=1.15, cex.lab=1.15)
beeswarm(BMIMetric~Obesity, data=EmployeeBiometric.df,
          col = c("red","blue"), add = TRUE, spacing=0.10)

```



Quality Assurance check using Sapiro-test to obtain normal/non normal distribution

```

shapiro.test((EmployeeBiometric.df$AgeYears))

##
## Shapiro-Wilk normality test
##
## data: (EmployeeBiometric.df$AgeYears)
## W = 0.9649, p-value <0.0000000000000002

shapiro.test(EmployeeBiometric.df$TotalCholesterolmgdL)

##

```

```

##  Shapiro-Wilk normality test
##
## data: EmployeeBiometric.df$TotalCholesterolmgdL
## W = 0.9783, p-value <0.0000000000000002

shapiro.test((EmployeeBiometric.df$SBPmmHg))

##  Shapiro-Wilk normality test
##
## data: (EmployeeBiometric.df$SBPmmHg)
## W = 0.95, p-value <0.0000000000000002

shapiro.test(EmployeeBiometric.df$DBPmmHg)

##  Shapiro-Wilk normality test
##
## data: EmployeeBiometric.df$DBPmmHg
## W = 0.9962, p-value = 0.0000000321

shapiro.test(EmployeeBiometric.df$BMIMetric)

```

```

##  Shapiro-Wilk normality test
##
## data: EmployeeBiometric.df$BMIMetric
## W = 0.9235, p-value <0.0000000000000002

```

Result of Shapiro test distribution : 1. Age = not normal 2. Total cholesterol = not normal 3. SBP = not normal 4. DBP = not normal 5. BMIMetrics = not normal

Distribution analysis using Q-Q Plot

```

par(ask=TRUE)
par(mfrow=c(2,3))
qqnorm(EmployeeBiometric.df$AgeYears, col="red",
       main = "Normality of AgeYears-Overall")
qqline(EmployeeBiometric.df$AgeYears, col="blue")

qqnorm(EmployeeBiometric.df$TotalCholesterolmgdL, col="red",
       main="Normality of Total Cholesterol - Overall")
qqline(EmployeeBiometric.df$TotalCholesterolmgdL, col="blue")

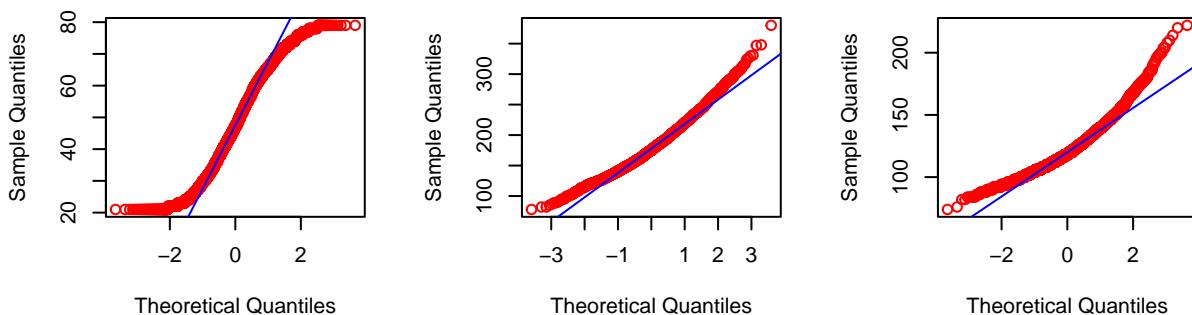
qqnorm(EmployeeBiometric.df$SBPmmHg, col="red",
       main="Normality of SBPmmHg - Overall")
qqline(EmployeeBiometric.df$SBPmmHg, col="blue")

qqnorm(EmployeeBiometric.df$DBPmmHg, col="red",
       main="Normality of DBPmmHg - Overall")
qqline(EmployeeBiometric.df$DBPmmHg, col="blue")

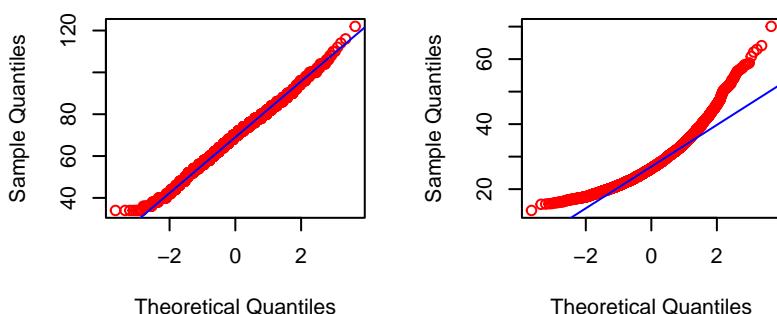
```

```
qqnorm(EmployeeBiometric.df$BMIMetric, col="red",
       main="Normality of BMI Metric - Overall")
qqline(EmployeeBiometric.df$BMIMetric, col="blue")
```

Normality of AgeYears–Overall **Normality of Total Cholesterol – Overall** **Normality of SBPmmHg – Overall**



Normality of DBPmmHg – Overall **Normality of BMI Metric – Overall**



As seen in the Q-Q plots provide ample evidence that the numeric object variables AgeYears, TotalCholesterolmgdL, SBPmmHg, DBPmmHg, and BMIMetric do not follow a normal distribution pattern.

The RVAideMemoire::byf.shapiro() function is an excellent choice to examine these questions on normality by breakouts. Observe below how the RVAideMemoire::byf.shapiro() function provides a simple and convenient summary of normality p-values by individual breakouts of the factor-type object variable.

1. Normality test of numeric object variables by Gender

```
RVAideMemoire:::byf.shapiro(AgeYears~Gender,
                             data=EmployeeBiometric.df)
```

```
##
##  Shapiro-Wilk normality tests
##
## data:  AgeYears by Gender
##
##          W            p-value
## Female  0.9661 < 0.000000000000022 ***
## Male    0.9633 < 0.000000000000022 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RVAideMemoire::byf.shapiro(TotalCholesterolmgdL~Gender,  
                           data=EmployeeBiometric.df)
```

```
##  
##  Shapiro-Wilk normality tests  
##  
## data: TotalCholesterolmgdL by Gender  
##  
##          W          p-value  
## Female 0.9806 0.000000000000047433 ***  
## Male   0.9739 0.00000000000002233 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RVAideMemoire::byf.shapiro(SBPmmHg ~ Gender,  
                           data=EmployeeBiometric.df)
```

```
##  
##  Shapiro-Wilk normality tests  
##  
## data: SBPmmHg by Gender  
##  
##          W          p-value  
## Female 0.9318 < 0.0000000000000022 ***  
## Male   0.9653 < 0.0000000000000022 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RVAideMemoire::byf.shapiro(DBPmmHg ~ Gender,  
                           data=EmployeeBiometric.df)
```

```
##  
##  Shapiro-Wilk normality tests  
##  
## data: DBPmmHg by Gender  
##  
##          W          p-value  
## Female 0.9945 0.000001046 ***  
## Male   0.9964  0.0004106 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RVAideMemoire::byf.shapiro(BMIMetric ~ Gender,  
                           data=EmployeeBiometric.df)
```

```
##  
##  Shapiro-Wilk normality tests  
##
```

```

## data: BMIMetric by Gender
##
##          W           p-value
## Female  0.9206 < 0.000000000000022 ***
## Male    0.9259 < 0.000000000000022 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

2. Normality test of numeric object variables by RaceEthnicity

```
RVAideMemoire::byf.shapiro(AgeYears ~ RaceEthnicity,
                           data=EmployeeBiometric.df)
```

```

##
## Shapiro-Wilk normality tests
##
## data: AgeYears by RaceEthnicity
##
##          W           p-value
## Asian   0.9702  0.00000002483267275 ***
## Black   0.9616  0.0000000000008035 ***
## Hispanic 0.9648  0.0000000000005845 ***
## Other   0.9336  0.00002059768431550 ***
## White   0.9625 < 0.000000000000022 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
RVAideMemoire::byf.shapiro(TotalCholesterolmgdL~RaceEthnicity,
                           data=EmployeeBiometric.df)
```

```

##
## Shapiro-Wilk normality tests
##
## data: TotalCholesterolmgdL by RaceEthnicity
##
##          W           p-value
## Asian   0.9784  0.0000299780295813 ***
## Black   0.9757  0.000000073145531 ***
## Hispanic 0.9783  0.0000000149272617 ***
## Other   0.9642       0.01574 *
## White   0.9777  0.000000000004407 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
RVAideMemoire::byf.shapiro(SBPmmHg~RaceEthnicity,
                           data=EmployeeBiometric.df)
```

```
##
```

```

## Shapiro-Wilk normality tests
##
## data: SBPmmHg by RaceEthnicity
##
##          W      p-value
## Asian    0.9415 0.0000000000222558 ***
## Black    0.9551 0.0000000000001435 ***
## Hispanic 0.9462 < 0.0000000000000022 ***
## Other    0.9320 0.00002641595847946 ***
## White    0.9448 < 0.0000000000000022 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
RVAideMemoire::byf.shapiro(DBPmmHg~RaceEthnicity,
                           data=EmployeeBiometric.df)
```

```

##
## Shapiro-Wilk normality tests
##
## data: DBPmmHg by RaceEthnicity
##
##          W      p-value
## Asian    0.9939 0.0657746 .
## Black    0.9935 0.0020140 **
## Hispanic 0.9942 0.0020703 **
## Other    0.9736 0.0263387 *
## White    0.9961 0.0005367 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
RVAideMemoire::byf.shapiro(BMIMetric~RaceEthnicity,
                           data=EmployeeBiometric.df)
```

```

##
## Shapiro-Wilk normality tests
##
## data: BMIMetric by RaceEthnicity
##
##          W      p-value
## Asian    0.9373 0.000000000002847 ***
## Black    0.9295 < 0.0000000000000022 ***
## Hispanic 0.9266 < 0.0000000000000022 ***
## Other    0.9217 0.0000038873197156 ***
## White    0.9118 < 0.0000000000000022 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

3. Normality test of numeric object variables by BMIstatus

```
RVAideMemoire::byf.shapiro(AgeYears~BMIStatus,  
                           data=EmployeeBiometric.df)
```

```
##  
## Shapiro-Wilk normality tests  
##  
## data: AgeYears by BMIStatus  
##  
##          W           p-value  
## Underweight 0.9526 0.0000193201320965490 ***  
## Normal Weight 0.9651 < 0.00000000000000022 ***  
## Overweight   0.9643 0.0000000000000006834 ***  
## Obese        0.9645 < 0.00000000000000022 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RVAideMemoire::byf.shapiro(TotalCholesterolmgdL~BMIStatus,  
                           data=EmployeeBiometric.df)
```

```
##  
## Shapiro-Wilk normality tests  
##  
## data: TotalCholesterolmgdL by BMIStatus  
##  
##          W           p-value  
## Underweight 0.9780 0.03646 *  
## Normal Weight 0.9710 0.0000000000006071 ***  
## Overweight   0.9785 0.0000000050969220 ***  
## Obese        0.9847 0.0000001972027389 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RVAideMemoire::byf.shapiro(SBPmmHg~BMIStatus,  
                           data=EmployeeBiometric.df)
```

```
##  
## Shapiro-Wilk normality tests  
##  
## data: SBPmmHg by BMIStatus  
##  
##          W           p-value  
## Underweight 0.8420 0.0000000001006922 ***  
## Normal Weight 0.9122 < 0.00000000000000022 ***  
## Overweight   0.9447 < 0.00000000000000022 ***  
## Obese        0.9702 0.0000000000001384 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RVAideMemoire::byf.shapiro(DBPmmHg~BMIStatus,  
                           data=EmployeeBiometric.df)
```

```
##  
##  Shapiro-Wilk normality tests  
##  
## data: DBPmmHg by BMIStatus  
##  
##          W      p-value  
## Underweight   0.9792  0.0180324 *  
## Normal Weight 0.9943  0.00005119 ***  
## Overweight    0.9947  0.0009071 ***  
## Obese         0.9944  0.0002697 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RVAideMemoire::byf.shapiro(BMIMetric~BMIStatus,  
                           data=EmployeeBiometric.df)
```

```
##  
##  Shapiro-Wilk normality tests  
##  
## data: BMIMetric by BMIStatus  
##  
##          W      p-value  
## Underweight   0.9201      0.00000005606 ***  
## Normal Weight 0.9626 < 0.00000000000000022 ***  
## Overweight    0.9590 < 0.00000000000000022 ***  
## Obese         0.8329 < 0.00000000000000022 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4. Normality test of numeric object variables by Obesity

```
RVAideMemoire::byf.shapiro(AgeYears~Obesity,  
                           data=EmployeeBiometric.df)
```

```
##  
##  Shapiro-Wilk normality tests  
##  
## data: AgeYears by Obesity  
##  
##          W      p-value  
## Not Obese  0.9647 < 0.00000000000000022 ***  
## Obese     0.9645 < 0.00000000000000022 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RVAideMemoire::byf.shapiro(TotalCholesterolmgdL~Obesity,  
                           data=EmployeeBiometric.df)
```

```
##  
##  Shapiro-Wilk normality tests  
##  
## data: TotalCholesterolmgdL by Obesity  
##  
##              W          p-value  
## Not Obese 0.9754 < 0.0000000000000002 ***  
## Obese     0.9847      0.0000001972 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RVAideMemoire::byf.shapiro(SBPmmHg~Obesity,  
                           data=EmployeeBiometric.df)
```

```
##  
##  Shapiro-Wilk normality tests  
##  
## data: SBPmmHg by Obesity  
##  
##              W          p-value  
## Not Obese 0.9298 < 0.0000000000000002 ***  
## Obese     0.9702      0.0000000000001384 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RVAideMemoire::byf.shapiro(DBPmmHg~Obesity,  
                           data=EmployeeBiometric.df)
```

```
##  
##  Shapiro-Wilk normality tests  
##  
## data: DBPmmHg by Obesity  
##  
##              W          p-value  
## Not Obese 0.9953 0.000000308 ***  
## Obese     0.9944  0.0002697 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RVAideMemoire::byf.shapiro(BMIMetric~Obesity,  
                           data=EmployeeBiometric.df)
```

```
##  
##  Shapiro-Wilk normality tests  
##
```

```

## data: BMIMetric by Obesity
##
##              W          p-value
## Not Obese  0.9789 < 0.0000000000000022 ***
## Obese      0.8329 < 0.0000000000000022 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

For additional confirmation, use the `ggplot2::ggplot()` function to produce QQ plots of the numeric object variables (e.g., AgeYears, TotalCholesterolmgdL, SBPmmHg, DBPmmHg, and BMIMetric) by breakouts of each of the four factor-type object variables (e.g., Gender, RaceEthnicity, BMIStatus, and Obesity)

1.AgeYears

```

QQAgeYearsbyGender<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=AgeYears))+
  stat_qq(color="red")+
  stat_qq_line(color="blue", linewidth=1.75)+
  facet_grid(.~Gender)+
  ggtitle("QQ-AgeYears")

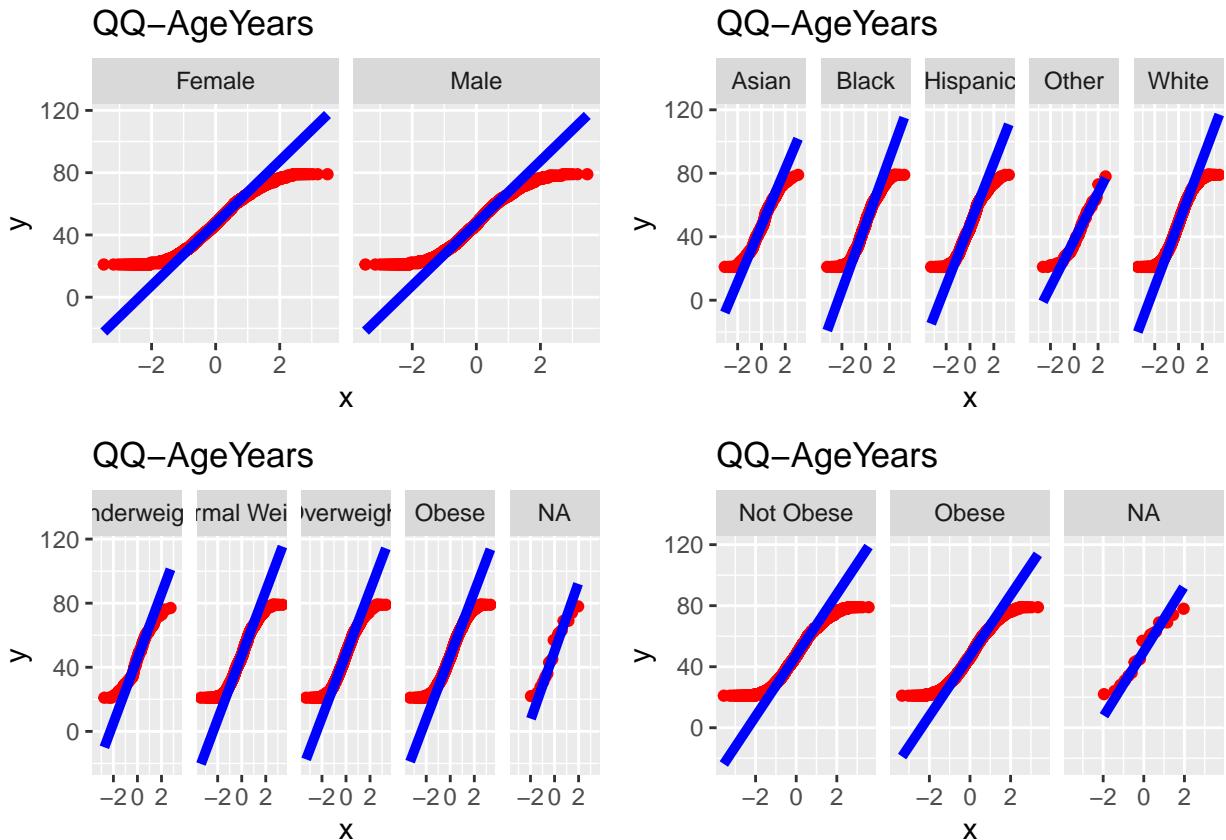
QQAgeYearsbyRaceEthnicity<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=AgeYears))+
  stat_qq(color="red")+
  stat_qq_line(color="blue", linewidth=1.75)+
  facet_grid(.~RaceEthnicity)+
  ggtitle("QQ-AgeYears")

QQAgeYearsbyBMISatus<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=AgeYears))+
  stat_qq(color="red")+
  stat_qq_line(color="blue", linewidth=1.75)+
  facet_grid(.~BMISatus)+
  ggtitle("QQ-AgeYears")

QQAgeYearsbyObesity<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=AgeYears))+
  stat_qq(color="red")+
  stat_qq_line(color="blue", linewidth=1.75)+
  facet_grid(.~Obesity)+
  ggtitle("QQ-AgeYears")

par(ask=TRUE); gridExtra::grid.arrange(
  QQAgeYearsbyGender,
  QQAgeYearsbyRaceEthnicity,
  QQAgeYearsbyBMISatus,
  QQAgeYearsbyObesity, ncol=2)

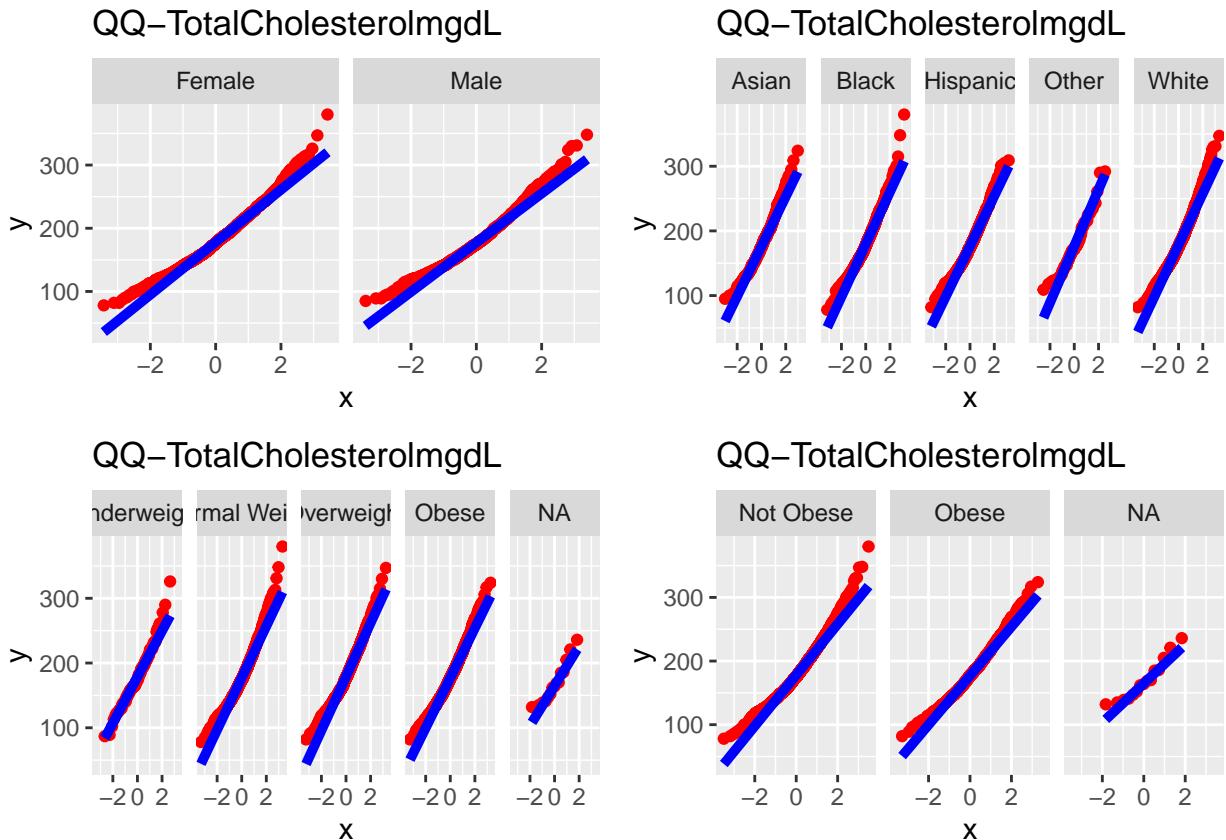
```



2. TotalCholesterol

```
QQTotalCholesterolbyGender<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=TotalCholesterolmgdL))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~Gender) +  
  ggtitle("QQ-TotalCholesterolmgdL")  
  
QQTotalCholesterolbyRaceEthnicity<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=TotalCholesterolmgdL))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~RaceEthnicity) +  
  ggtitle("QQ-TotalCholesterolmgdL")  
  
QQTotalCholesterolbyBMIStatus<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=TotalCholesterolmgdL))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~BMIStatus) +  
  ggtitle("QQ-TotalCholesterolmgdL")  
  
QQTotalCholesterolbyObesity<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=TotalCholesterolmgdL))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~Obesity) +  
  ggtitle("QQ-TotalCholesterolmgdL")  
  
par(ask=TRUE); gridExtra::grid.arrange(
```

```
QQTotalCholesterolbyGender,  
QQTotalCholesterolbyRaceEthnicity,  
QQTotalCholesterolbyBMIStatus,  
QQTotalCholesterolbyObesity, ncol=2)  
  
## Warning: Removed 894 rows containing non-finite outside the scale  
## range ('stat_qq()').  
  
## Warning: Removed 894 rows containing non-finite outside the scale  
## range ('stat_qq_line()').  
  
## Warning: Removed 894 rows containing non-finite outside the scale  
## range ('stat_qq()').  
  
## Warning: Removed 894 rows containing non-finite outside the scale  
## range ('stat_qq_line()').  
  
## Warning: Removed 894 rows containing non-finite outside the scale  
## range ('stat_qq()').  
  
## Warning: Removed 894 rows containing non-finite outside the scale  
## range ('stat_qq_line()').  
  
## Warning: Removed 894 rows containing non-finite outside the scale  
## range ('stat_qq()').
```



3. SBPmmHg

```

QQSBPbyGender<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=SBPmmHg))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~Gender) +  
  ggttitle("QQ–SBPmmHg")

QQSBPbyRaceEthnicity<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=SBPmmHg))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~RaceEthnicity) +  
  ggttitle("QQ–SBPmmHg")

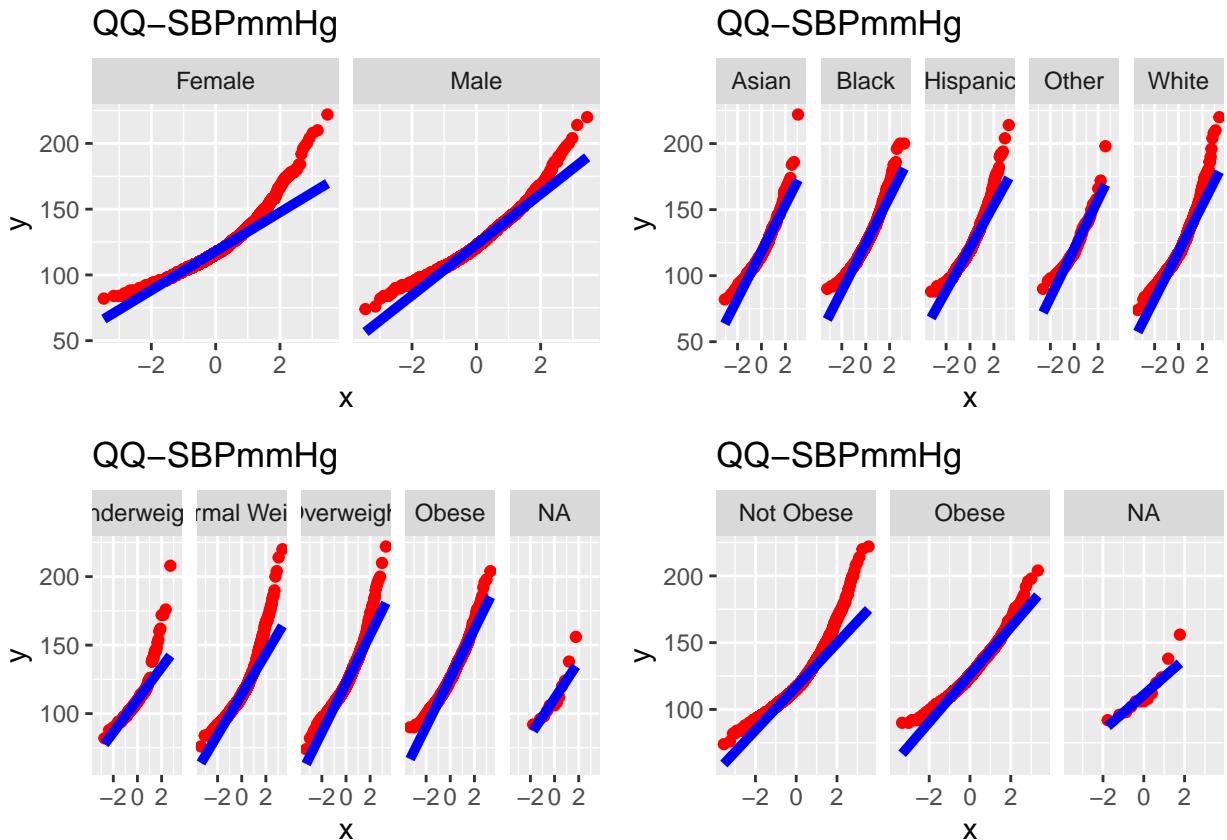
QQSBPbyBMIStatus<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=SBPmmHg))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~BMIStatus) +  
  ggttitle("QQ–SBPmmHg")

QQSBPbyObesity<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=SBPmmHg))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~Obesity) +  
  ggttitle("QQ–SBPmmHg")

par(ask=TRUE); gridExtra::grid.arrange(

```

```
QQSBPbyGender,  
QQSBPbyRaceEthnicity,  
QQSBPbyBMIStatus,  
QQSBPbyObesity, ncol=2)  
  
## Warning: Removed 209 rows containing non-finite outside the scale  
## range ('stat_qq()').  
  
## Warning: Removed 209 rows containing non-finite outside the scale  
## range ('stat_qq_line()').  
  
## Warning: Removed 209 rows containing non-finite outside the scale  
## range ('stat_qq()').  
  
## Warning: Removed 209 rows containing non-finite outside the scale  
## range ('stat_qq_line()').  
  
## Warning: Removed 209 rows containing non-finite outside the scale  
## range ('stat_qq()').  
  
## Warning: Removed 209 rows containing non-finite outside the scale  
## range ('stat_qq_line()').  
  
## Warning: Removed 209 rows containing non-finite outside the scale  
## range ('stat_qq()').
```



4. DBPmmHg

```

QQDBPbyGender<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=DBPmmHg))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~Gender) +  
  ggttitle("QQ-DBPmmHg")

QQDBPbyRaceEthnicity<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=DBPmmHg))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~RaceEthnicity) +  
  ggttitle("QQ-DBPmmHg")

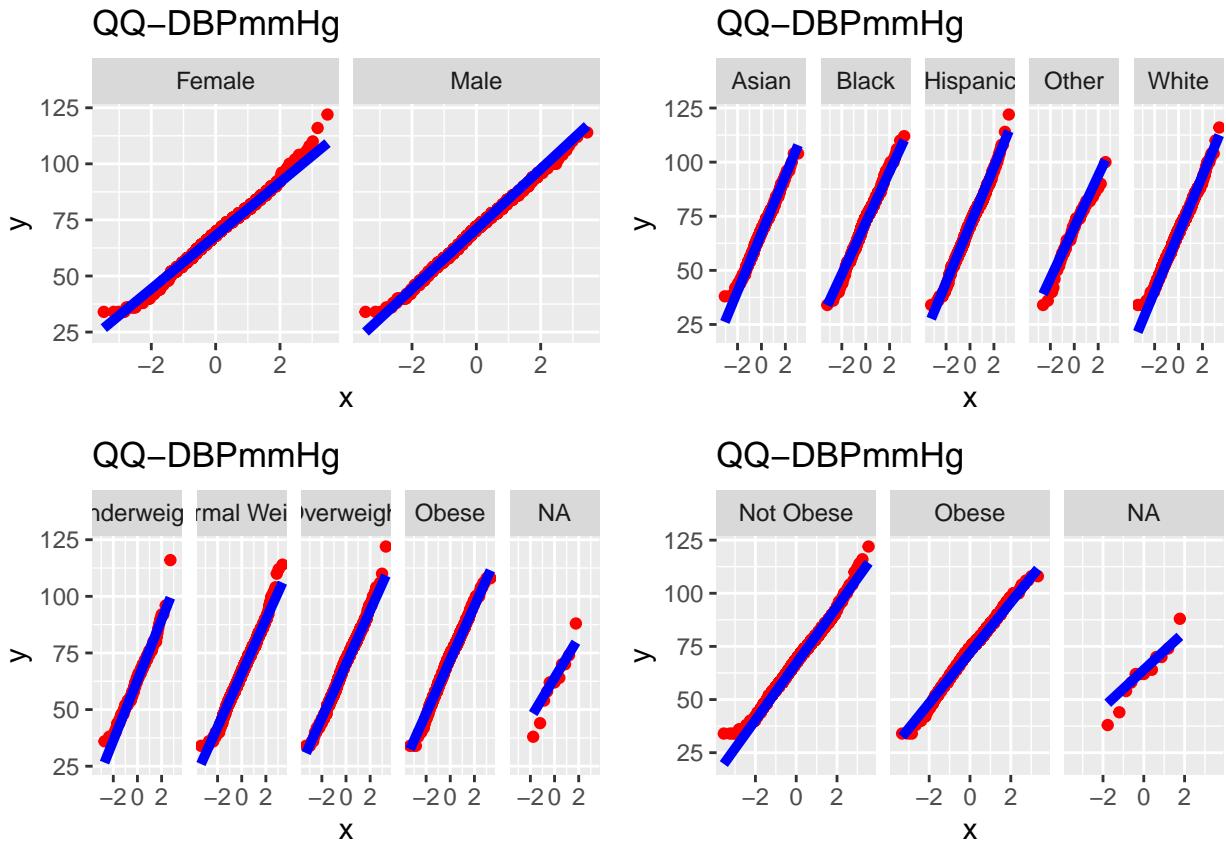
QQDBPbyBMIStatus<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=DBPmmHg))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~BMIStatus) +  
  ggttitle("QQ-DBPmmHg")

QQDBPbyObesity<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=DBPmmHg))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~Obesity) +  
  ggttitle("QQ-DBPmmHg")

par(ask=TRUE); gridExtra::grid.arrange(

```

```
QQDBPbyGender,  
QQDBPbyRaceEthnicity,  
QQDBPbyBMIStatus,  
QQDBPbyObesity, ncol=2)  
  
## Warning: Removed 217 rows containing non-finite outside the scale  
## range ('stat_qq()').  
  
## Warning: Removed 217 rows containing non-finite outside the scale  
## range ('stat_qq_line()').  
  
## Warning: Removed 217 rows containing non-finite outside the scale  
## range ('stat_qq()').  
  
## Warning: Removed 217 rows containing non-finite outside the scale  
## range ('stat_qq_line()').  
  
## Warning: Removed 217 rows containing non-finite outside the scale  
## range ('stat_qq()').  
  
## Warning: Removed 217 rows containing non-finite outside the scale  
## range ('stat_qq_line()').  
  
## Warning: Removed 217 rows containing non-finite outside the scale  
## range ('stat_qq()').
```



5. BMIMetric

```

QQBMIMetricbyGender<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=BMIMetric))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~Gender) +  
  ggtitle("QQ-BMIMetric")

QQBMIMetricbyRaceEthnicity<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=BMIMetric))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~RaceEthnicity) +  
  ggtitle("QQ-BMIMetric")

QQBMIMetricbyBMIStatus<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=BMIMetric))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~BMIStatus) +  
  ggtitle("QQ-BMIMetric")

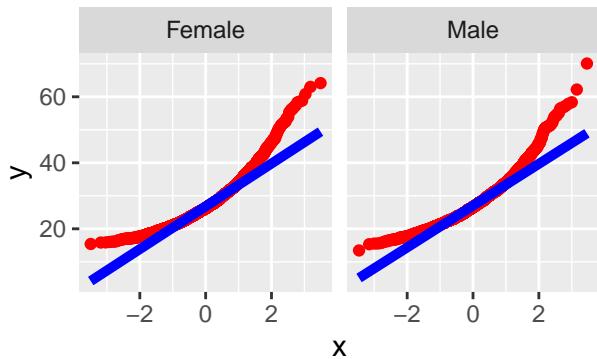
QQBMIMetricbyObesity<-ggplot2::ggplot(EmployeeBiometric.df,aes(sample=BMIMetric))+  
  stat_qq(color="red") +  
  stat_qq_line(color="blue", linewidth=1.75) +  
  facet_grid(.~Obesity) +  
  ggtitle("QQ-BMIMetric")

par(ask=TRUE); gridExtra::grid.arrange(

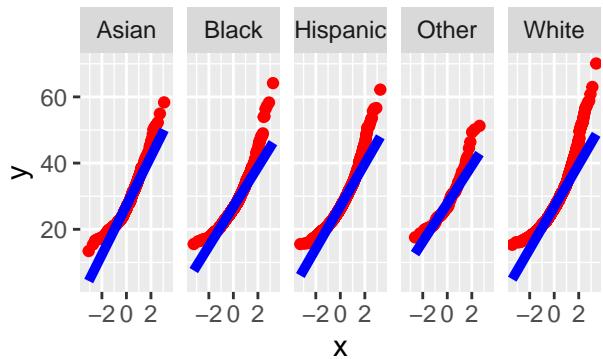
```

```
QQBMIMetricbyGender,  
QQBMIMetricbyRaceEthnicity,  
QQBMIMetricbyBMIStatus,  
QQBMIMetricbyObesity, ncol=2)  
  
## Warning: Removed 20 rows containing non-finite outside the scale  
## range ('stat_qq()').  
  
## Warning: Removed 20 rows containing non-finite outside the scale  
## range ('stat_qq_line()').  
  
## Warning: Removed 20 rows containing non-finite outside the scale  
## range ('stat_qq()').  
  
## Warning: Removed 20 rows containing non-finite outside the scale  
## range ('stat_qq_line()').  
  
## Warning: Removed 20 rows containing non-finite outside the scale  
## range ('stat_qq()').  
  
## Warning: Removed 20 rows containing non-finite outside the scale  
## range ('stat_qq_line()').  
  
## Warning: Removed 20 rows containing non-finite outside the scale  
## range ('stat_qq()').
```

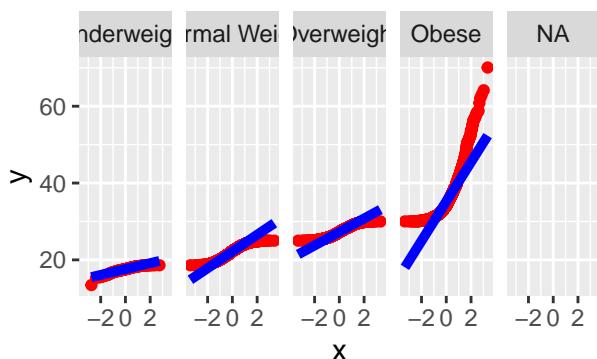
QQ-BMIMetric



QQ-BMIMetric



QQ-BMIMetric



QQ-BMIMetric

