

```

# Creating a dataframe
# Example: RPI Weather dataframe,

days <- c('Mon','Tue', 'Wed', 'Thur', 'Fri', 'Sat', 'Sun') # days
temp <- c(28,30.5,32,31.2,29.3,27.9,26.4) # Temperature in F' during the winter
snowed <- c('T','T','F','F','T','T','F') # Snowed on that day:T = TRUE, F= FALSE
help("data.frame")

## starting httpd help server ... done

RPI_Weather_Week <- data.frame(days, temp, snowed) # creating the dataframe using the data.frame

RPI_Weather_Week

##   days temp snowed
## 1 Mon 28.0      T
## 2 Tue 30.5      T
## 3 Wed 32.0      F
## 4 Thur 31.2      F
## 5 Fri 29.3      T
## 6 Sat 27.9      T
## 7 Sun 26.4      F

head(RPI_Weather_Week) # head of the data frame, NOTE: it will show only 6 rows, usually head() function

##   days temp snowed
## 1 Mon 28.0      T
## 2 Tue 30.5      T
## 3 Wed 32.0      F
## 4 Thur 31.2      F
## 5 Fri 29.3      T
## 6 Sat 27.9      T

str(RPI_Weather_Week) # we can take a look at the structure of the dataframe using the str() function

## 'data.frame':   7 obs. of  3 variables:
## $ days : Factor w/ 7 levels "Fri","Mon","Sat",...: 2 6 7 5 1 3 4
## $ temp : num  28 30.5 32 31.2 29.3 27.9 26.4
## $ snowed: Factor w/ 2 levels "F","T": 2 2 1 1 2 2 1

summary(RPI_Weather_Week) # summary of the dataframe using the summary function

##   days      temp      snowed
## Fri :1   Min.   :26.40   F:3
## Mon :1  1st Qu.:27.95   T:4
## Sat :1   Median :29.30
## Sun :1    Mean  :29.33
## Thur:1  3rd Qu.:30.85
## Tue :1    Max.   :32.00
## Wed :1

RPI_Weather_Week[1,] # showing the 1st row and all columns

##   days temp snowed
## 1 Mon 28      T

RPI_Weather_Week[,1] # showing the 1st column and all rows

```

```
## [1] Mon Tue Wed Thur Fri Sat Sun
## Levels: Fri Mon Sat Sun Thur Tue Wed
RPI_Weather_Week[, 'snowed']

## [1] T T F F T T F
## Levels: F T
RPI_Weather_Week[, 'days']

## [1] Mon Tue Wed Thur Fri Sat Sun
## Levels: Fri Mon Sat Sun Thur Tue Wed
RPI_Weather_Week[, 'temp']

## [1] 28.0 30.5 32.0 31.2 29.3 27.9 26.4
RPI_Weather_Week[1:5, c("days", "temp")]

##   days temp
## 1 Mon 28.0
## 2 Tue 30.5
## 3 Wed 32.0
## 4 Thur 31.2
## 5 Fri 29.3
RPI_Weather_Week$temp

## [1] 28.0 30.5 32.0 31.2 29.3 27.9 26.4
subset(RPI_Weather_Week, SUBSET = snowed == TRUE)

##   days temp snowed
## 1 Mon 28.0      T
## 2 Tue 30.5      T
## 3 Wed 32.0      F
## 4 Thur 31.2      F
## 5 Fri 29.3      T
## 6 Sat 27.9      T
## 7 Sun 26.4      F
sorted.snowed <- order(RPI_Weather_Week['snowed'])
sorted.snowed

## [1] 3 4 7 1 2 5 6
RPI_Weather_Week[sorted.snowed,]

##   days temp snowed
## 3 Wed 32.0      F
## 4 Thur 31.2      F
## 7 Sun 26.4      F
## 1 Mon 28.0      T
## 2 Tue 30.5      T
## 5 Fri 29.3      T
## 6 Sat 27.9      T
# RPI_Weather_Week[descending_snowed,]
dec.snow <- order(-RPI_Weather_Week$temp)
dec.snow
```

```
## [1] 3 4 2 5 1 6 7
#creating dataframes
#creating an empty dataframe
empty.dataframe <- data.frame()
v1 <- 1:10
v1

## [1] 1 2 3 4 5 6 7 8 9 10
letters

## [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l" "m" "n" "o" "p" "q" "r" "s"
## [20] "t" "u" "v" "w" "x" "y" "z"

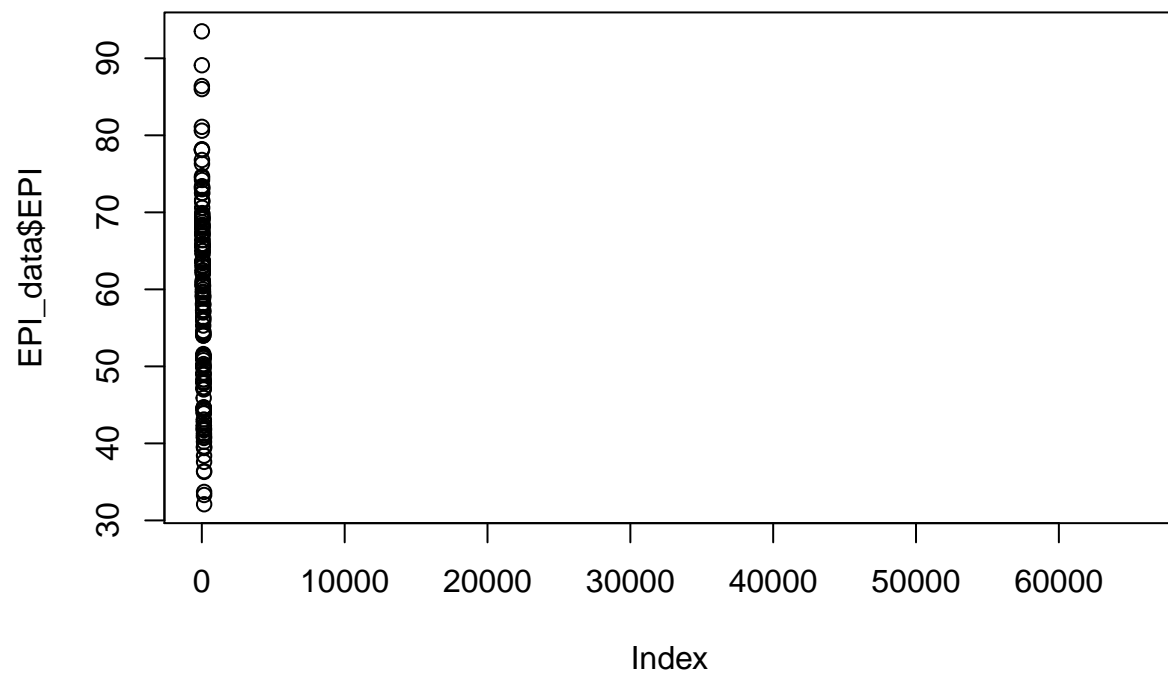
v2 <- letters[1:10]
df <- data.frame(col.name.1 = v1, col.name.2 = v2)
df

##      col.name.1 col.name.2
## 1           1          a
## 2           2          b
## 3           3          c
## 4           4          d
## 5           5          e
## 6           6          f
## 7           7          g
## 8           8          h
## 9           9          i
## 10          10          j

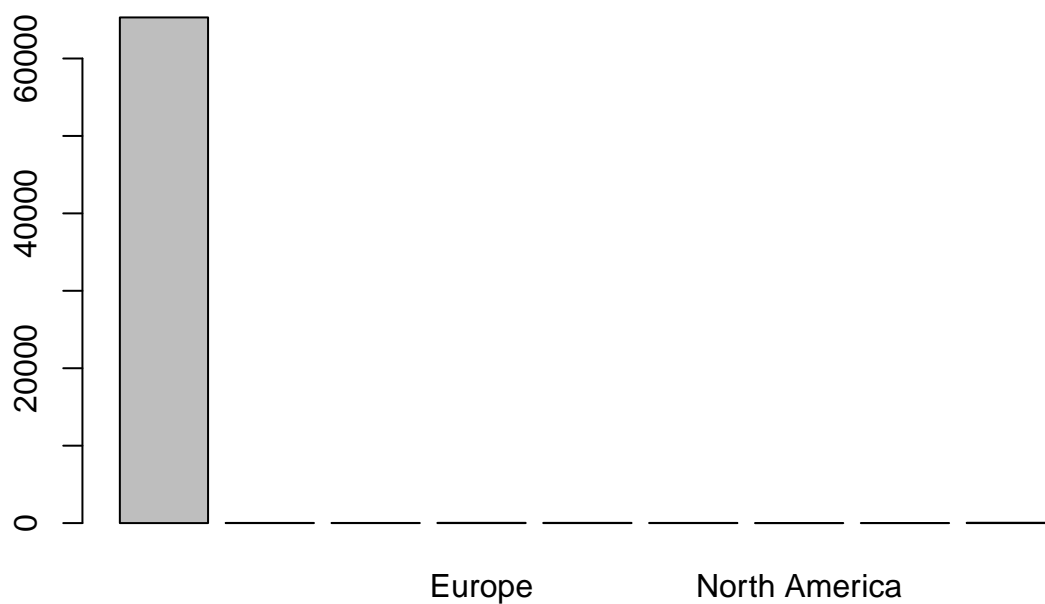
# Exercise - getting data in
rm(list=ls())
setwd("H:/RPI/Spring 2020/Data Analytics/group1")

GPW3 <- read.csv("GPW3_GRUMP_SummaryInformation_2010.csv")
#GPW3 <- read.csv(file.choose(), header = TRUE)
EPI_data <- read.csv("2010EPI_data.csv", skip=1)

plot(EPI_data$EPI)
```



```
plot(EPI_data$EPI_regions)
```



```
data()
help(data)

#Files
help("read.csv")
View(EPI_data)

attach(EPI_data)
fix(EPI_data)
EPI
```

```
##      [1] 93.5 89.1 86.4 86.0 81.1 80.6 78.2 78.1 78.1 76.8 76.3 74.7 74.5 74.2
##      [15] 73.4 73.3 73.2 73.1 73.0 72.5 72.5 71.6 71.4 71.4 70.6 69.9 69.8 69.6
##      [29] 69.4 69.3 69.3 69.2 69.1 69.1 68.7 68.4 68.3 68.2 68.2 68.0 67.8 67.4
##      [43] 67.3 67.1 67.0 66.4 66.4 65.9 65.9 65.7 65.7 65.6 65.4 65.0 65.0 64.6
##      [57] 63.8 63.7 63.6 63.5 63.5 63.4 63.1 62.9 62.5 62.4 62.2 62.0 61.2 61.0
##      [71] 60.9 60.8 60.6 60.6 60.5 60.4 60.4 60.0 59.7 59.6 59.3 59.2 59.1 59.1
##      [85] 59.0 58.8 58.2 58.1 58.0 57.9 57.3 57.3 57.1 57.0 56.4 56.3 56.1 55.9
##      [99] 55.3 54.6 54.4 54.3 54.2 54.0 54.0 51.6 51.4 51.4 51.3 51.3 51.3 51.2
##     [113] 51.1 51.1 50.8 50.3 50.1 49.9 49.8 49.2 49.0 48.9 48.3 48.3 48.0 47.9
##     [127] 47.8 47.3 47.1 47.0 45.9 44.7 44.6 44.6 44.6 44.4 44.3 44.3 44.0 43.9
##     [141] 43.1 42.8 42.3 42.3 42.0 41.9 41.8 41.7 41.3 41.0 40.8 40.7 40.2 39.6
##     [155] 39.5 39.4 38.4 37.6 36.4 36.3 33.7 33.3 32.1  NA   NA   NA   NA   NA
##     [169]  NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA
##     [183]  NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA
##     [197]  NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA
##     [211]  NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA
```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```
## [65241] NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65255] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65269] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65283] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65297] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65311] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65325] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65339] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65353] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65367] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65381] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65395] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65409] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65423] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65437] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65451] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [65465] NA NA NA
```

```
tf <- is.na(EPI)
E <- EPI[!tf]
```

```
# Exercise 1: exploring the distribution
summary(EPI)
```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##    32.10  48.60   59.20   58.37   67.60   93.50   65304
```

```
fivenum(EPI,na.rm=TRUE)
```

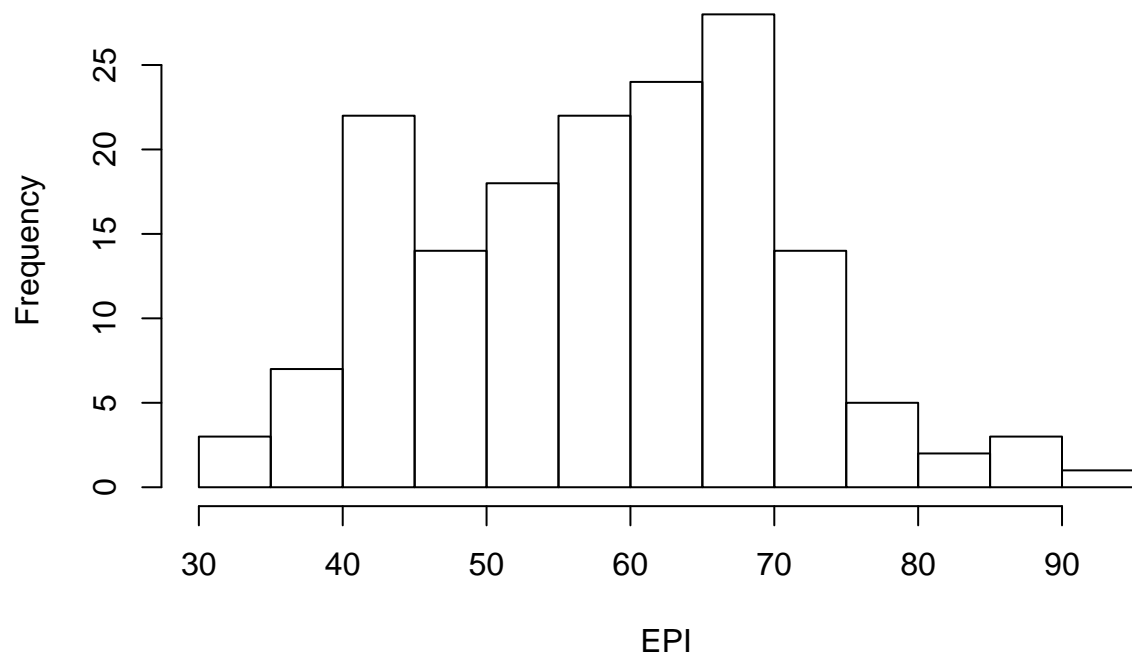
```
## [1] 32.1 48.6 59.2 67.6 93.5
```

```
help(stem)
stem(EPI)
```

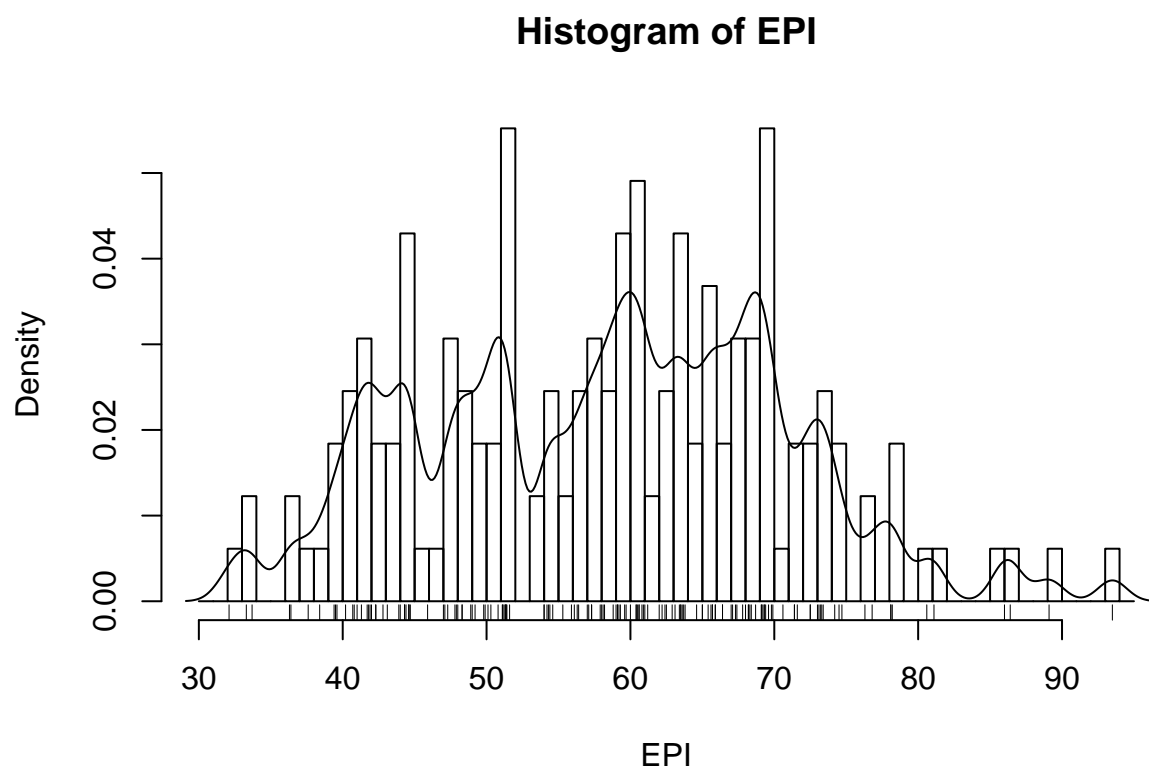
```
##
## The decimal point is 1 digit(s) to the right of the |
##
## 3 | 234
## 3 | 66889
## 4 | 00011112222223344444
## 4 | 5555677788888999
## 5 | 0000111111111244444
## 5 | 55666677778888999999
## 6 | 000001111111222333344444
## 6 | 5555666666677778888889999999
## 7 | 000111233333334
## 7 | 5567888
## 8 | 11
## 8 | 669
## 9 | 4
```

```
help(hist)
hist(EPI)
```

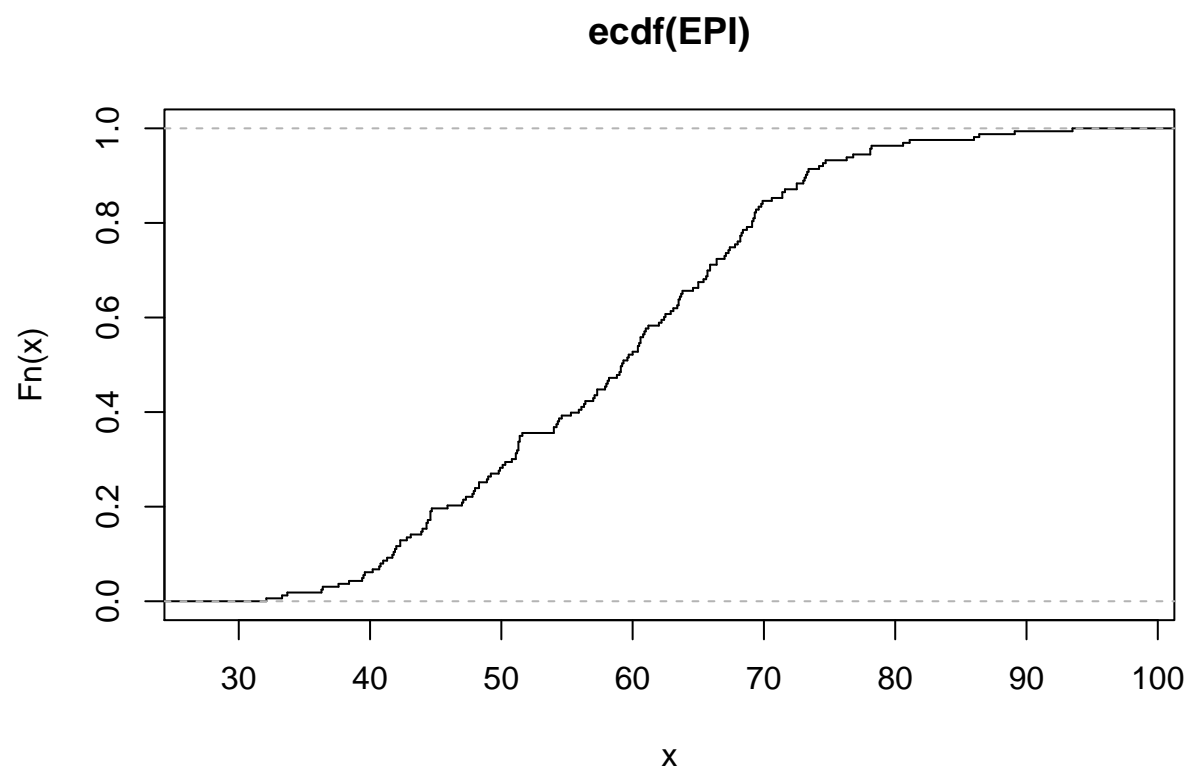
Histogram of EPI



```
hist(EPI, seq(30., 95., 1.0), prob=TRUE)
help(lines)
lines(density(EPI, na.rm=TRUE, bw=1.))
help(rug)
rug(EPI)
```

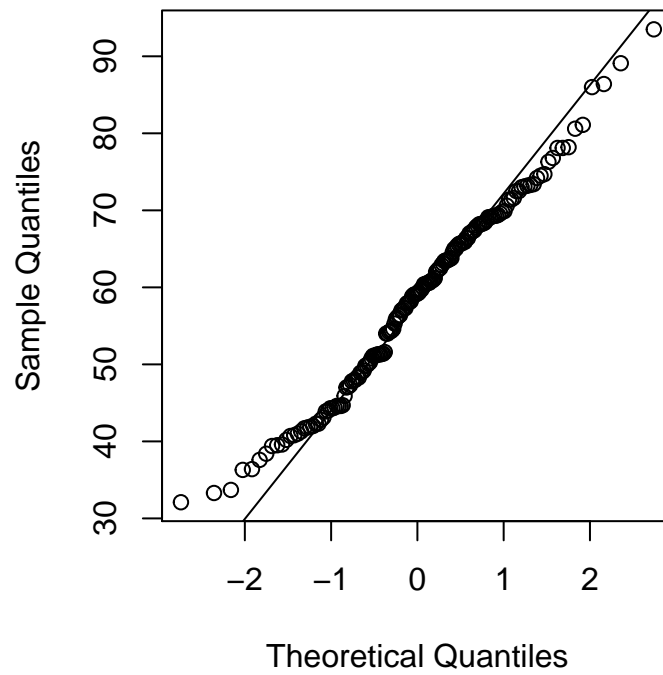


```
# Exercise 1: fitting a distribution beyond histograms  
plot(ecdf(EPI), do.points=FALSE, verticals=TRUE)
```

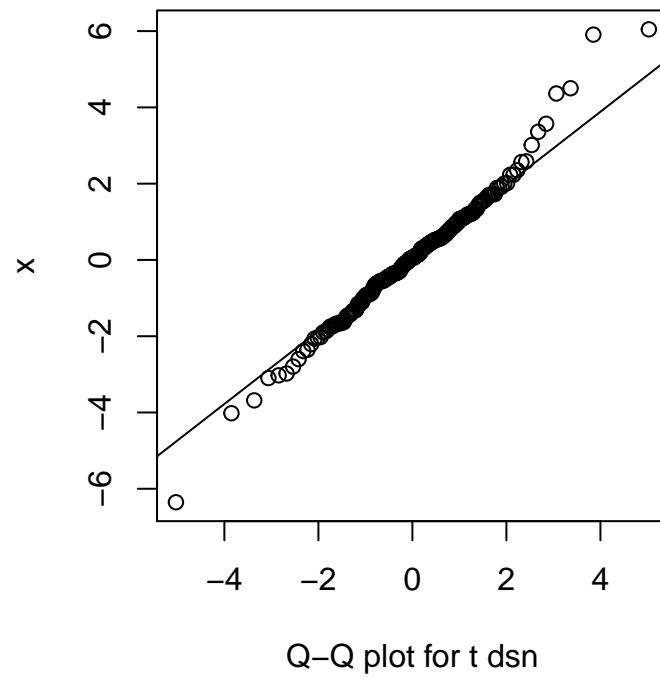


```
par(pty="s")  
?ecdf()  
qqnorm(EPI); qqline(EPI)
```

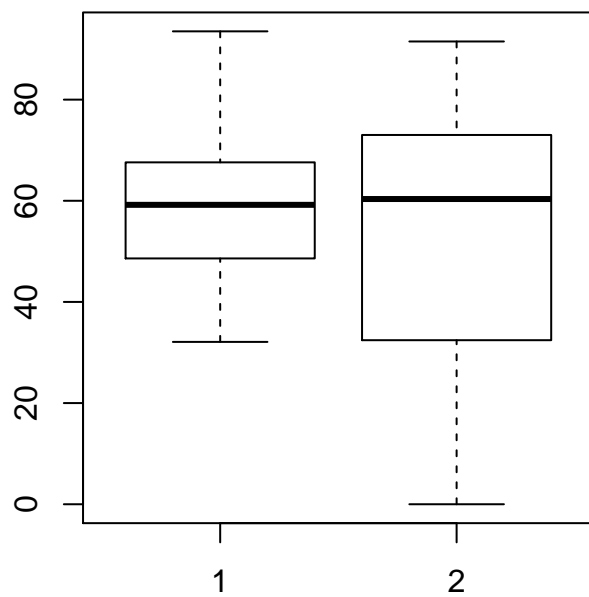
Normal Q-Q Plot



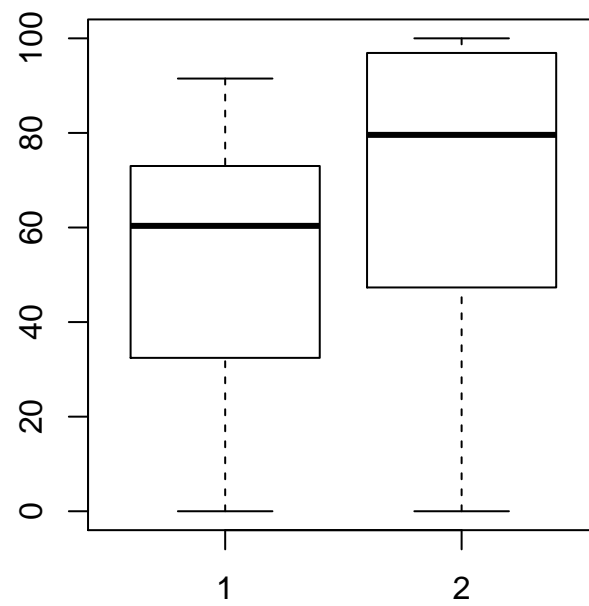
```
x <- rt(250, df = 5)
qqplot(qt(ppoints(250), df = 5), x, xlab = "Q-Q plot for t dsn")
?qt()
qqline(x)
```

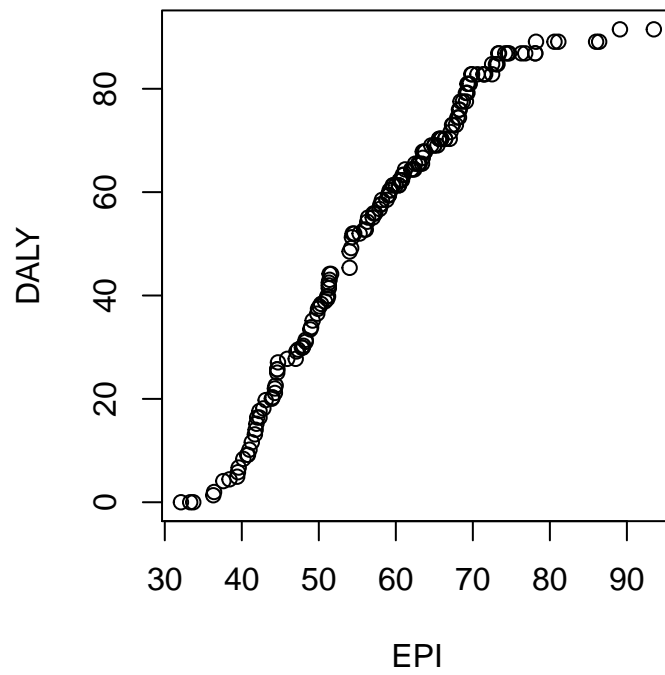
```
#Exercise 1  
boxplot(EPI, DALY)
```



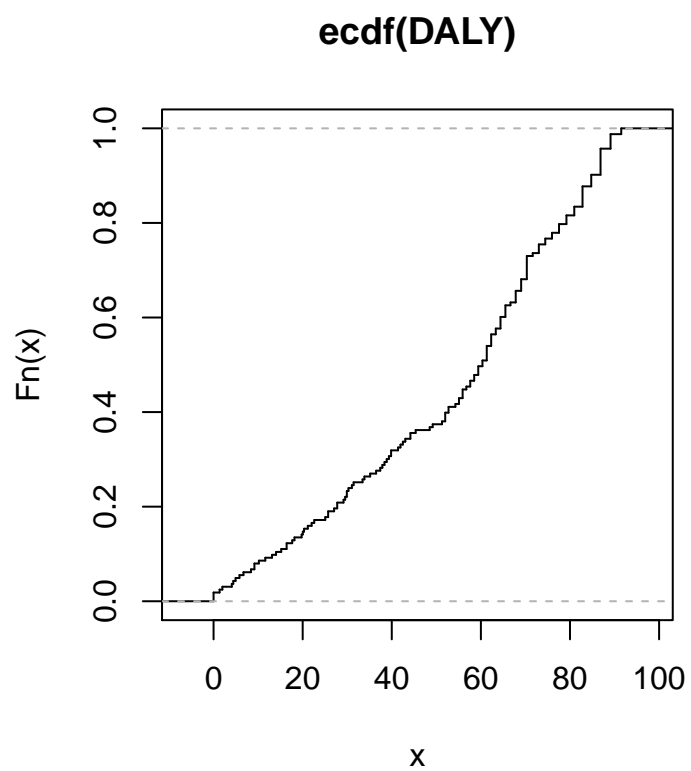
```
boxplot(DALY, WATER_H)
```



```
qqplot(EPI,DALY)
```

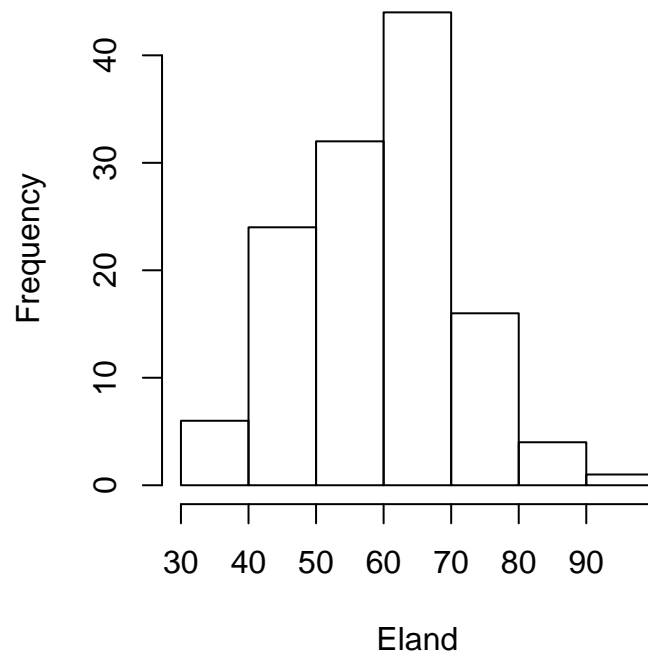


```
plot(ecdf(DALY), do.points=FALSE, verticals=TRUE)
```



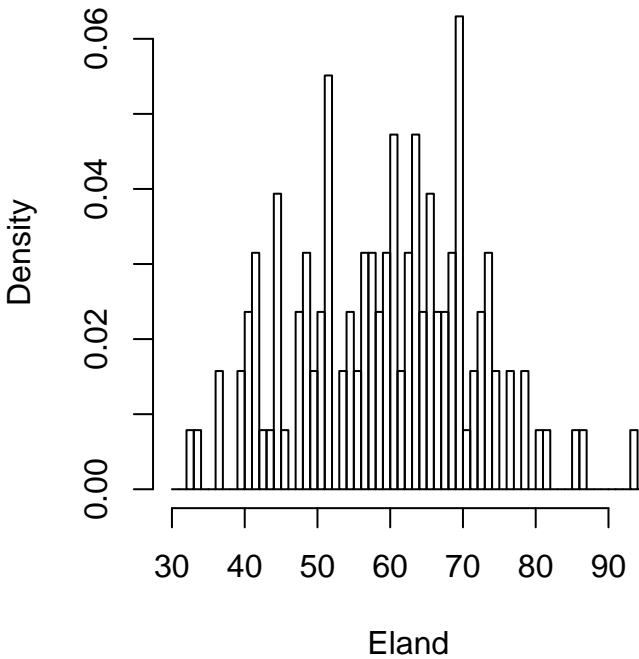
```
help("distributions")  
  
# Exercise 2: filtering(populations)  
EPIland <- EPI[!Landlock]  
Eland <- EPIland[!is.na(EPIland)]  
hist(Eland)
```

Histogram of Eland



```
hist(Eland, seq(30., 95., 1.0), prob=TRUE)
```

Histogram of Eland



#GPW3

View(GPW3)

```
a <- is.na(GPW3)
```

```
b <- GPW3[!a]
```

Exercise 1: exploring the distribution

```
summary(GPW3$NumUnits)
```

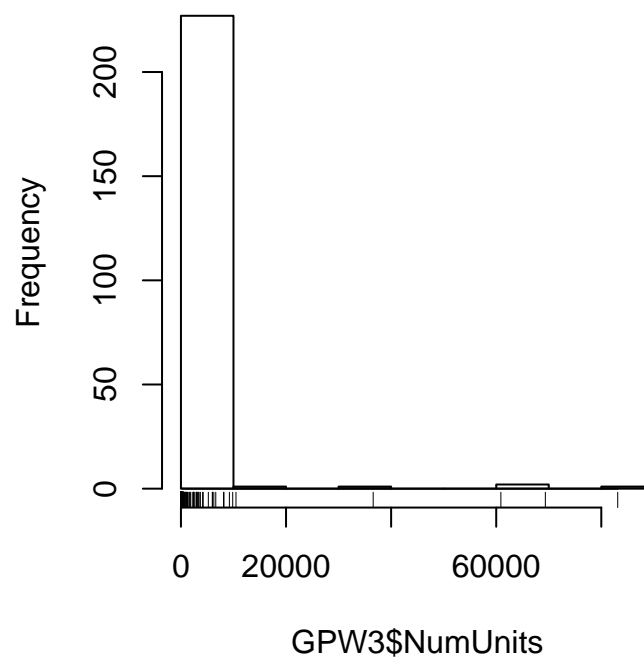
##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
##	1.0	10.0	54.5	1723.0	289.0	83125.0	259

```
stem(GPW3$NumUnits)
```

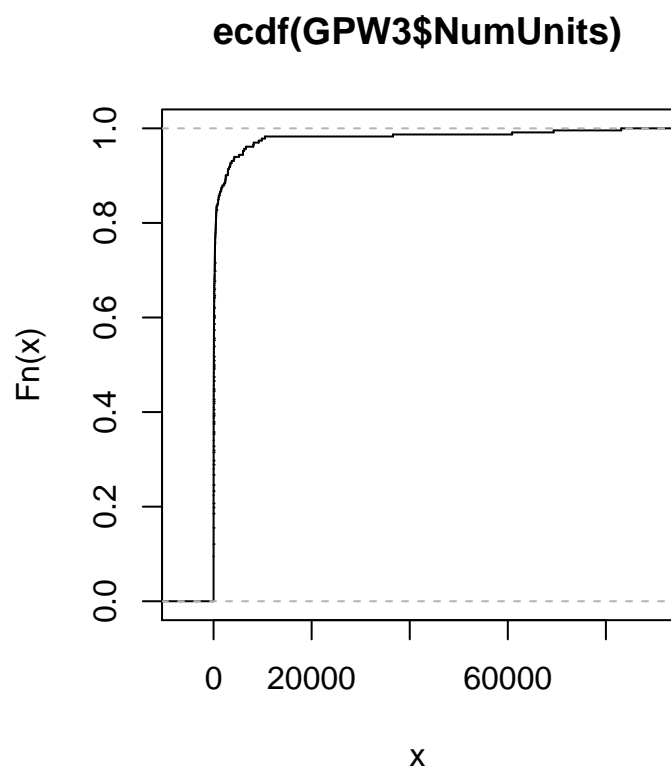
```
## 5 |
## 6 | 1
## 6 | 9
## 7 |
## 7 |
## 8 | 3
```

```
hist(GPW3$NumUnits)
lines(density(GPW3$NumUnits,na.rm=TRUE,bw=1.))
rug(GPW3$NumUnits)
```

Histogram of GPW3\$NumUnits



```
plot(ecdf(GPW3$NumUnits), do.points=FALSE, verticals=TRUE)
```

```
qqnorm(GPW3$NumUnits); qqline(GPW3$NumUnits)
```

Normal Q-Q Plot

