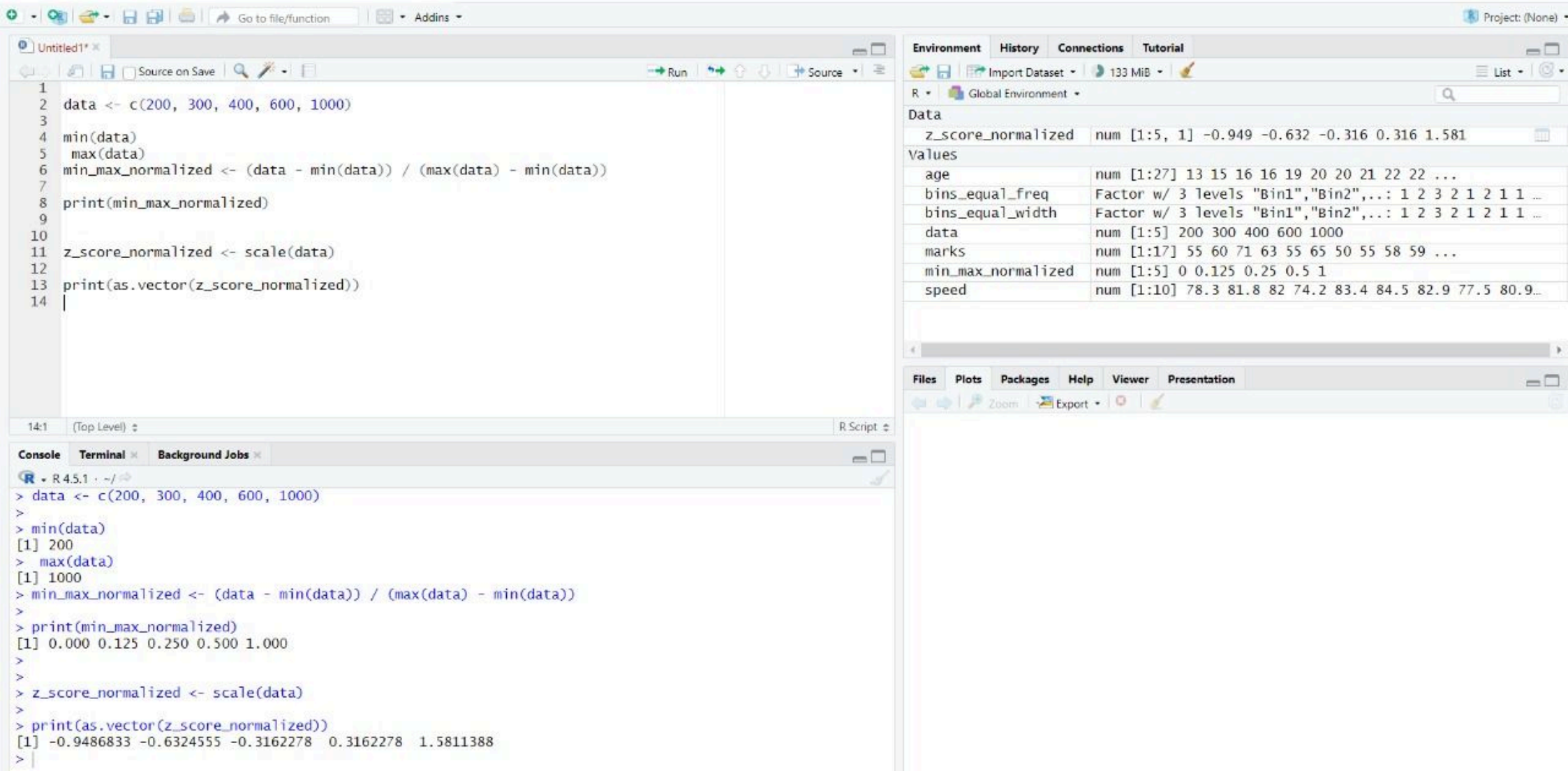

Untitled1*
Source on Save
Run
Source
1 data <- c(200, 300, 400, 600, 1000)
2
3 min(data)
4 max(data)
5 min_max_normalized <- (data - min(data)) / (max(data) - min(data))
6
7 print(min_max_normalized)
8
9
10 z_score_normalized <- scale(data)
11
12 print(as.vector(z_score_normalized))
13
14
14:1 (Top Level) R Script
Console Terminal Background Jobs
R 4.5.1 · ~/
> data <- c(200, 300, 400, 600, 1000)
>
> min(data)
[1] 200
> max(data)
[1] 1000
> min_max_normalized <- (data - min(data)) / (max(data) - min(data))
>
> print(min_max_normalized)
[1] 0.000 0.125 0.250 0.500 1.000
>
>
> z_score_normalized <- scale(data)
>
> print(as.vector(z_score_normalized))
[1] -0.9486833 -0.6324555 -0.3162278 0.3162278 1.5811388
>
Environment History Connections Tutorial
Import Dataset
133 MiB
List
R
Global Environment
Data
z_score_normalized num [1:5, 1] -0.949 -0.632 -0.316 0.316 1.581
Values
age num [1:27] 13 15 16 16 19 20 20 21 22 22 ...
bins_equal_freq Factor w/ 3 levels "Bin1","Bin2",..: 1 2 3 2 1 2 1 1 ...
bins_equal_width Factor w/ 3 levels "Bin1","Bin2",..: 1 2 3 2 1 2 1 1 ...
data num [1:5] 200 300 400 600 1000
marks num [1:17] 55 60 71 63 55 65 50 55 58 59 ...
min_max_normalized num [1:5] 0 0.125 0.25 0.5 1
speed num [1:10] 78.3 81.8 82 74.2 83.4 84.5 82.9 77.5 80.9...
Files Plots Packages Help Viewer Presentation
Zoom
Export

Go to file/function | Addins | Project: (None)

Untitled1*

Source on Save | Run | Source

```
1
2  scores <- c(25, 30, 28, 35, 40, 22, 27, 29, 150, 26, 24, 30, 32)
3
4  boxplot(scores,
5          main = "Boxplot of Tennis Players' Scores",
6          ylab = "Points Scored",
7          col = "lightblue",
8          horizontal = TRUE )
9
```

9:1 (Top Level) R Script

Console | Terminal | Background Jobs

R 4.5.1 · ~/

```
> scores <- c(25, 30, 28, 35, 40, 22, 27, 29, 150, 26, 24, 30, 32)
>
> boxplot(scores,
+         main = "Boxplot of Tennis Players' Scores",
+         ylab = "Points Scored",
+         col = "lightblue",
+         horizontal = TRUE )
>
```

Environment | History | Connections | Tutorial

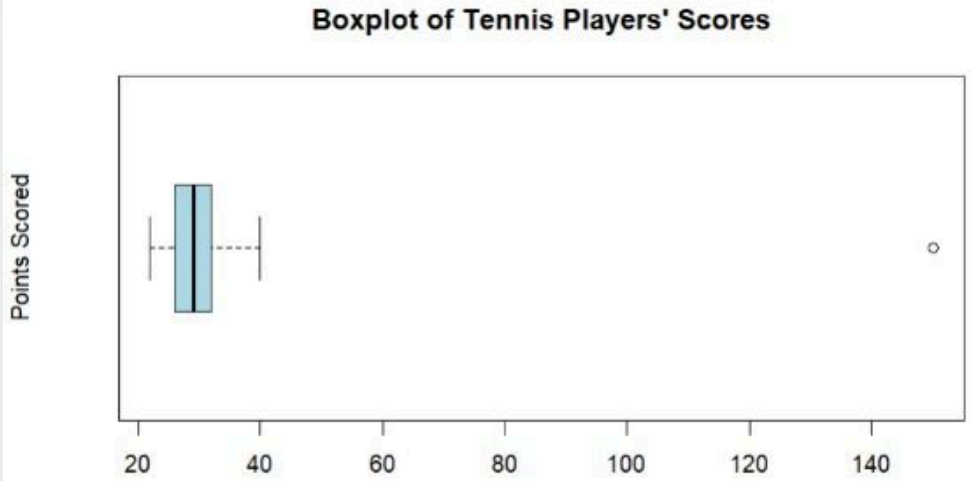
Import Dataset | 134 MiB | List

R | Global Environment

| Data | |
|---|---|
| mtcars | 32 obs. of 11 variables |
| water | 61 obs. of 4 variables |
| **Values** | |
| outliers | 150 |
| scores | num [1:13] 25 30 28 35 40 22 27 29 150 26 ... |

Files | Plots | Packages | Help | Viewer | Presentation

Zoom | Export | Publish



Untitled1*

```r
B<-c(13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33,
     33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70)
hist(B)
mean(B)
median(B)

get_mode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}

get_mode(B)

IQR(B)
quantile(B, 0.75)
summary(B)   # lowercase 'summary'

boxplot(B)
```

20:1 (Top Level) R Script

Console Terminal Background Jobs

R 4.5.1 · ~/

```
> hist(B)
> mean(B)
[1] 29.96296
> median(B)
[1] 25
>
> get_mode <- function(v) {
+   uniqv <- unique(v)
+   uniqv[which.max(tabulate(match(v, uniqv)))]
+ }
>
> get_mode(B)
[1] 25
>
> IQR(B)
[1] 14.5
> quantile(B, 0.75)
75%
 35
> summary(B)   # lowercase 'summary'
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  13.00   20.50   25.00   29.96   35.00   70.00
>
> boxplot(B)
>
```

Environment History Connections Tutorial

Import Dataset 229 MiB List

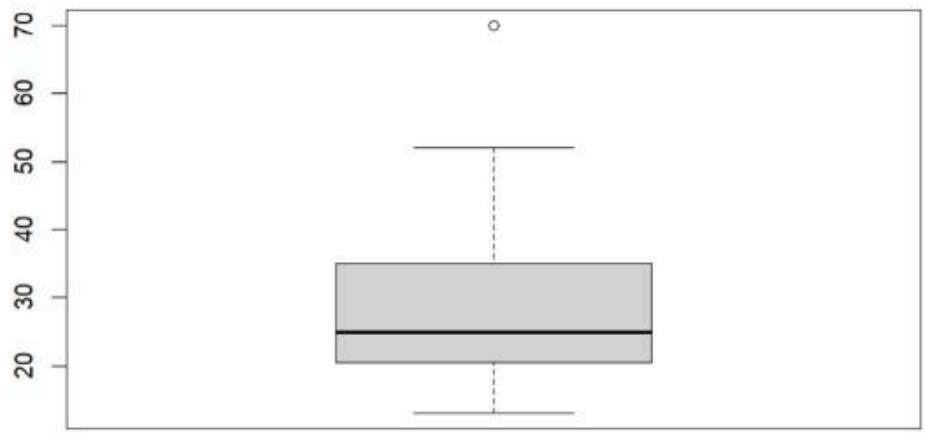
R Global Environment

Values

| | |
|---|---|
| B | num [1:27] 13 15 16 16 19 20 20 21 22 22 ... |
| cum_freq | num [1:6] 200 650 950 2450 3150 ... |
| f | 1500 |
| F | 950 |
| frequencies | num [1:6] 200 450 300 1500 700 44 |
| h | 30 |
| L | 20 |
| lower_bounds | num [1:6] 1 5 15 20 50 80 |
| median | 32.94 |
| median_class_index | 4L |
| N | 3194 |
| upper_bounds | num [1:6] 5 15 20 50 80 110 |

Files Plots Packages Help Viewer Presentation

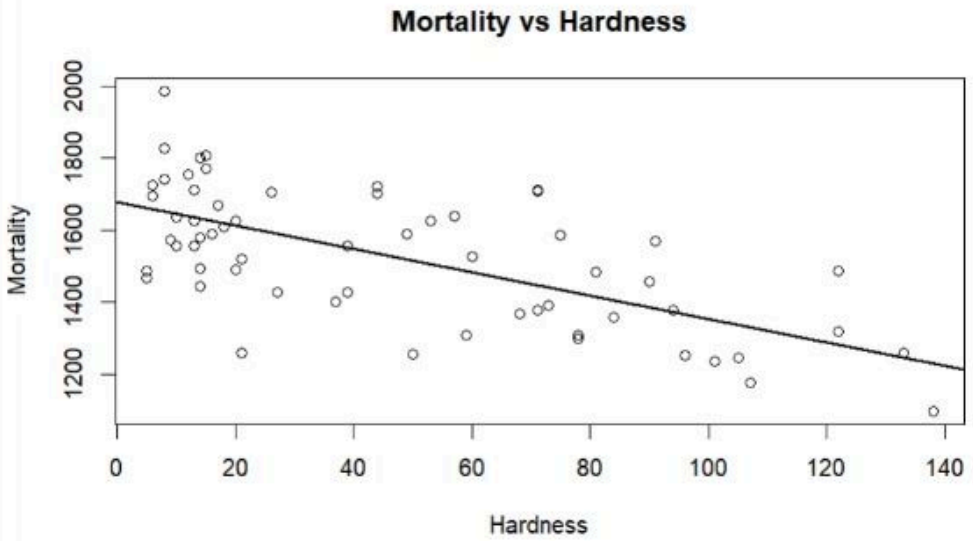
Zoom Export Publish



```r
data("water")

head(water)
plot(water$hardness, water$mortality,
     main = "Mortality vs Hardness",
     xlab = "Hardness",
     ylab = "Mortality",
     )
abline(model, lwd = 2)
predicted_mortality <- predict(model, new_data)
print("Predicted mortality for hardness = 88 is:")
print(predicted_mortality)
```

```
2    North Birkenhead      1668       17
3    South Birmingham      1466        5
4    North  Blackburn      1800       14
5    North  Blackpool      1609       18
6    North     Bolton      1558       10
> plot(water$hardness, water$mortality,
+      main = "Mortality vs Hardness",
+      xlab = "Hardness",
+      ylab = "Mortality",
+
+      )
> abline(model, lwd = 2)
>
> predicted_mortality <- predict(model, new_data)
> print("Predicted mortality for hardness = 88 is:")
[1] "Predicted mortality for hardness = 88 is:"
> print(predicted_mortality)
       1
1392.46
>
```

| Data | |
|---|---|
| data | 32 obs. of 2 variables |
| model | List of 12 |
| new_data | 1 obs. of 1 variable |
| water | 61 obs. of 4 variables |
| **Values** | |
| AirPassengers | Time-Series [1:144] from 1949 to 1961: 112 118 132 129 12… |
| breaks | num [1:5] 100 250 400 550 700 |
| lwd | 2 |
| predicted_mortality | Named num 1392 |

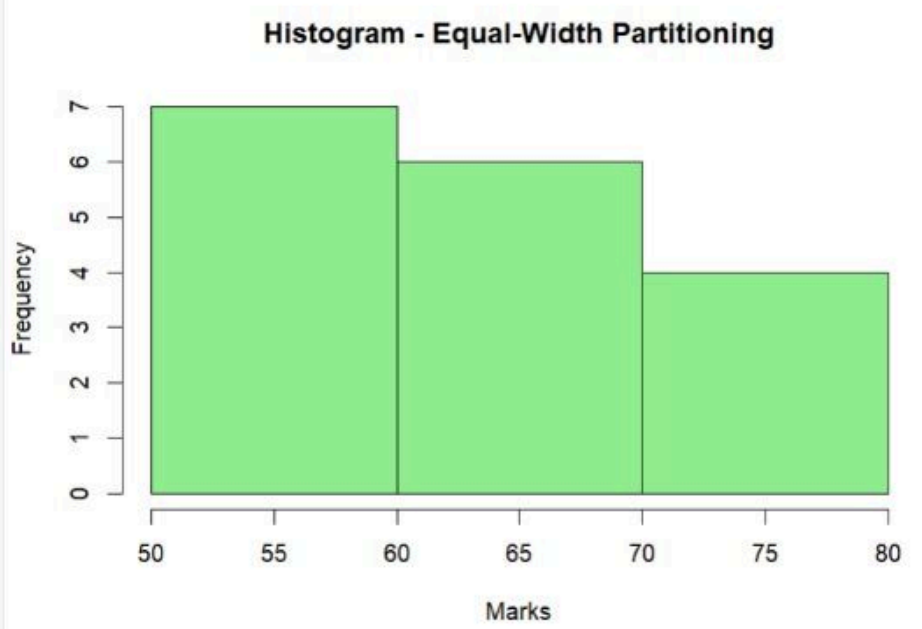


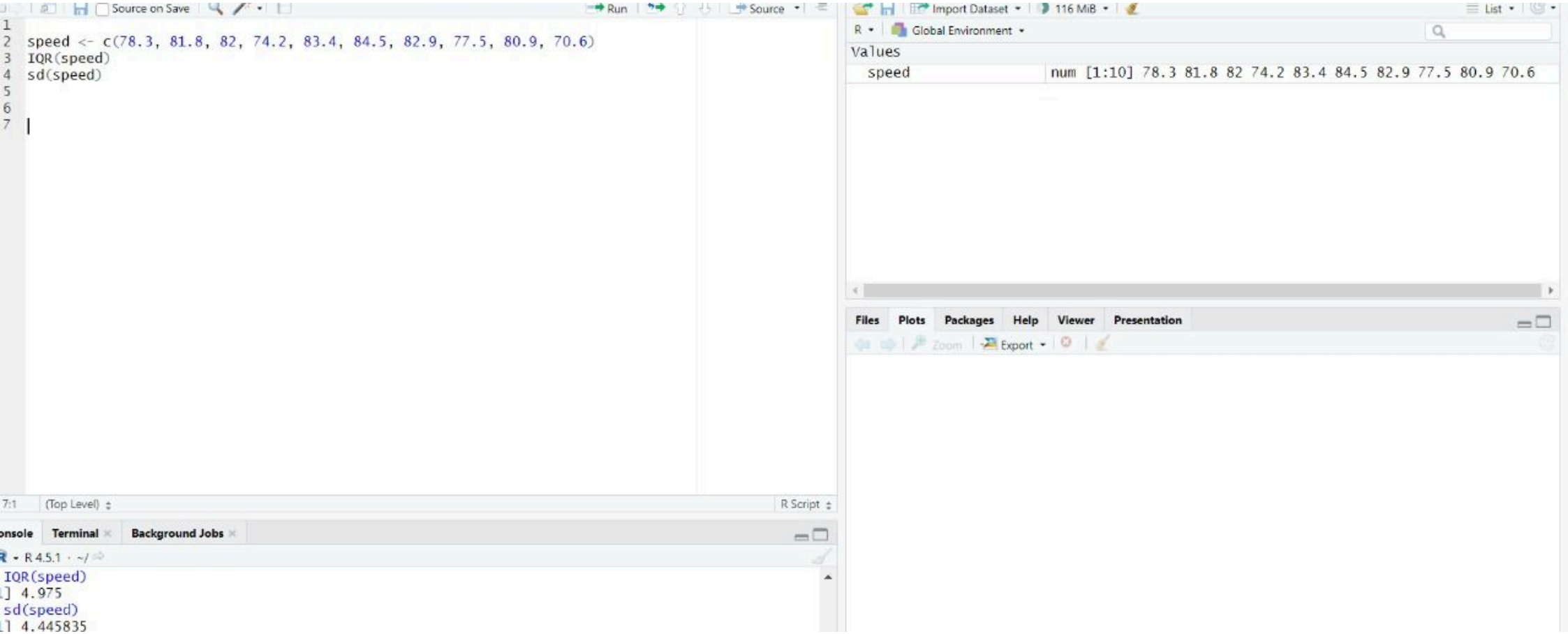
```r
# (a) Equal-frequency (equi-depth) partitioning using quantiles
bins_equal_freq <- cut(marks,
                       breaks = quantile(marks, probs = seq(0, 1, length = 4)),
                       include.lowest = TRUE,
                       labels = c("Bin1", "Bin2", "Bin3"))

# Plot histogram for equal-frequency bins
hist(marks,
     breaks = quantile(marks, probs = seq(0, 1, length = 4)),
     main = "Histogram - Equal-Frequency Partitioning",
     xlab = "Marks",
     col = "lightblue",
     border = "black")

# (b) Equal-width partitioning
bins_equal_width <- cut(marks, breaks = 3, labels = c("Bin1", "Bin2", "Bin3"), include.lowest = TRU

# Plot histogram for equal-width bins
hist(marks,
     breaks = 3,
     main = "Histogram - Equal-Width Partitioning",
     xlab = "Marks",
     col = "lightgreen",
     border = "black")
```

```
+      xlab = "Marks",
+      col = "lightblue",
+      border = "black")
>
> # (b) Equal-width partitioning
> bins_equal_width <- cut(marks, breaks = 3, labels = c("Bin1", "Bin2", "Bin3"), include.lowest = TRUE)
>
> # Plot histogram for equal-width bins
> hist(marks,
+      breaks = 3,
+      main = "Histogram - Equal-Width Partitioning",
+      xlab = "Marks",
+      col = "lightgreen",
+      border = "black")
>
```

Values

| | |
|---|---|
| age | num [1:27] 13 15 16 16 19 20 20 21 22 22 ... |
| bins_equal_freq | Factor w/ 3 levels "Bin1","Bin2",..: 1 2 3 2 1 2 1 1 |
| bins_equal_width | Factor w/ 3 levels "Bin1","Bin2",..: 1 2 3 2 1 2 1 1 |
| marks | num [1:17] 55 60 71 63 55 65 50 55 58 59 ... |
| speed | num [1:10] 78.3 81.8 82 74.2 83.4 84.5 82.9 77.5 80.9 |

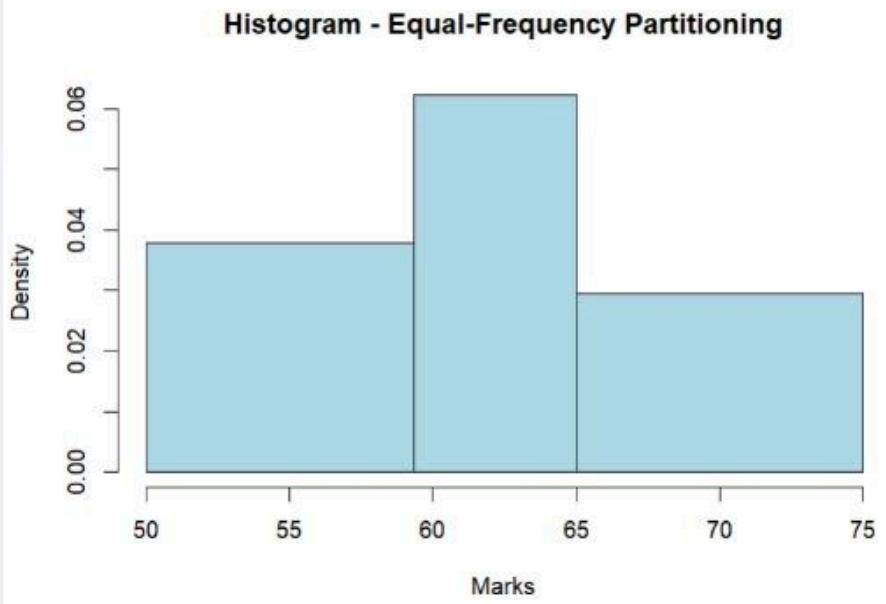



Source on Save
Run
Source
1
2 speed <- c(78.3, 81.8, 82, 74.2, 83.4, 84.5, 82.9, 77.5, 80.9, 70.6)
3 IQR(speed)
4 sd(speed)
5
6
7
7:1
(Top Level)
R Script
Import Dataset
116 MiB
List
R
Global Environment
Values
speed
num [1:10] 78.3 81.8 82 74.2 83.4 84.5 82.9 77.5 80.9 70.6
Files
Plots
Packages
Help
Viewer
Presentation
Zoom
Export
onsole
Terminal
Background Jobs
R 4.5.1 · ~/
IQR(speed)
] 4.975
sd(speed)
] 4.445835

```r
# Given marks
marks <- c(55, 60, 71, 63, 55, 65, 50, 55, 58, 59, 61, 63, 65, 67, 71, 72, 75)

# (a) Equal-frequency (equi-depth) partitioning using quantiles
bins_equal_freq <- cut(marks,
                       breaks = quantile(marks, probs = seq(0, 1, length = 4)),
                       include.lowest = TRUE,
                       labels = c("Bin1", "Bin2", "Bin3"))

# Plot histogram for equal-frequency bins
hist(marks,
     breaks = quantile(marks, probs = seq(0, 1, length = 4)),
     main = "Histogram - Equal-Frequency Partitioning",
     xlab = "Marks",
     col = "lightblue",
     border = "black")

# (b) Equal-width partitioning
bins_equal_width <- cut(marks, breaks = 3, labels = c("Bin1", "Bin2", "Bin3"), include.lowest = TRU

# Plot histogram for equal-width bins
hist(marks,
     breaks = 3,
     main = "Histogram - Equal-Width Partitioning",
     xlab = "Marks",
     col = "lightgreen",
     border = "black")
```

```
> # (a) Equal-frequency (equi-depth) partitioning using quantiles
> bins_equal_freq <- cut(marks,
+                        breaks = quantile(marks, probs = seq(0, 1, length = 4)),
+                        include.lowest = TRUE,
+                        labels = c("Bin1", "Bin2", "Bin3"))
>
> # Plot histogram for equal-frequency bins
> hist(marks,
+      breaks = quantile(marks, probs = seq(0, 1, length = 4)),
+      main = "Histogram - Equal-Frequency Partitioning",
+      xlab = "Marks",
+      col = "lightblue",
+      border = "black")
>
```

| Values | |
|---|---|
| age | num [1:27] 13 15 16 16 19 20 20 21 22 22 ... |
| bins_equal_freq | Factor w/ 3 levels "Bin1","Bin2",..: 1 2 3 2 1 2 1 1 ... |
| bins_equal_width | Factor w/ 3 levels "Bin1","Bin2",..: 1 2 3 2 1 2 1 1 ... |
| marks | num [1:17] 55 60 71 63 55 65 50 55 58 59 ... |
| speed | num [1:10] 78.3 81.8 82 74.2 83.4 84.5 82.9 77.5 80.9... |



Untitled1*

Source on Save | Run | Source

```
1 data("AirPassengers")
2 breaks <- seq(100, 700, by = 150)
3 hist(AirPassengers,
4      breaks = breaks,)
```

4:23 (Top Level) R Script

Console | Terminal | Background Jobs

R 4.5.1 · ~/

>

Environment | History | Connections | Tutorial

Import Dataset | 123 MiB | List

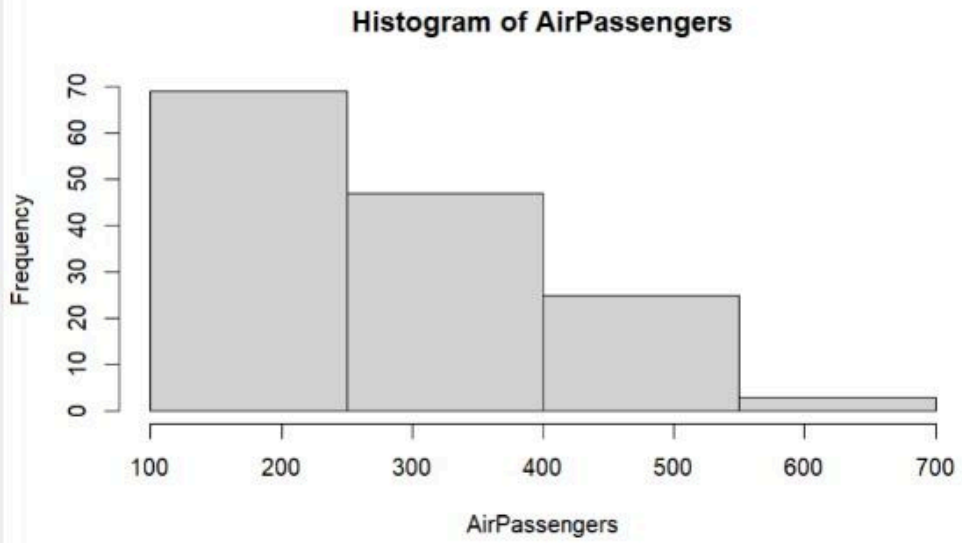
R | Global Environment

Values

| | |
|---|---|
| AirPassengers | Time-Series [1:144] from 1949 to 1961: 112 118 132 129 12… |
| breaks | num [1:5] 100 250 400 550 700 |

Files | Plots | Packages | Help | Viewer | Presentation

Zoom | Export | Publish



```r
# Creating vectors for x and y
x <- c(4, 1, 5, 7, 10, 2, 50, 25, 90, 36)
y <- c(12, 5, 13, 19, 31, 7, 153, 72, 275, 110)

# Creating the scatter plot
plot(x, y,
     main = "Scatter Plot of Mobile Phones Sold vs Money Earned",
     xlab = "Number of Mobile Phones Sold",
     ylab = "Money Earned",
     pch = 19,
     col = "blue")
```

(Top Level) R Script

Terminal Background Jobs

R 4.5.1 · ~/

```r
<- c(4, 1, 5, 7, 10, 2, 50, 25, 90, 36)
<- c(12, 5, 13, 19, 31, 7, 153, 72, 275, 110)

Creating the scatter plot
lot(x, y,
    main = "Scatter Plot of Mobile Phones Sold vs Money Earned",
    xlab = "Number of Mobile Phones Sold",
    ylab = "Money Earned",
    pch = 19,
    col = "blue")
```

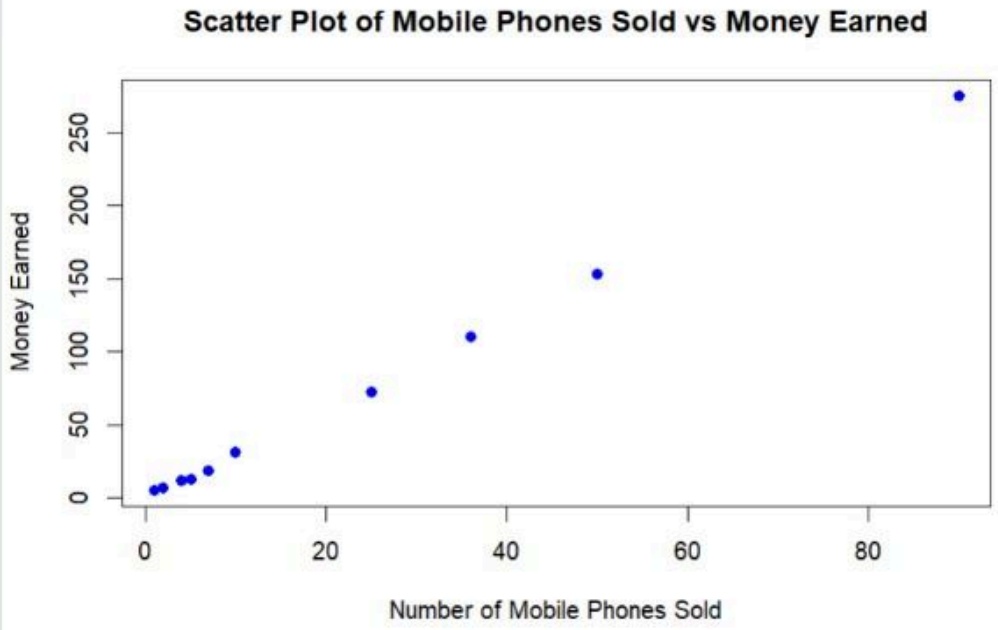
Import Dataset 94 MiB List

R Global Environment

| Values | |
|---|---|
| ages | num [1:18] 23 45 34 54 29 31 39 40 28 25 ... |
| decimal_scaling_norm | 0.35 |
| j | 2 |
| mode_value | 9 |
| pencils | num [1:10] 9 25 23 12 11 6 7 8 9 10 |
| sd_val | 12.94 |
| x | num [1:10] 4 1 5 7 10 2 50 25 90 36 |
| y | num [1:10] 12 5 13 19 31 7 153 72 275 110 |
| **Functions** | |
| get_mode | function (x) |

Files Plots Packages Help Viewer Presentation

Zoom Export Publish

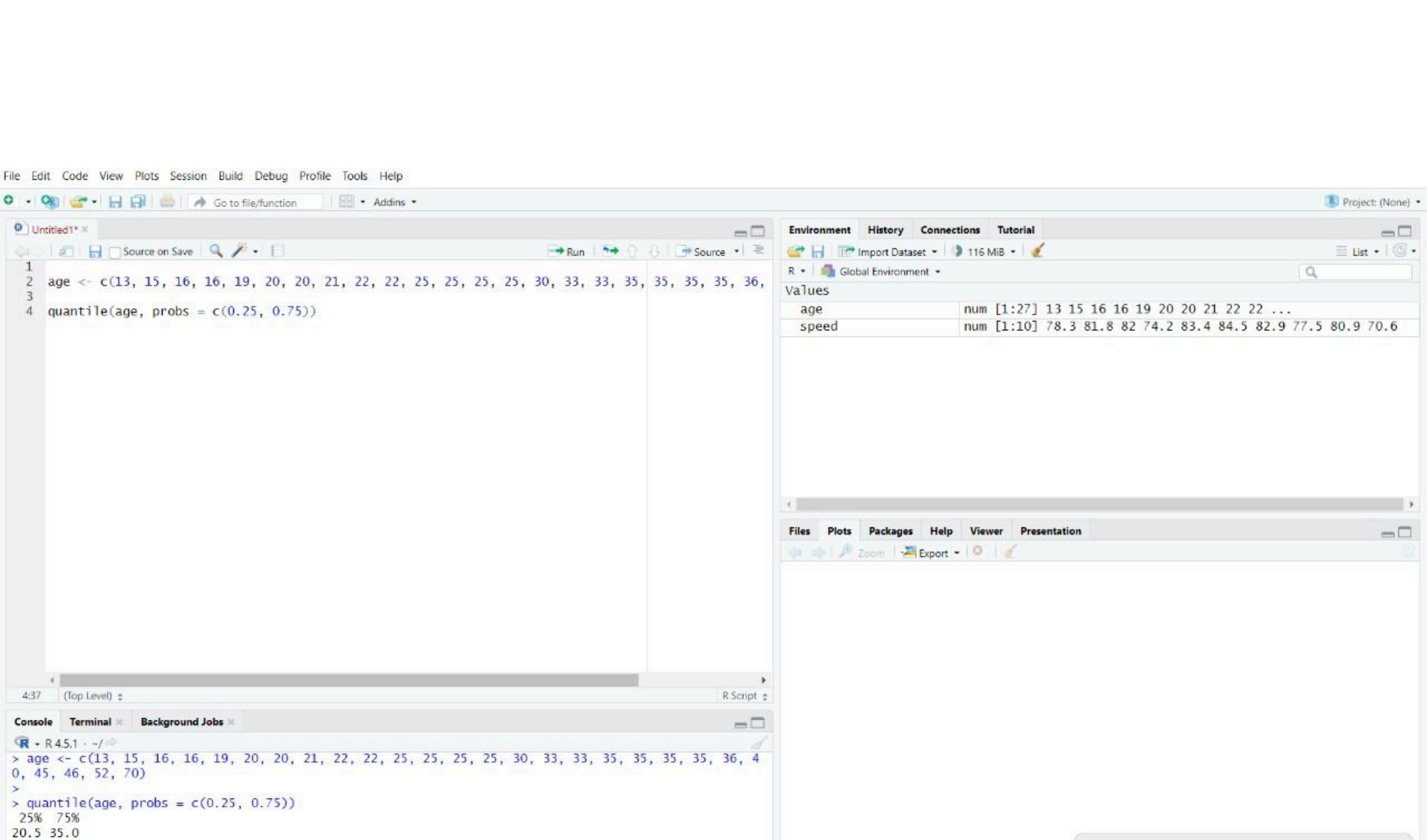


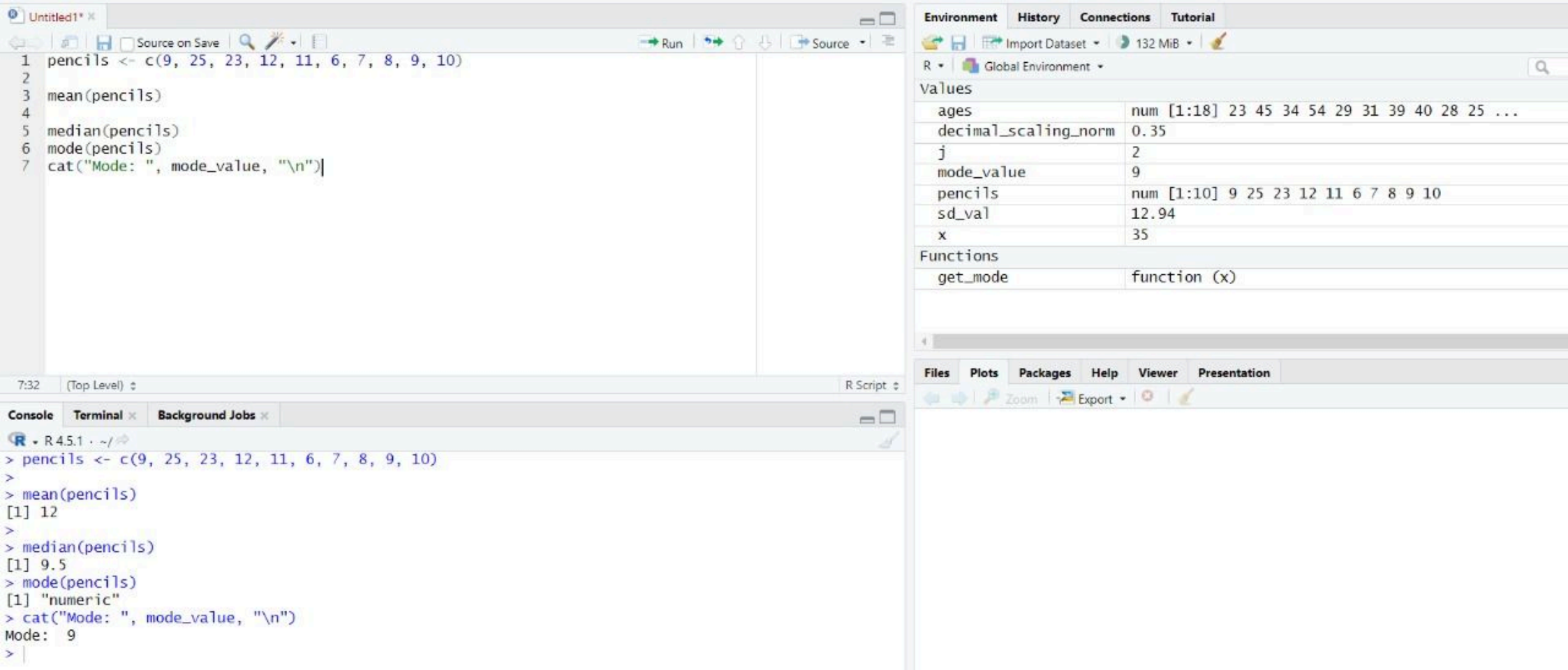

27°C Partly cloudy

Search

ENG IN

11-09-20

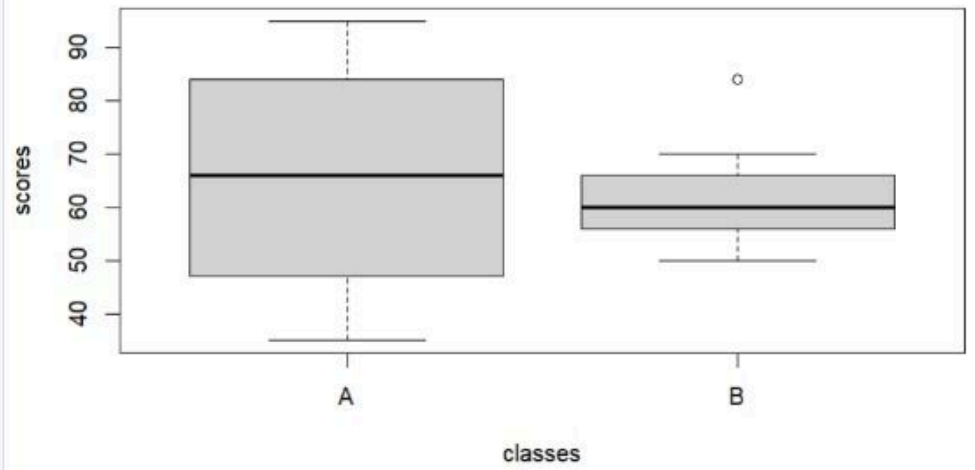

File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Untitled1*
Source on Save
Run
Source
1 age <- c(13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36,
2
3 quantile(age, probs = c(0.25, 0.75))
4
4:37
(Top Level)
R Script
Console
Terminal
Background Jobs
R 4.5.1 · ~/
> age <- c(13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 4
0, 45, 46, 52, 70)
>
> quantile(age, probs = c(0.25, 0.75))
25% 75%
20.5 35.0
Environment
History
Connections
Tutorial
Import Dataset
116 MiB
List
R
Global Environment
Values
age num [1:27] 13 15 16 16 19 20 20 21 22 22 ...
speed num [1:10] 78.3 81.8 82 74.2 83.4 84.5 82.9 77.5 80.9 70.6
Files
Plots
Packages
Help
Viewer
Presentation
Zoom
Export


Untitled1*
Source on Save
Run
Source
1 pencils <- c(9, 25, 23, 12, 11, 6, 7, 8, 9, 10)
2
3 mean(pencils)
4
5 median(pencils)
6 mode(pencils)
7 cat("Mode: ", mode_value, "\n")
7:32
(Top Level)
R Script
Console
Terminal
Background Jobs
R 4.5.1 · ~/
> pencils <- c(9, 25, 23, 12, 11, 6, 7, 8, 9, 10)
>
> mean(pencils)
[1] 12
>
> median(pencils)
[1] 9.5
> mode(pencils)
[1] "numeric"
> cat("Mode: ", mode_value, "\n")
Mode: 9
>
Environment
History
Connections
Tutorial
Import Dataset
132 MiB
R
Global Environment
Values
ages num [1:18] 23 45 34 54 29 31 39 40 28 25 ...
decimal_scaling_norm 0.35
j 2
mode_value 9
pencils num [1:10] 9 25 23 12 11 6 7 8 9 10
sd_val 12.94
x 35
Functions
get_mode function (x)
Files
Plots
Packages
Help
Viewer
Presentation
Zoom
Export

```r
a <- c(76, 35, 47, 64, 95, 66, 89, 36, 84)
b <- c(51, 56, 84, 60, 59, 70, 63, 66, 50)

 mean(a)
 mean(b)
 median(a)
 median(b)

 max(a) - min(a)
 max(b) - min(b)

scores <- c(a,b)
classes <- c(rep("A", length(a)), rep("B", length(b)))
boxplot(scores ~ classes)
```

```
[1] 62.11111
> median(a)
[1] 66
> median(b)
[1] 60
>
> max(a) - min(a)
[1] 60
> max(b) - min(b)
[1] 34
>
> scores <- c(a,b)
> classes <- c(rep("A", length(a)), rep("B", length(b)))
> boxplot(scores ~ classes)
>
```

| Values | |
|---|---|
| a | num [1:9] 76 35 47 64 95 66 89 36 84 |
| b | num [1:9] 51 56 84 60 59 70 63 66 50 |
| classes | chr [1:18] "A" "A" "A" "A" "A" "A" "A" "A" "A" "B" "B" "B... |
| scores | num [1:18] 76 35 47 64 95 66 89 36 84 51 ... |



Untitled1*

Source on Save | Run | Source

```r
ages <- c(23, 45, 34, 54, 29, 31, 39, 40, 28, 25, 36, 38, 33, 50, 44, 30, 27,41)
x <- 35

 min(ages)
 max(ages)
 (x - min(ages) / max(ages) - min(ages))
 mean(ages)
sd_val <- 12.94
 (x - mean(ages)) / sd_val
j <- ceiling(log10(max(abs(x))))

decimal_scaling_norm <- x / (10^j)

cat("Decimal Scaling Normalization of 35:", decimal_scaling_norm, "\n")

```

15:1 (Top Level) R Script

Console | Terminal | Background Jobs

R 4.5.1 · ~/

```r
ages <- c(23, 45, 34, 54, 29, 31, 39, 40, 28, 25, 36, 38, 33, 50, 44, 30, 27,41)
x <- 35

 min(ages)
1] 23
 max(ages)
1] 54
 (x - min(ages) / max(ages) - min(ages))
1] 11.57407
 mean(ages)
1] 35.94444
sd_val <- 12.94
 (x - mean(ages)) / sd_val
1] -0.07298643
j <- ceiling(log10(max(abs(x))))

decimal_scaling_norm <- x / (10^j)

cat("Decimal Scaling Normalization of 35:", decimal_scaling_norm, "\n")
ecimal Scaling Normalization of 35: 0.35
```

Environment | History | Connections | Tutorial

Import Dataset | 138 MiB

R | Global Environment

Values

| | |
|---|---|
| ages | num [1:18] 23 45 34 54 29 31 39 40 28 25 ... |
| decimal_scaling_norm | 0.35 |
| j | 2 |
| sd_val | 12.94 |
| x | 35 |

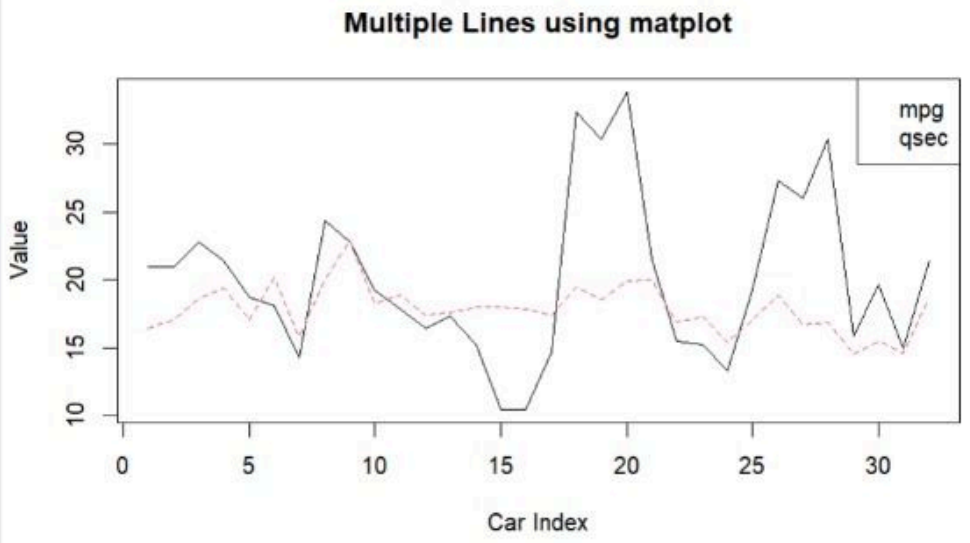
Files | Plots | Packages | Help | Viewer | Presentation

Zoom | Export

```r
data <- mtcars[c("mpg", "qsec")]


matplot(data, type = "l",
        xlab = "Car Index", ylab = "Value",main = "Multiple Lines using matplot" )

legend("topright", legend = c("mpg", "qsec"))
```

```
> legend("topright", legend = c("mpg", "qsec"))
> data <- mtcars[c("mpg", "qsec")]
>
>
>
> matplot(data,
+         xlab = "Car Index", ylab = "Value",main = "Multiple Lines using matplot" )
>
>
> legend("topright", legend = c("mpg", "qsec"))
> data <- mtcars[c("mpg", "qsec")]
>
>
>
> matplot(data, type = "l",
+         xlab = "Car Index", ylab = "Value",main = "Multiple Lines using matplot" )
>
>
> legend("topright", legend = c("mpg", "qsec"))
>
```

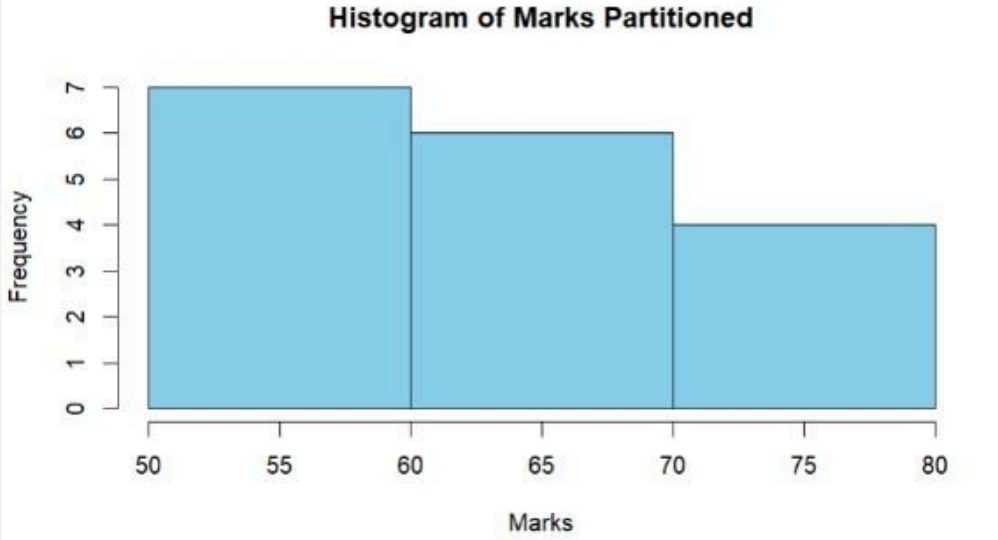
| Data | |
|---|---|
| data | 32 obs. of 2 variables |
| **Values** | |
| AirPassengers | Time-Series [1:144] from 1949 to 1961: 112 118 132 129 12… |
| breaks | num [1:5] 100 250 400 550 700 |



```r
marks <- c(55,60,71,63,55,65,50,55,58,59,61,63,65,67,71,72,75)

sorted_marks <- sort(marks)
bins_eqfreq <- split(sorted_marks, cut(seq_along(sorted_marks), 3, labels=FALSE))

bins_eqfreq

bins_eqwidth <- cut(marks, breaks=3, include.lowest=TRUE)
table(bins_eqwidth)

hist(marks, breaks=3, col="skyblue",
     main="Histogram of Marks Partitioned",
     xlab="Marks", ylab="Frequency", border="black")
```

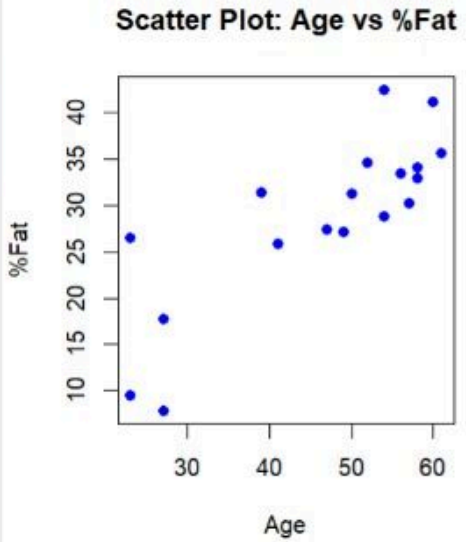
```
$`3`
[1] 65 67 71 71 72 75

>
> bins_eqwidth <- cut(marks, breaks=3, include.lowest=TRUE)
> table(bins_eqwidth)
bins_eqwidth
  [50,58.3] (58.3,66.7]   (66.7,75]
          5           7           5
>
> hist(marks, breaks=3, col="skyblue",
+      main="Histogram of Marks Partitioned",
+      xlab="Marks", ylab="Frequency", border="black")
>
```

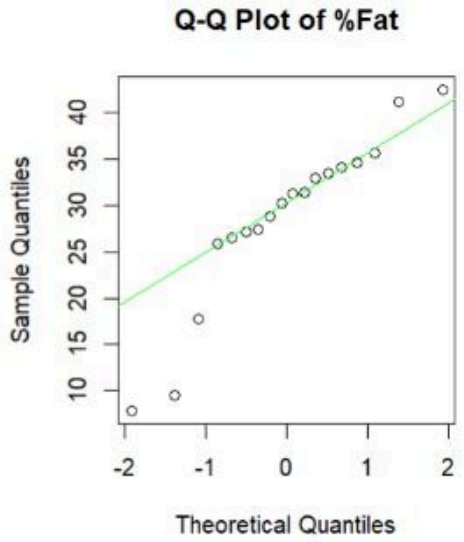
| Data | |
|---|---|
| bins_eqfreq | List of 3 |
| **Values** | |
| bins_eqwidth | Factor w/ 3 levels "[50,58.3]","(58.3,66.7]",..: 1 2 3 2 ... |
| marks | num [1:17] 55 60 71 63 55 65 50 55 58 59 ... |
| mode | "9" |
| p | num [1:11] 9 25 23 12 12 11 6 7 8 9 ... |
| sorted_marks | num [1:17] 50 55 55 55 58 59 60 61 63 63 ... |
| x | num [1:10] 4 1 5 7 10 2 50 25 90 36 |
| y | num [1:10] 12 5 13 19 31 7 153 72 275 110 |



```r
fat <- c(9.5,26.5,7.8,17.8,31.4,25.9,27.4,27.2,31.2,34.6,42.5,28.8,33.4,30.2,34.1,32.9,41.2,3
mean(age)
median(age)
sd(age)
mean(fat)
median(fat)
sd(fat)
# Boxplots
par(mfrow=c(1,2))   # two plots side by side
boxplot(age, main="Boxplot of Age", col="skyblue")
boxplot(fat, main="Boxplot of %Fat", col="red")

# Scatter plot of Age vs %Fat
plot(age, fat, main="Scatter Plot: Age vs %Fat",
     xlab="Age", ylab="%Fat", pch=19, col="blue")

# Q-Q plot for %Fat
qqnorm(fat, main="Q-Q Plot of %Fat")
qqline(fat, col="green")
```

```
> sd(fat)
[1] 9.254395
> # Boxplots
> par(mfrow=c(1,2))   # two plots side by side
> boxplot(age, main="Boxplot of Age", col="skyblue")
> boxplot(fat, main="Boxplot of %Fat", col="red")
>
> # Scatter plot of Age vs %Fat
> plot(age, fat, main="Scatter Plot: Age vs %Fat",
+      xlab="Age", ylab="%Fat", pch=19, col="blue")
>
> # Q-Q plot for %Fat
> qqnorm(fat, main="Q-Q Plot of %Fat")
> qqline(fat, col="green")
>
```



File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function | Addins | Project: (None)

Untitled1* | Untitled2* | Untitled3* | Untitled4* | Untitled5* | R 1.R*

Source on Save | Run | Source

```
1 x<-c(4,1,5,7,10,2,50,25,90,36)
2 y<-c(12,5,13,19,31,7,153,72,275,110)
3 plot(x,y,main="plot:x vs y",xlab="mobiles sold",ylab="money",pch=19,col="blue")
```

3:80 (Top Level) | R Script

Console | Terminal | Background Jobs

R 4.5.1 · ~/

```
> x<-c(4,1,5,7,10,2,50,25,90,36)
> y<-c(12,5,13,19,31,7,153,72,275,110)
> plot(x,y,main="plot:x vs y",xlab="mobiles sold",ylab="money",pch=19,col="blue")
>
```

Environment | History | Connections | Tutorial

Import Dataset | 187 MiB | List

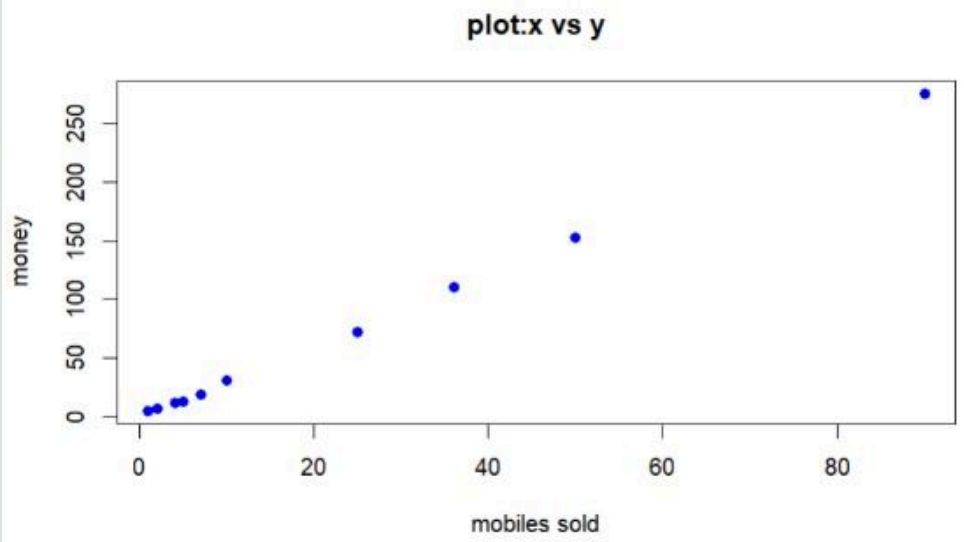
R | Global Environment

Values

| | |
|---|---|
| mode | "9" |
| p | num [1:11] 9 25 23 12 12 11 6 7 8 9 ... |
| x | num [1:10] 4 1 5 7 10 2 50 25 90 36 |
| y | num [1:10] 12 5 13 19 31 7 153 72 275 110 |

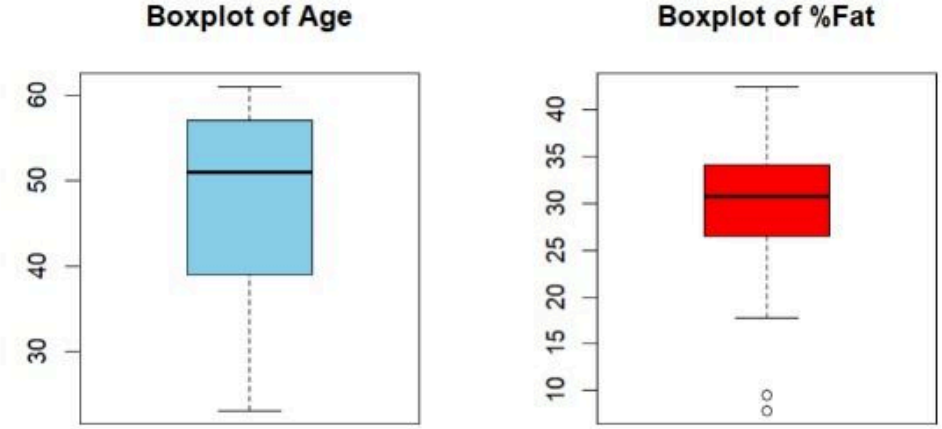
Files | Plots | Packages | Help | Viewer | Presentation

Zoom | Export | Publish



```r
fat <- c(9.5,26.5,7.8,17.8,31.4,25.9,27.4,27.2,31.2,34.6,42.5,28.8,33.4,30.2,34.1,32.9,41.2,35

mean(age)
median(age)
sd(age)
mean(fat)
median(fat)
sd(fat)
# Boxplots
par(mfrow=c(1,2))   # two plots side by side
boxplot(age, main="Boxplot of Age", col="skyblue")
boxplot(fat, main="Boxplot of %Fat", col="red")

# Scatter plot of Age vs %Fat
plot(age, fat, main="Scatter Plot: Age vs %Fat",
     xlab="Age", ylab="%Fat", pch=19, col="blue")

# Q-Q plot for %Fat
qqnorm(fat, main="Q-Q Plot of %Fat")
qqline(fat, col="green")
```

```
> sd(fat)
[1] 9.254395
> # Boxplots
> par(mfrow=c(1,2))   # two plots side by side
> boxplot(age, main="Boxplot of Age", col="skyblue")
> boxplot(fat, main="Boxplot of %Fat", col="red")
>
> # Scatter plot of Age vs %Fat
> plot(age, fat, main="Scatter Plot: Age vs %Fat",
+      xlab="Age", ylab="%Fat", pch=19, col="blue")
>
> # Q-Q plot for %Fat
> qqnorm(fat, main="Q-Q Plot of %Fat")
> qqline(fat, col="green")
>
```



```r
age <- c(23,23,27,27,39,41,47,49,50,52,54,54,56,57,58,58,60,61)
x <- 35

min_max <- (x - min(age)) / (max(age) - min(age))

# (ii) Z-score Normalization
mean_age <- mean(age)
sd_age <- 12.94   # given
z_score <- (x - mean_age) / sd_age

j <- ceiling(log10(max(abs(age))))
decimal_scaling <- x / (10^j)

min_max
z_score
decimal_scaling
```

18:1 (Top Level) R Script

Console Terminal Background Jobs

R 4.5.1 · ~/

```
> decimal_scaling
Error: object 'decimal_scaling' not found

> j <- ceiling(log10(max(abs(age))))
> decimal_scaling <- x / (10^j)
>
> min_max
[1] 0.000 0.125 0.250 0.500 1.000
> z_score
[1] -0.9486833 -0.6324555 -0.3162278  0.3162278  1.5811388
> decimal_scaling
[1]  2  3  4  6 10
>
```

Show Traceback
Rerun with Debug

Connection Status

Files Plots Packages Help Viewer Presentation

Zoom Export

```
data(mtcars)
boxplot(mpg ~ as.factor(cyl), data = mtcars,
        main = "Boxplot of MPG by Number of Cylinders",
        xlab = "Number of Cylinders",
        ylab = "Miles Per Gallon (mpg)",
        col = c("lightblue", "lightgreen", "lightpink"))
```

```
> data(mtcars)
> boxplot(mpg ~ as.factor(cyl), data = mtcars,
+         main = "Boxplot of MPG by Number of Cylinders",
+         xlab = "Number of Cylinders",
+         ylab = "Miles Per Gallon (mpg)",
+         col = c("lightblue", "lightgreen", "lightpink"))
>
```

| Data | |
|---|---|
| mtcars | 32 obs. of 11 variables |
| water | 61 obs. of 4 variables |