project

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
#data processing
#load the data; load the data in R
#Statistical Analysis
#data processing
#load the data; load the data in R
library(readxl)
library(dplyr)
```

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

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

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

```
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------------------------------------

## v ggplot2 3.2.1     v readr   1.3.1
## v tibble  2.1.3     v purrr   0.3.2
## v tidyr   1.0.0     v stringr 1.4.0
## v ggplot2 3.2.1     v forcats 0.4.0

## -- Conflicts ---------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
# xlsx files

my_data2 <- read_excel("stat_data2.xlsx")
```

```
## New names:
## * Index -> Index...2
## * `` -> ...5
## * `` -> ...7
## * Index -> Index...15
```

```
head(my_data2)
```

```
## # A tibble: 6 x 20
##   Pet   Index...2 PHappyScore PHealthyScore ...5  Ptime ...7  PMonthlyMoney
##   <lgl>     <dbl>       <dbl>         <dbl> <lgl> <dbl> <lgl>         <dbl>
## 1 NA            1           6             8 NA      1.3 NA              100
## 2 NA            2           8             8 NA      1.5 NA              150
## 3 NA            3           8             8 NA      1   NA              120
## 4 NA            4           7             6 NA      0.5 NA               80
## 5 NA            5          10             8 NA      2   NA              150
## 6 NA            6           8             9 NA      2   NA              200
## # ... with 12 more variables: PGenderIndex <dbl>, PGender <chr>,
## #   PAge <dbl>, PetTypeIndex <dbl>, PetType <chr>, `No Pet` <lgl>,
## #   Index...15 <dbl>, NHappyScore <dbl>, NHealthyScore <dbl>,
## #   NGenderIndex <dbl>, NGender <chr>, NAge <dbl>
```

```
summary(my_data2)
```

```
##     Pet            Index...2      PHappyScore     PHealthyScore
##  Mode:logical   Min.   : 1.00   Min.   : 5.000   Min.   : 4.000
##  NA's:38        1st Qu.:10.25   1st Qu.: 7.000   1st Qu.: 7.000
##                 Median :19.50   Median : 8.000   Median : 7.000
##                 Mean   :19.50   Mean   : 7.658   Mean   : 7.368
##                 3rd Qu.:28.75   3rd Qu.: 8.000   3rd Qu.: 8.000
##                 Max.   :38.00   Max.   :10.000   Max.   :10.000
##
##     ...5             Ptime         ...7         PMonthlyMoney
##  Mode:logical   Min.   :0.100   Mode:logical   Min.   :  15.0
##  NA's:38        1st Qu.:0.500   NA's:38        1st Qu.: 100.0
##                 Median :1.000                  Median : 177.5
##                 Mean   :1.033                  Mean   : 194.9
##                 3rd Qu.:1.275                  3rd Qu.: 200.0
##                 Max.   :3.000                  Max.   :1000.0
##
##   PGenderIndex     PGender               PAge        PetTypeIndex
##  Min.   :1.000   Length:38          Min.   :23.00   Min.   :1.000
##  1st Qu.:1.000   Class  :character   1st Qu.:25.00   1st Qu.:1.000
##  Median :2.000   Mode   :character   Median :28.00   Median :1.000
##  Mean   :1.579                      Mean   :31.39   Mean   :1.289
##  3rd Qu.:2.000                      3rd Qu.:34.00   3rd Qu.:2.000
##  Max.   :2.000                      Max.   :69.00   Max.   :2.000
##
##    PetType           No Pet         Index...15  NHappyScore
##  Length:38          Mode:logical   Min.   : 1   Min.   :6.000
##  Class :character   NA's:38        1st Qu.: 5   1st Qu.:7.000
##  Mode  :character                  Median : 9   Median :8.000
##                                    Mean   : 9   Mean   :7.647
##                                    3rd Qu.:13   3rd Qu.:8.000
##                                    Max.   :17   Max.   :9.000
##                                    NA's   :21   NA's   :21
##  NHealthyScore     NGenderIndex     NGender               NAge
##  Min.   : 5.000   Min.   :1.000   Length:38          Min.   :21.00
##  1st Qu.: 7.000   1st Qu.:1.000   Class :character   1st Qu.:24.00
##  Median : 8.000   Median :1.000   Mode  :character   Median :27.00
##  Mean   : 7.765   Mean   :1.471                      Mean   :27.35
##  3rd Qu.: 8.000   3rd Qu.:2.000                      3rd Qu.:29.00
##  Max.   :10.000   Max.   :2.000                      Max.   :40.00
##  NA's   :21       NA's   :21                         NA's   :21
```

```r
#Exploratory Analysis
# sample size
n1 <- length(my_data2$PHappyScore)
n1
```

```
## [1] 38
```

```r
n2 <- length(my_data2$PHealthyScore)
n2
```

```
## [1] 38
```

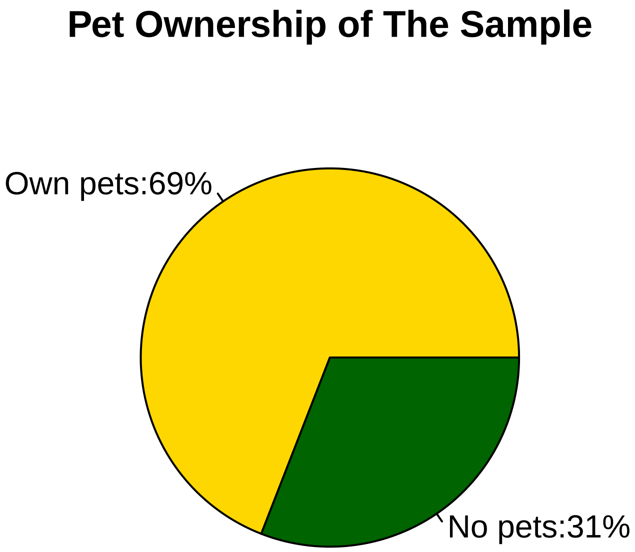
```r
n3 <- length(my_data2$NHappyScore[!is.na(my_data2$NHappyScore)])
n3
```

```
## [1] 17
```

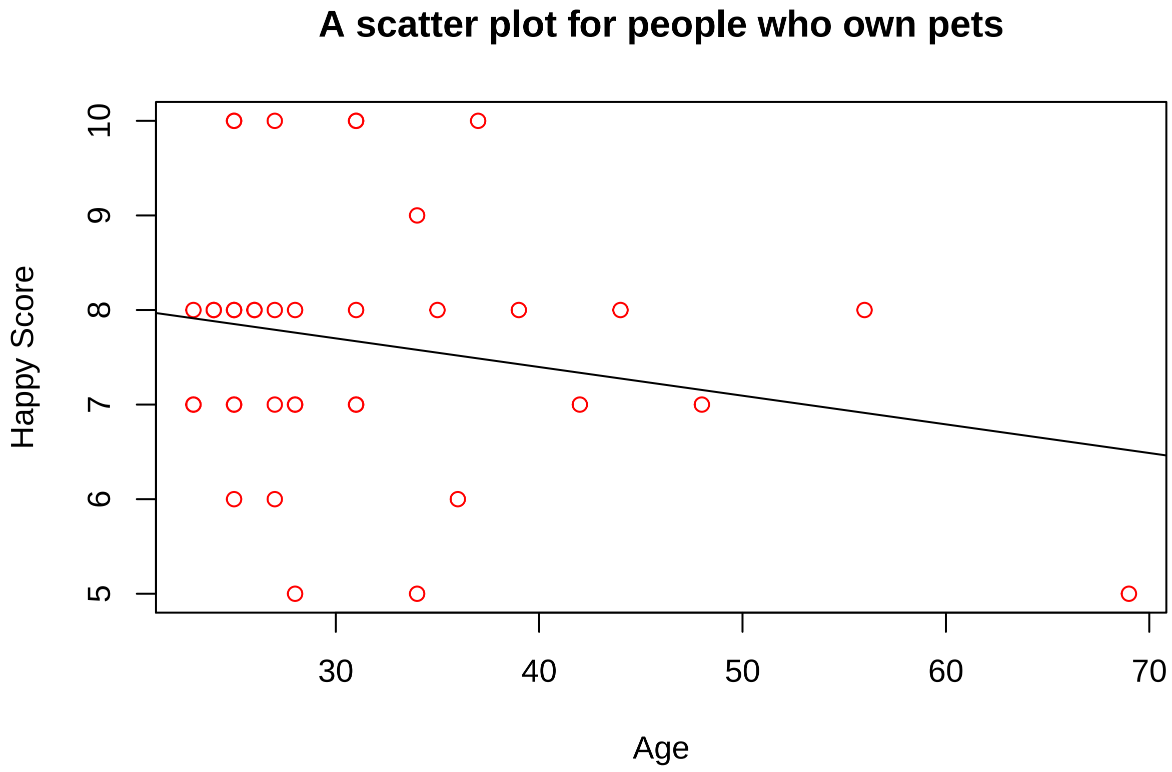
```r
n4 <- length(my_data2$NHealthyScore[!is.na(my_data2$NHealthyScore)])
n4
```

```
## [1] 17
```

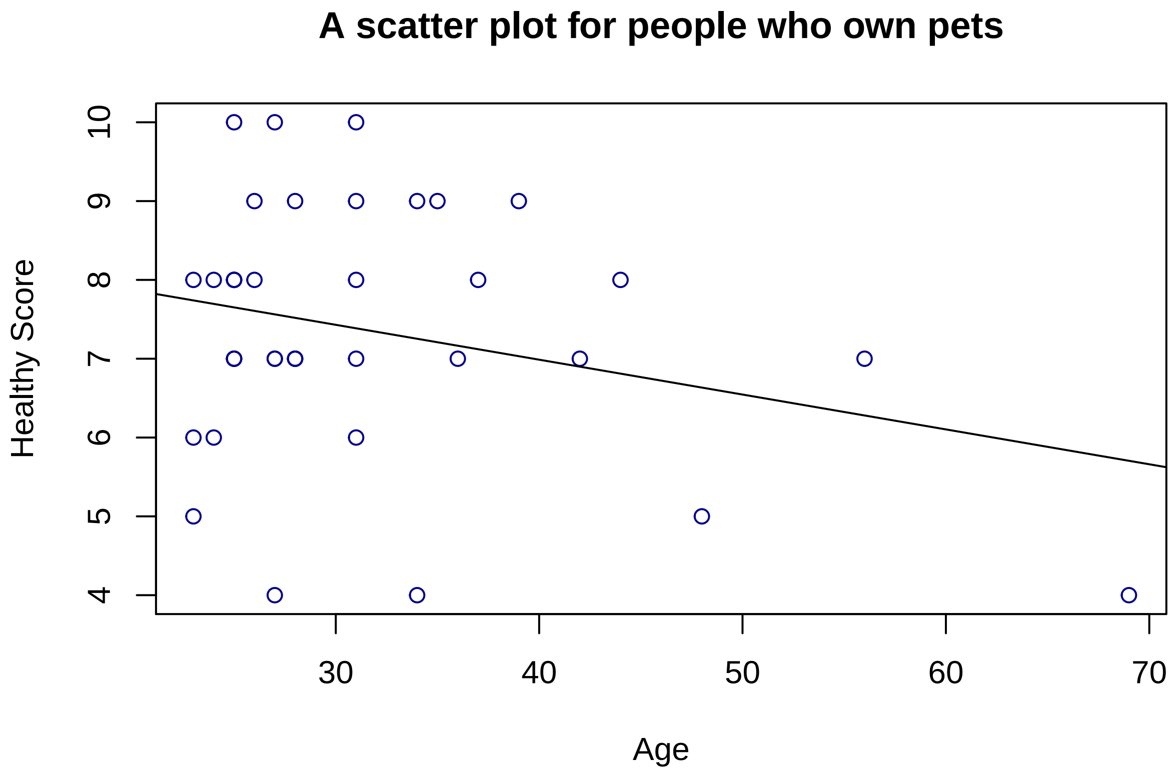
```r
# the pie chart
counts <- c(38,17)
lbls <- c("Own pets","No pets")
label <- paste(lbls,":",round(counts/sum(counts)*100), "%", sep="")
pie(counts,labels= label, col=c("gold","darkgreen"),
    main="Pet Ownership of The Sample")
```



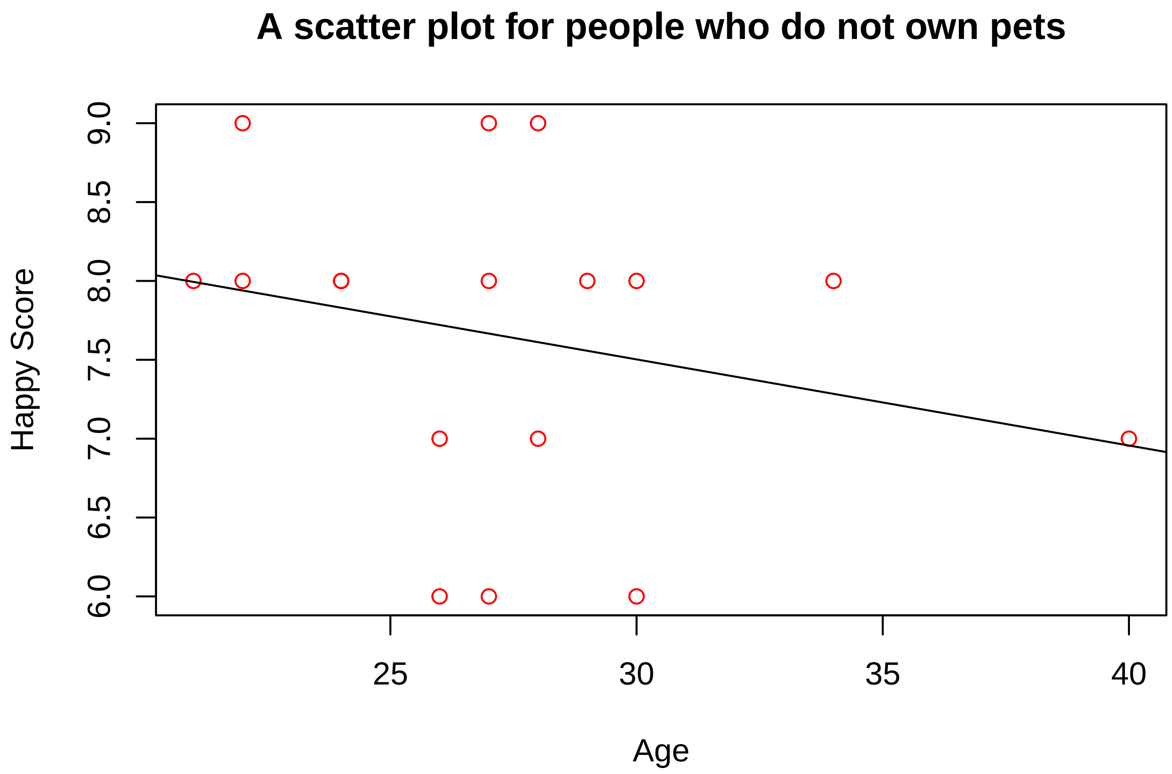
```r
#scatter plot
plot (my_data2$PAge,my_data2$PHappyScore,main="A scatter plot for people who own pets",
      xlab="Age", ylab="Happy Score", col='red')
abline(lm(my_data2$PHappyScore~my_data2$PAge ))
```



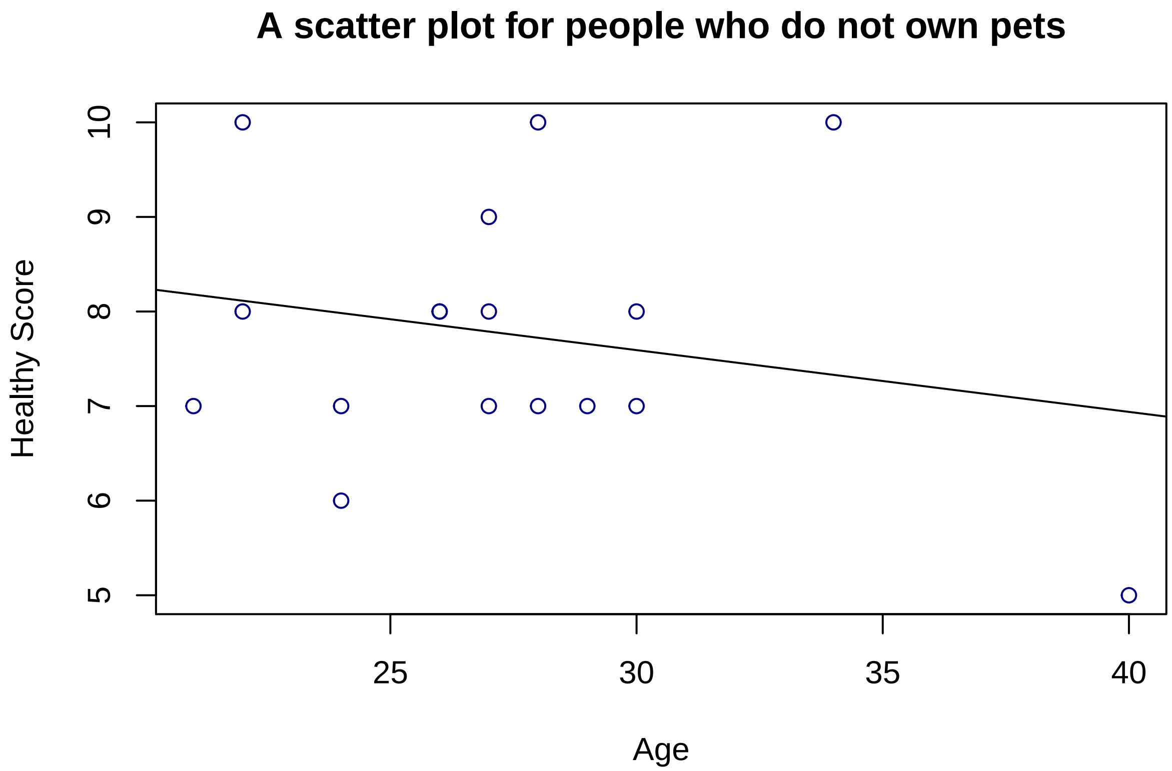
```
plot(my_data2$PAge,my_data2$PHealthyScore,main="A scatter plot for people who own pets",
     xlab="Age", ylab="Healthy Score", col='darkblue')
abline(lm(my_data2$PHealthyScore~my_data2$PAge))
```



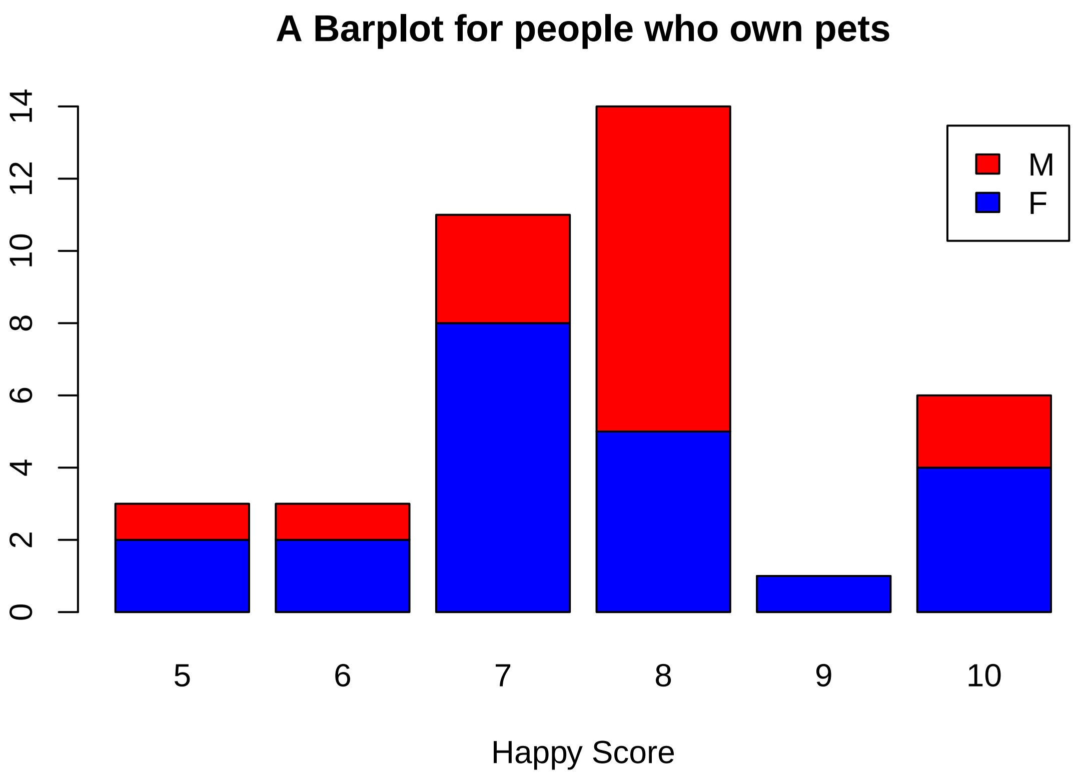
```
plot(my_data2$NAge,my_data2$NHappyScore,main="A scatter plot for people who do not own pets",
     xlab="Age", ylab="Happy Score", col='red')
abline(lm(my_data2$NHappyScore~my_data2$NAge))
```



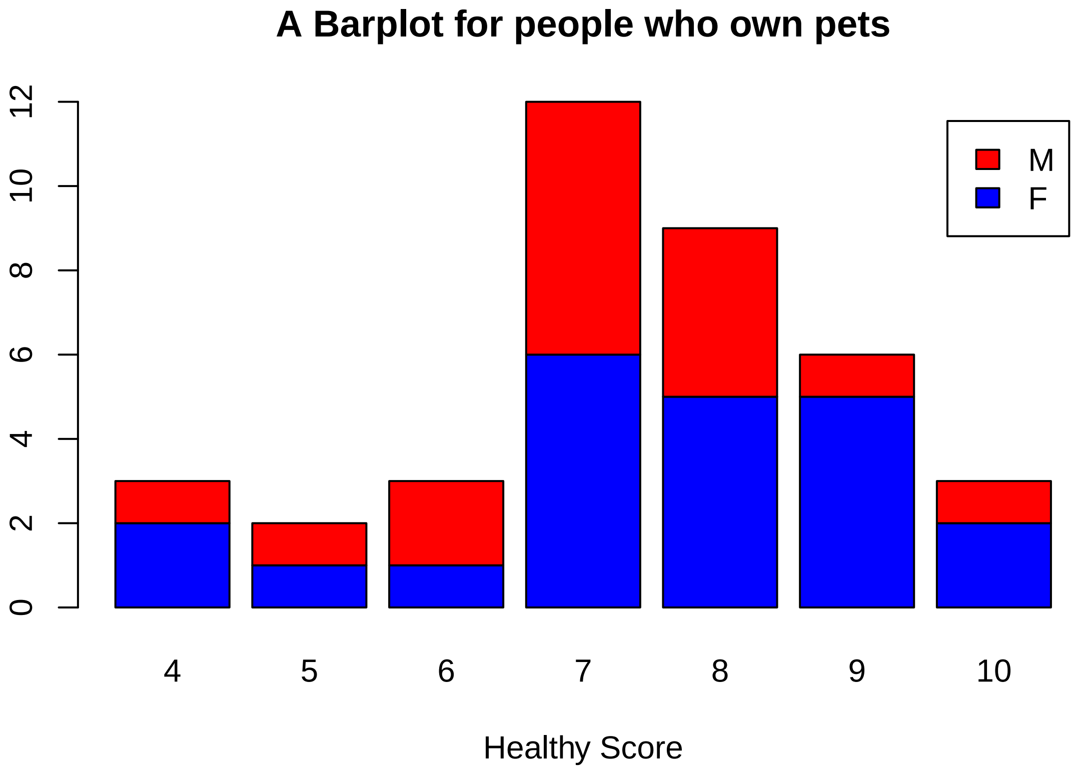
```
plot(my_data2$NAge,my_data2$NHealthyScore,main="A scatter plot for people who do not own pets",
     xlab="Age", ylab="Healthy Score", col='darkblue')
abline(lm(my_data2$NHealthyScore~my_data2$NAge))
```



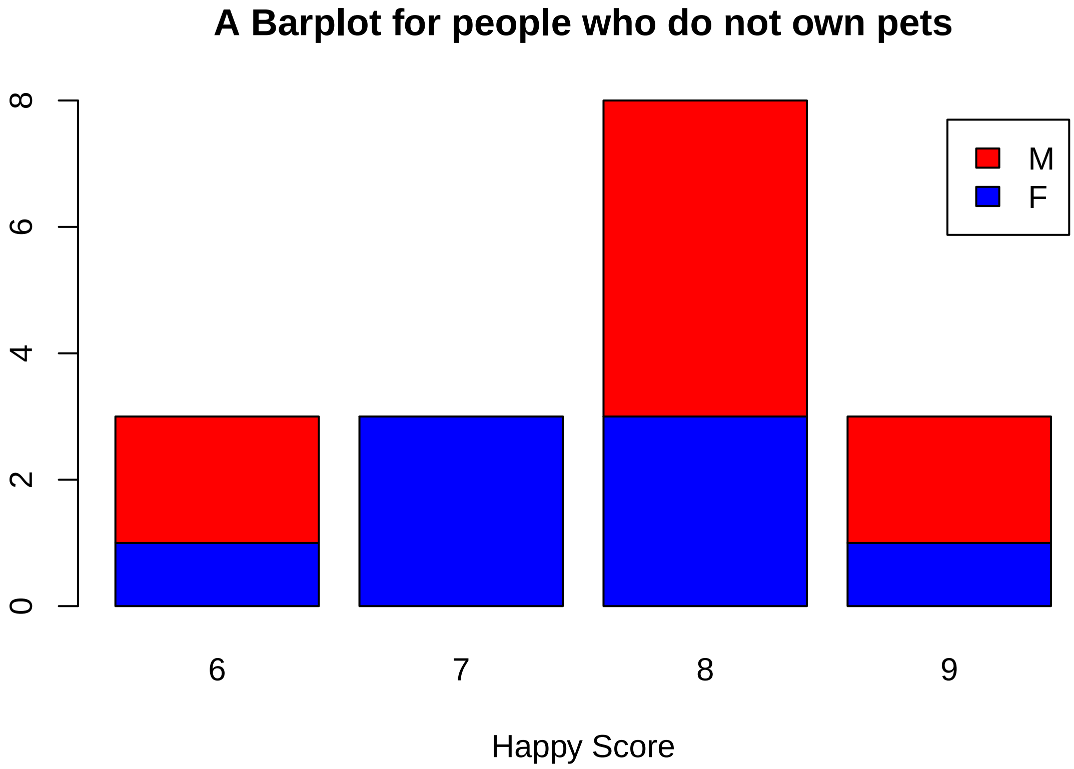
```
#barplot
#Gender vs Scores
counts1 <- table(my_data2$PGender,my_data2$PHappyScore)
barplot(counts1,main=" A Barplot for people who own pets ",
  xlab="Happy Score",
  col=c("blue","red"),
  legend = rownames(counts1))
```



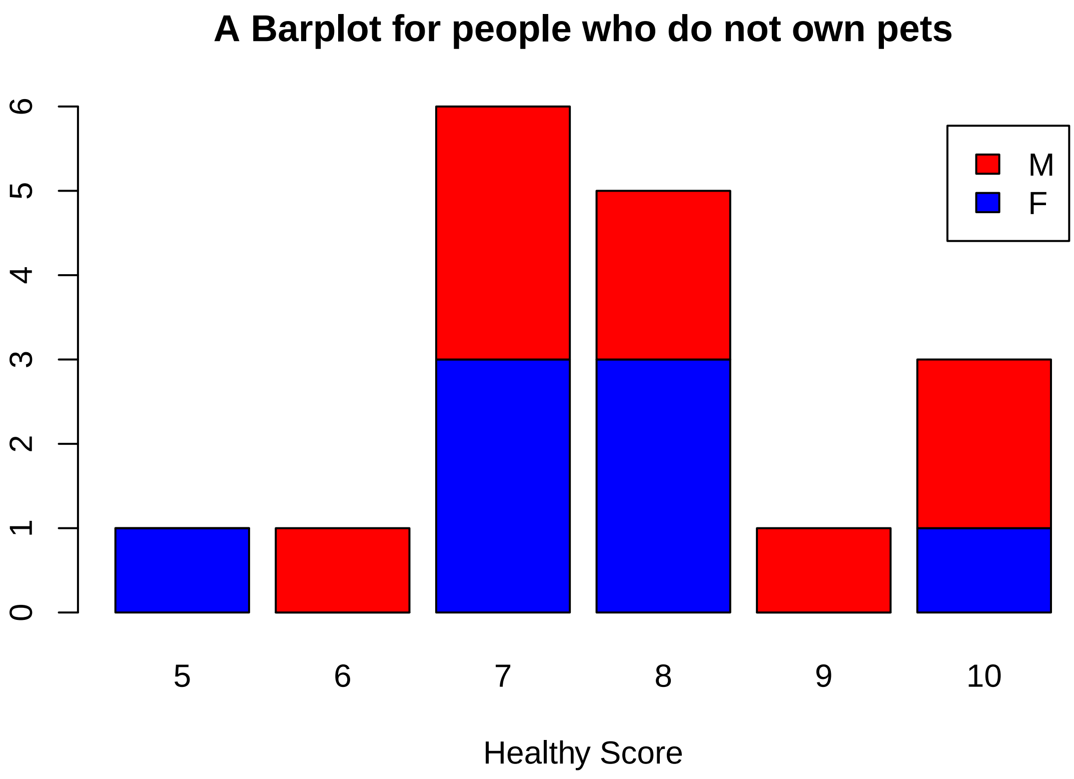
```
counts2 <- table(my_data2$PGender,my_data2$PHealthyScore)
barplot(counts2,main=" A Barplot for people who own pets ",
  xlab="Healthy Score",
  col=c("blue","red"),
  legend = rownames(counts2))
```



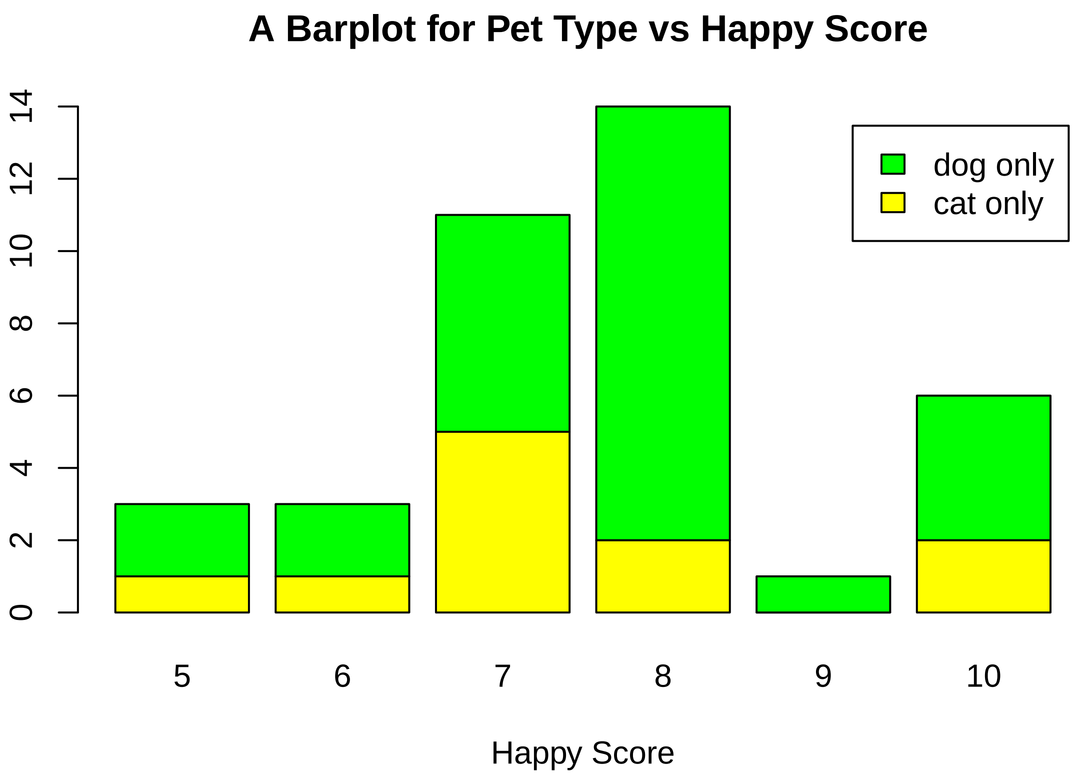
```
counts3 <- table(my_data2$NGender,my_data2$NHappyScore)
barplot(counts3,main=" A Barplot for people who do not own pets ",
  xlab="Happy Score",
  col=c("blue","red"),
  legend = rownames(counts3))
```



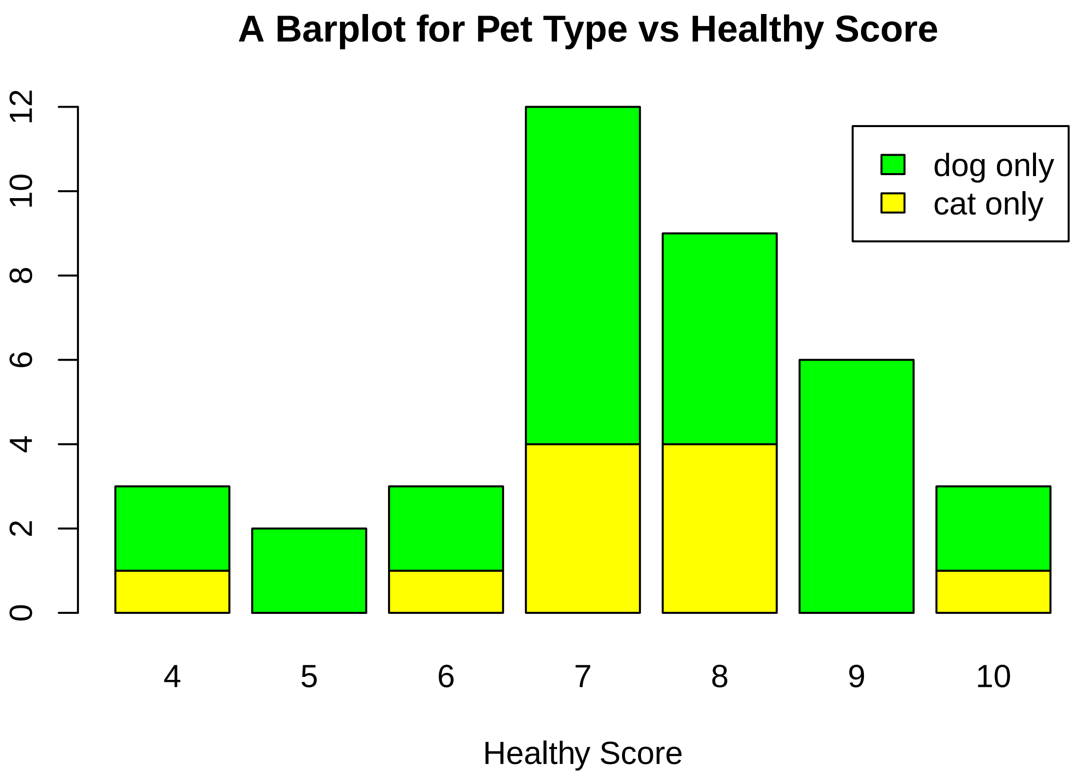
```
counts4 <- table(my_data2$NGender,my_data2$NHealthyScore)
barplot(counts4,main=" A Barplot for people who do not own pets ",
  xlab="Healthy Score",
  col=c("blue","red"),
  legend = rownames(counts4))
```



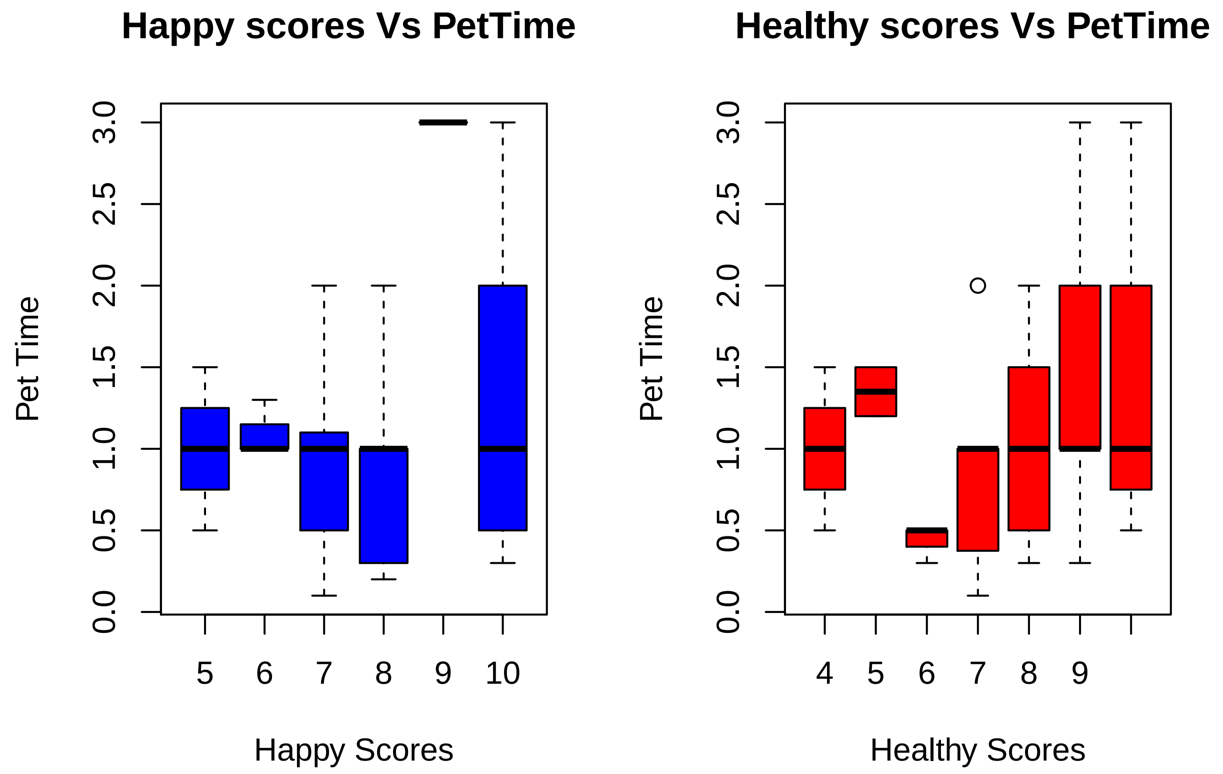
```
#Pet Type Vs Scores
counts5 <- table(my_data2$PetType,my_data2$PHappyScore)
barplot(counts5,main=" A Barplot for Pet Type vs Happy Score",
  xlab="Happy Score",
  col=c("yellow","green"),
  legend = rownames(counts5))
```



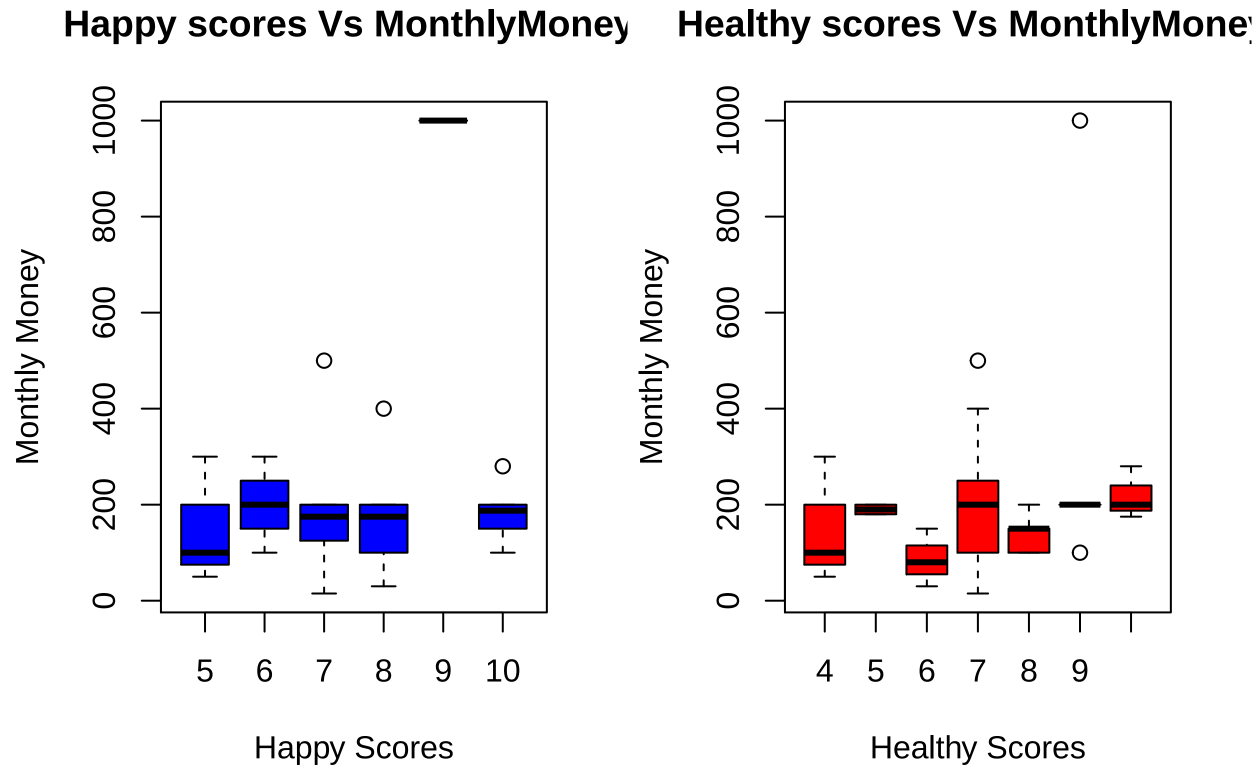
```
counts6 <- table(my_data2$PetType,my_data2$PHealthyScore)
barplot(counts6,main=" A Barplot for Pet Type vs Healthy Score",
  xlab="Healthy Score",
  col=c("yellow","green"),
  legend = rownames(counts6))
```



```r
#Box-Plot
par(mfrow=c(1,2))
# Happy scores Vs PetTime
plot1 <- boxplot(Ptime~PHappyScore,
      data=my_data2,
      main="Happy scores Vs PetTime ",
      xlab = "Happy Scores",
      ylab = "Pet Time",
      col="blue",
      border="black"
      )
# Healthy scores Vs PetTime
plot2 <- boxplot(Ptime~PHealthyScore,
      data=my_data2,
      main="Healthy scores Vs PetTime ",
      xlab = "Healthy Scores",
      ylab = "Pet Time",
      col="red",
      border="black"
      )
```



```r
# Happy scores Vs MonthlyMoney
plot3 <- boxplot(PMonthlyMoney~PHappyScore,
      data=my_data2,
      main="Happy scores Vs MonthlyMoney ",
      xlab = "Happy Scores",
      ylab = "Monthly Money",
      col="blue",
      border="black"
      )
# Healthy scores Vs MonthlyMoney
plot4 <- boxplot(PMonthlyMoney~PHealthyScore,
      data=my_data2,
      main="Healthy scores Vs MonthlyMoney ",
      xlab = "Healthy Scores",
      ylab = "Monthly Money",
      col="red",
      border="black"
      )
```



```
#sample means
mean(my_data2$PHappyScore)
```

```
## [1] 7.657895
```

```
mean(my_data2$PHealthyScore)
```

```
## [1] 7.368421
```

```
mean(my_data2$NHappyScore, na.rm=TRUE)
```

```
## [1] 7.647059
```

```
mean(my_data2$NHealthyScore,na.rm=TRUE)
```

```
## [1] 7.764706
```

```
#sample st.dev
sd(my_data2$PHappyScore)
```

```
## [1] 1.380879
```

```r
sd(my_data2$PHealthyScore)
```

```
## [1] 1.58406
```

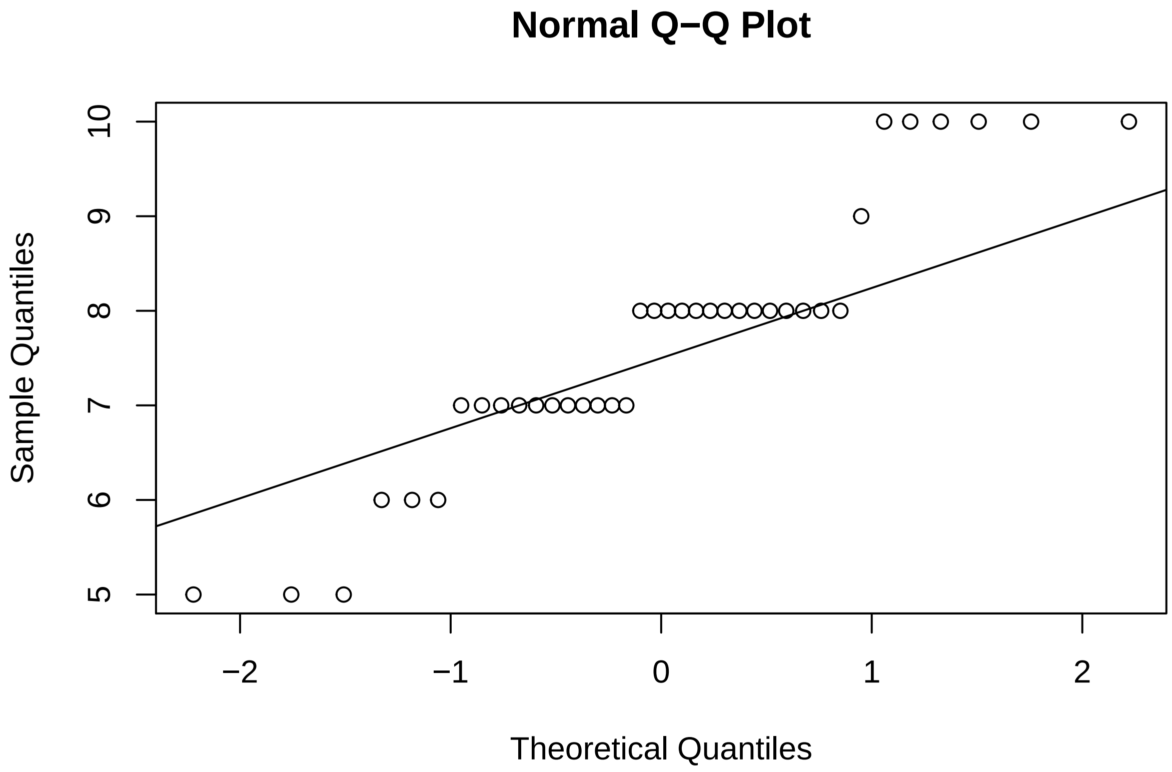
```r
sd(my_data2$NHappyScore, na.rm=TRUE)
```

```
## [1] 0.9963167
```

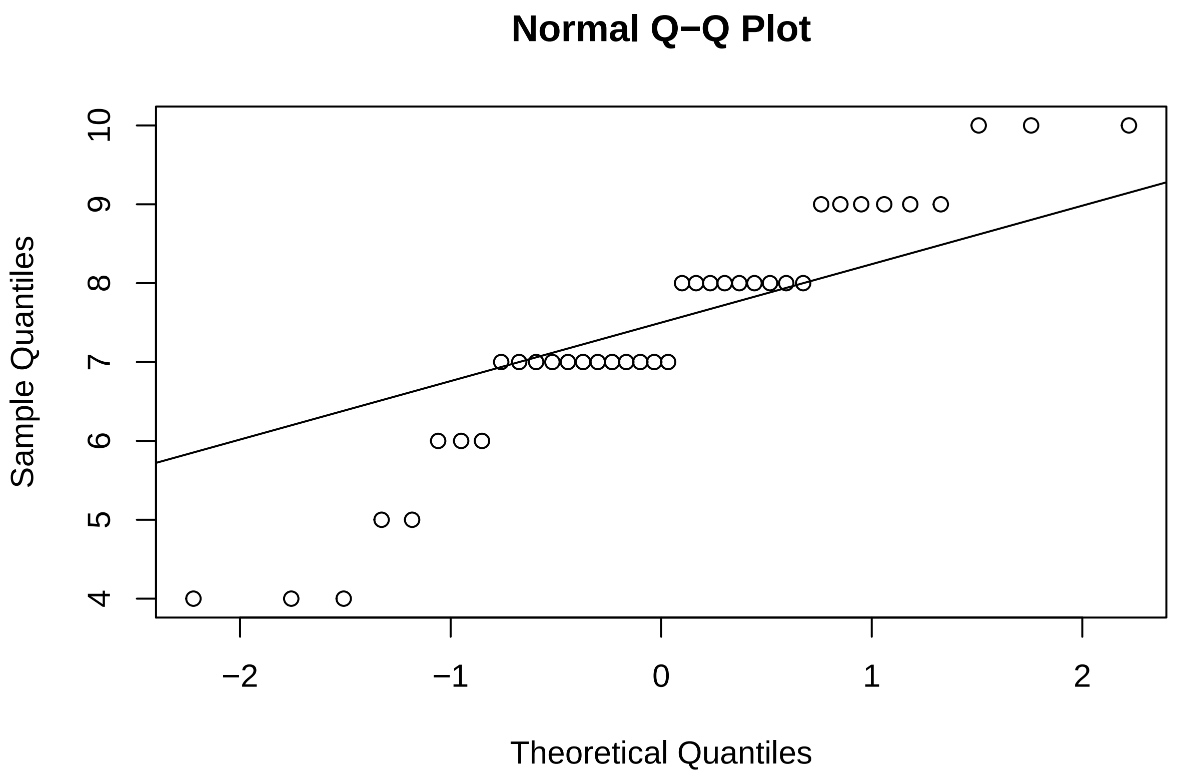
```r
sd(my_data2$NHealthyScore,na.rm=TRUE)
```

```
## [1] 1.393261
```

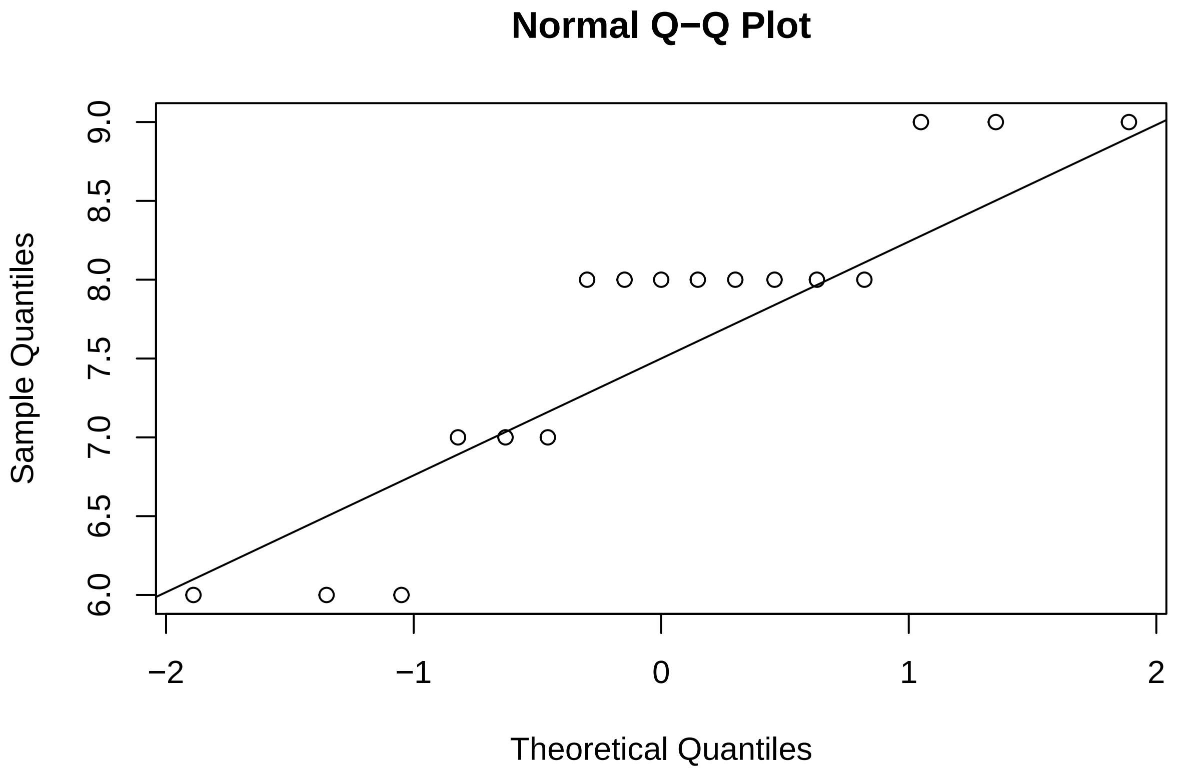
```r
#check the population data is normally distributed
#check the qqplot firstly
qqnorm(my_data2$PHappyScore)
qqline(my_data2$PHappyScore)
```



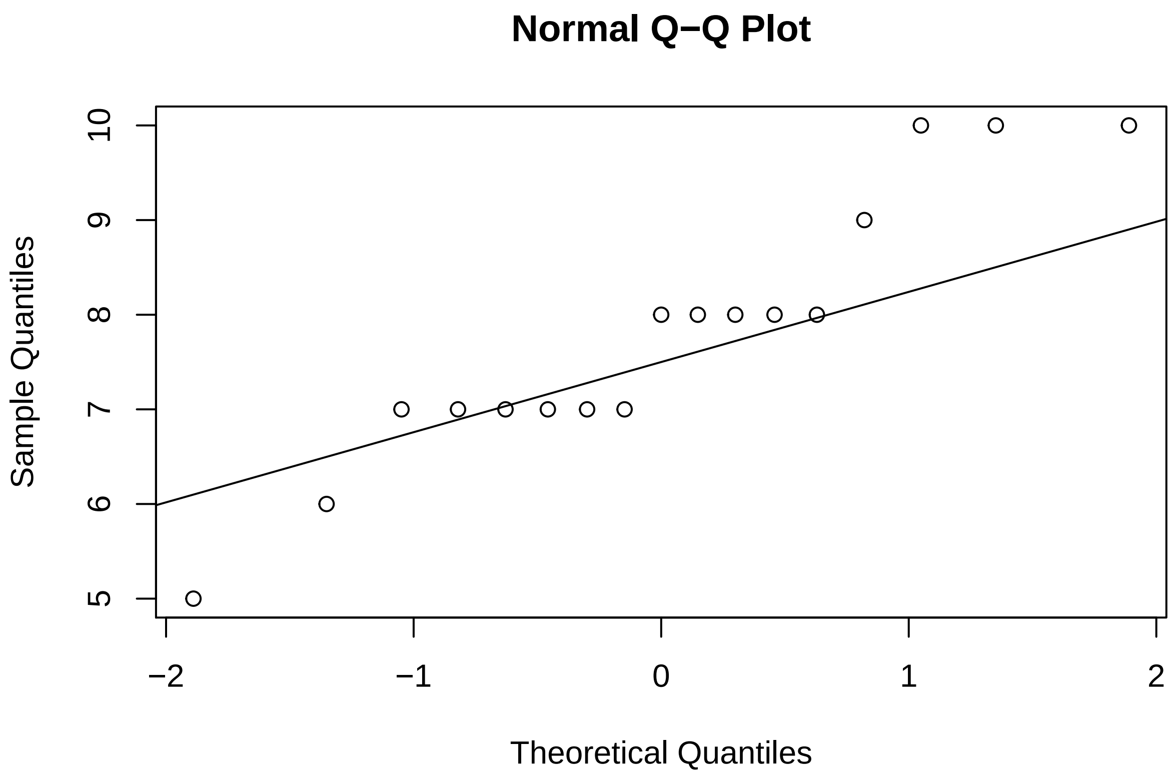
```r
qqnorm(my_data2$PHealthyScore)
qqline(my_data2$PHealthyScore)
```



```
qqnorm(my_data2$NHappyScore)
qqline(my_data2$NHappyScore)
```



```
qqnorm(my_data2$NHealthyScore)
qqline(my_data2$NHealthyScore)
```



```
#order the data from the smallest to largest
#calculate the quantile
order1 <- order(my_data2$PHappyScore)
quantile(order1,seq(0.01,0.99,0.1))
```

```
##    1%   11%   21%   31%   41%   51%   61%   71%   81%   91% 
##  1.37  5.07  8.77 12.47 16.17 19.87 23.57 27.27 30.97 34.67
```

```
order2 <- order(my_data2$PHealthyScore)
quantile(order2,seq(0.01,0.99,0.1))
```

```
##    1%   11%   21%   31%   41%   51%   61%   71%   81%   91% 
##  1.37  5.07  8.77 12.47 16.17 19.87 23.57 27.27 30.97 34.67
```

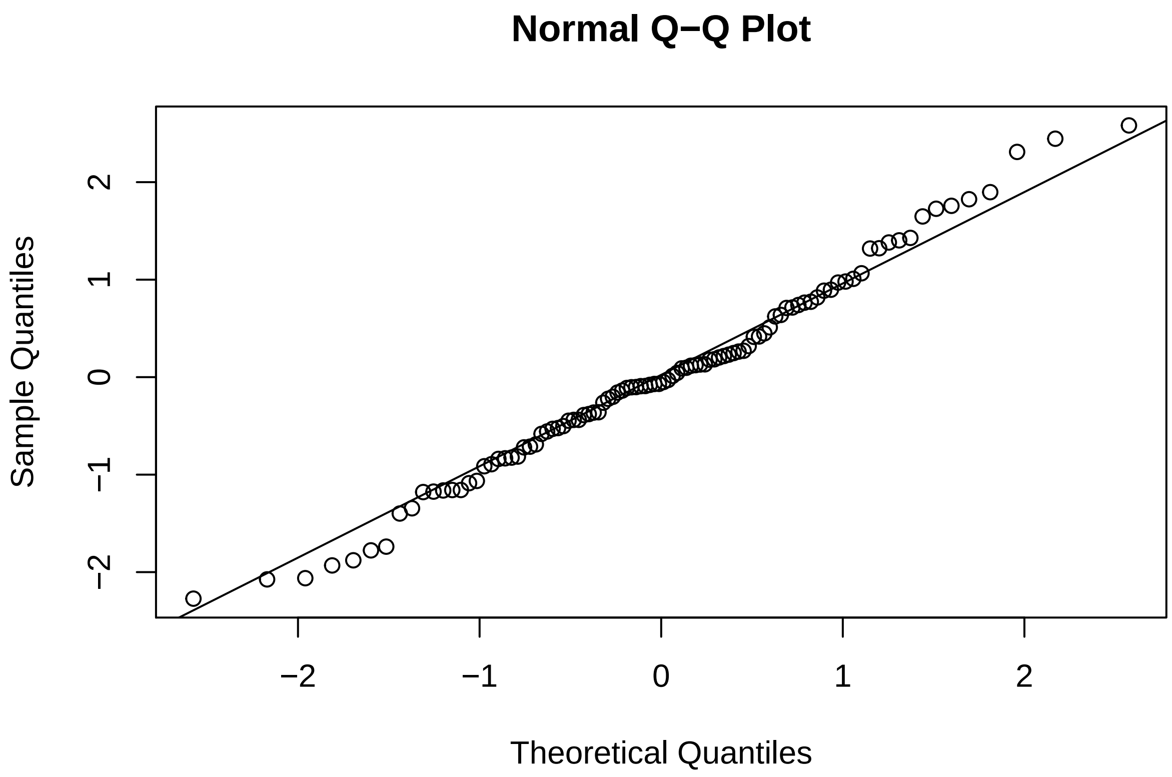
```
order3 <- order(my_data2$NHappyScore)
quantile(order2,seq(0.01,0.99,0.1))
```

```
##    1%   11%   21%   31%   41%   51%   61%   71%   81%   91% 
##  1.37  5.07  8.77 12.47 16.17 19.87 23.57 27.27 30.97 34.67
```

```
order4 <- order(my_data2$NHealthyScore)
quantile(order2,seq(0.01,0.99,0.1))
```

```
##    1%   11%   21%   31%   41%   51%   61%   71%   81%   91% 
##  1.37  5.07  8.77 12.47 16.17 19.87 23.57 27.27 30.97 34.67
```

```
#Normal distrbution
set.seed(100)
x <- rnorm(100)
qqnorm(x)
qqline(x)
```



```
quantile(x,seq(0.01,0.99,0.1))
```

```
##          1%         11%         21%         31%         41%         51% 
## -2.07637996 -1.16369702 -0.81666399 -0.44112017 -0.14611953 -0.03986303 
##         61%         71%         81%         91% 
##  0.18044155  0.42648148  0.83055240  1.40546230
```

```
#Two-sided t-test with 95% confidence interval (alpha=0.05)
#Happy
hypoth1 <- t.test(my_data2$PHappyScore,my_data2$NHappyScore,
                  data=my_data2,var.equal=TRUE)
hypoth1
```

```
## 
##  Two Sample t-test
## 
## data:  my_data2$PHappyScore and my_data2$NHappyScore
## t = 0.02908, df = 53, p-value = 0.9769
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.7365551  0.7582269
## sample estimates:
## mean of x mean of y 
##  7.657895  7.647059
```

```
hypoth1$conf.int
```

```
## [1] -0.7365551  0.7582269
## attr(,"conf.level")
## [1] 0.95
```

```r
hypoth1$p.value
```

```
## [1] 0.9769101
```

```r
#Healthy
hypoth2 <- t.test(my_data2$PHealthyScore,my_data2$NHealthyScore,
                  data=my_data2,var.equal=TRUE)
hypoth2
```

```
##
##  Two Sample t-test
##
## data:  my_data2$PHealthyScore and my_data2$NHealthyScore
## t = -0.88827, df = 53, p-value = 0.3784
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.2911145  0.4985448
## sample estimates:
## mean of x mean of y
##  7.368421  7.764706
```

```r
hypoth2$conf.int
```

```
## [1] -1.2911145  0.4985448
## attr(,"conf.level")
## [1] 0.95
```

```r
hypoth2$p.value
```

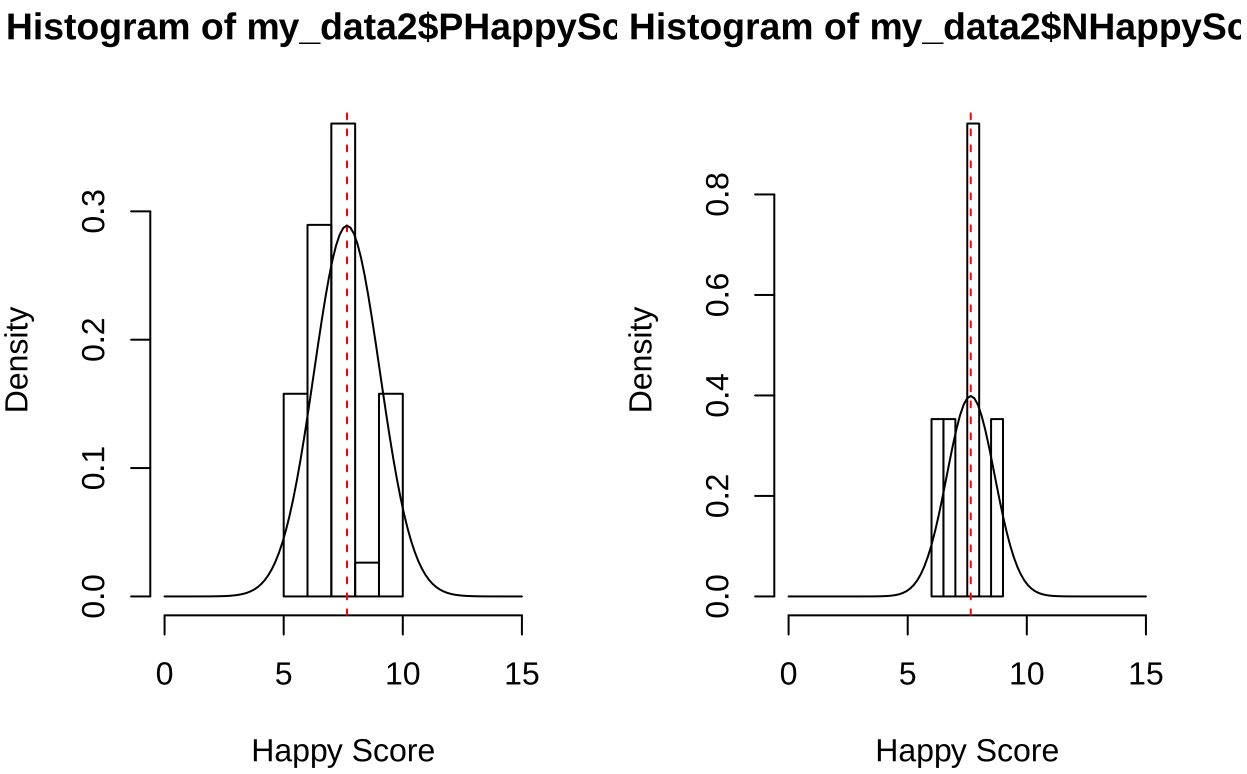
```
## [1] 0.3784114
```

```r
#Plot a histogram of the sampling distribution with normal distribution curve
#people have pets
par(mfrow=c(1,2))

h1 <-hist(my_data2$PHappyScore,
    xlab="Happy Score",xlim=c(0,15),ylab="Density", freq=FALSE)
abline(v=mean(my_data2$PHappyScore),col="red",lty="dashed")
curve(dnorm(x, mean=mean(my_data2$PHappyScore),
            sd=sd(my_data2$PHappyScore)), add=TRUE, col="black")


h2 <-hist(my_data2$NHappyScore,
     xlab="Happy Score",xlim=c(0,15),ylab="Density", freq=FALSE)
abline(v=mean(my_data2$NHappyScore,na.rm=TRUE),col="red",lty="dashed")
curve(dnorm(x, mean=mean(my_data2$NHappyScore, na.rm=TRUE)),
      sd=sd(mean(my_data2$NHappyScore, na.rm=TRUE)), add=TRUE, col="black")
```

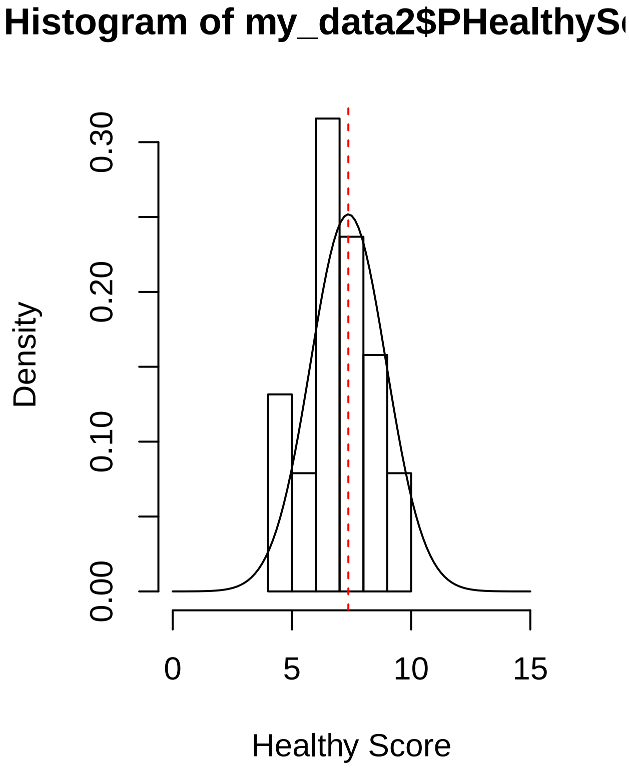
```
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "sd" is not a
## graphical parameter
```



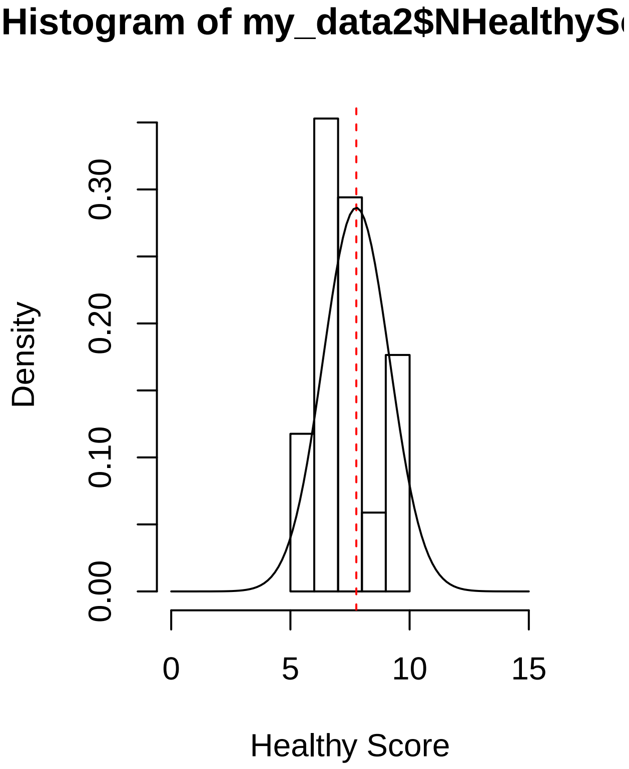
```
#people do not have pets
par(mfrow=c(1,2))

h3 <- hist(my_data2$PHealthyScore,xlab="Healthy Score",xlim=c(0,15),
           ylab="Density", freq=FALSE)
abline(v=mean(my_data2$PHealthyScore),col="red",lty="dashed")
curve(dnorm(x, mean=mean(my_data2$PHealthyScore),
            sd=sd(my_data2$PHealthyScore)), add=TRUE, col="black")

hist(my_data2$NHealthyScore,
     xlab="Healthy Score",xlim=c(0,15),ylab="Density", freq=FALSE)
abline(v=mean(my_data2$NHealthyScore,na.rm=TRUE),col="red",lty="dashed")
curve(dnorm(x, mean=mean(my_data2$NHealthyScore,na.rm=TRUE),
            sd=sd(my_data2$NHealthyScore,na.rm=TRUE)), add=TRUE, col="black")
```



Histogram of my_data2$PHealthyS



Histogram of my_data2$NHealthyS