R language and data analysis: plot for summary statistics

Qiang Shen

Dec. 26, 2017

categorical and quantitative data

- categorical data:

bar chart*, pie chart

- quantitative data:

histogram, box plot, dot plot

data

```
library(vcd);library(grid)
# Arthritis
str(Arthritis)
```

```
'data.frame':   84 obs. of  5 variables:
 $ ID        : int  57 46 77 17 36 23 75 39 33 55 ...
 $ Treatment: Factor w/ 2 levels "Placebo","Treated": 2 2 2
 $ Sex       : Factor w/ 2 levels "Female","Male": 2 2 2 2 2
 $ Age       : int  27 29 30 32 46 58 59 59 63 63 ...
 $ Improved : Ord.factor w/ 3 levels "None"<"Some"<..: 2 1
```

```
glimpse(Arthritis)
```

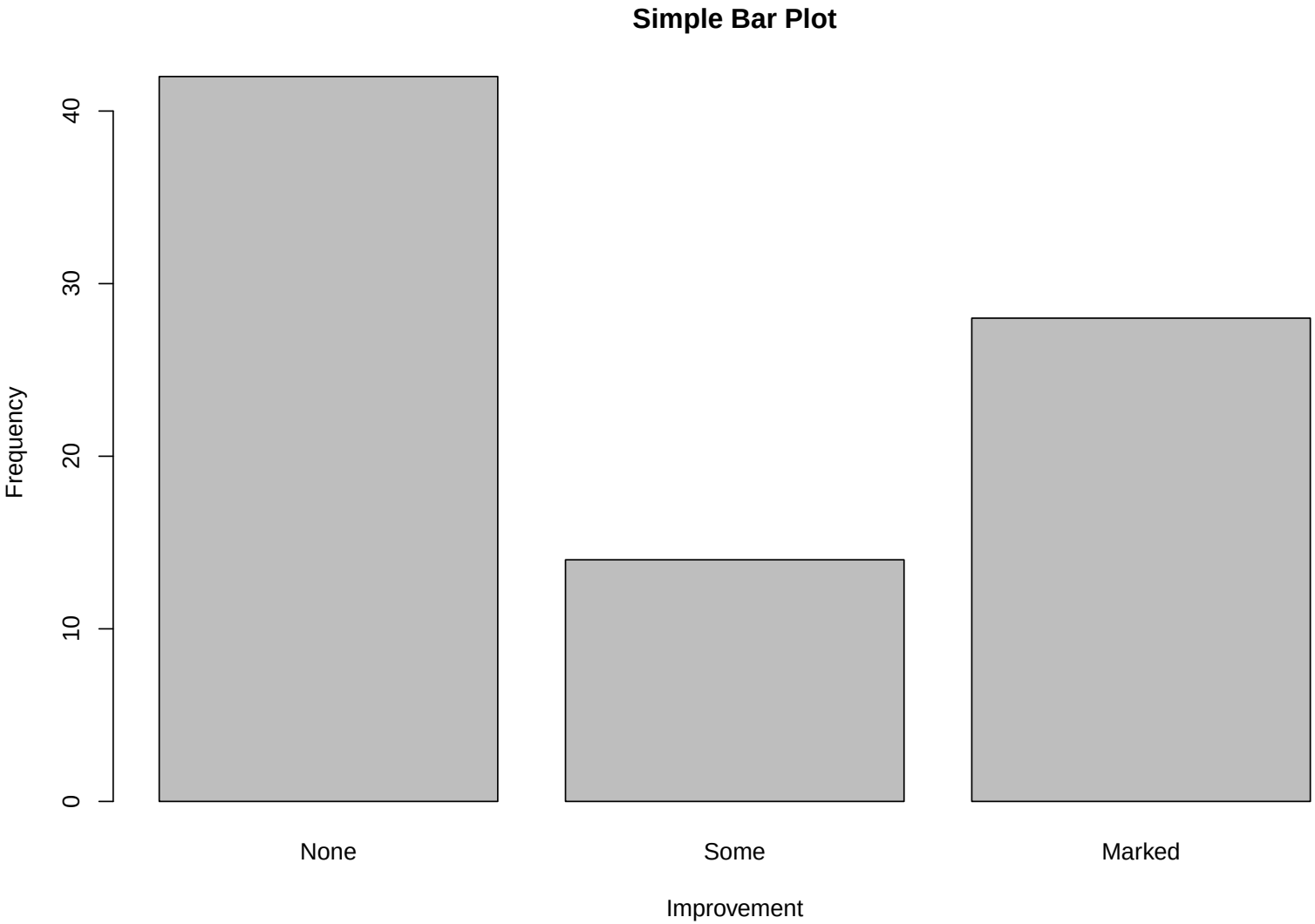
```
Observations: 84
Variables: 5
$ ID        <int> 57, 46, 77, 17, 36, 23, 75, 39, 33, 55, 3
$ Treatment <fctr> Treated, Treated, Treated, Treated, Trea
```

simple bar plot

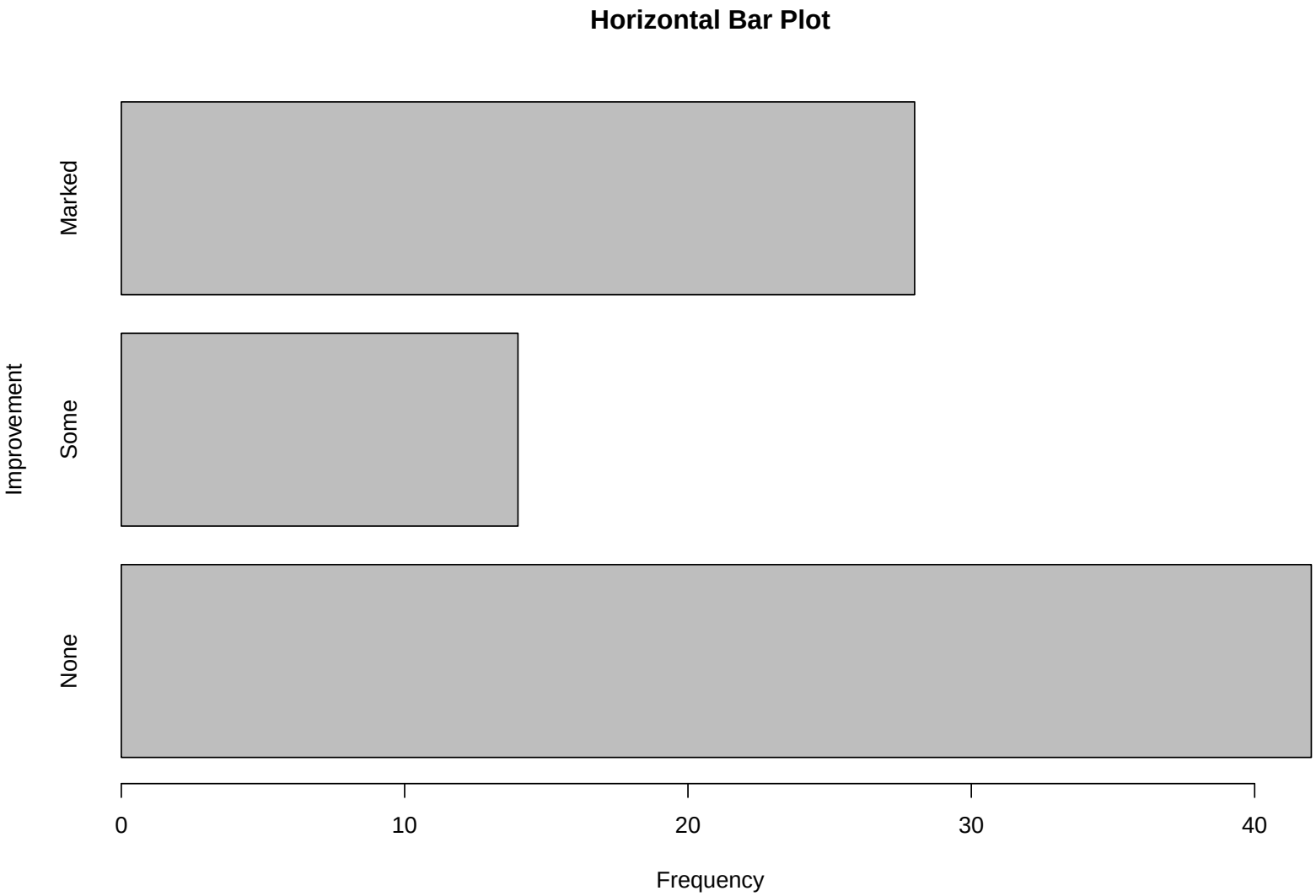
barplot

```
counts <- table(Arthritis$Improved)
barplot(counts, main = "Simple Bar Plot", xlab =
"Improvement", ylab = "Frequency")
```

simple bar plot



horizontal bar plot



horizontal bar plot

`horiz = TRUE`

```
counts <- table(Arthritis$Improved)
barplot(counts, main = "Horizontal Bar Plot",
xlab = "Frequency",ylab = "Improvement", horiz = TRUE)
```

bar plot with 2 conditions

```
## get counts for Improved by Treatment table
counts <- table(Arthritis$Improved, Arthritis$Treatment)
counts
```

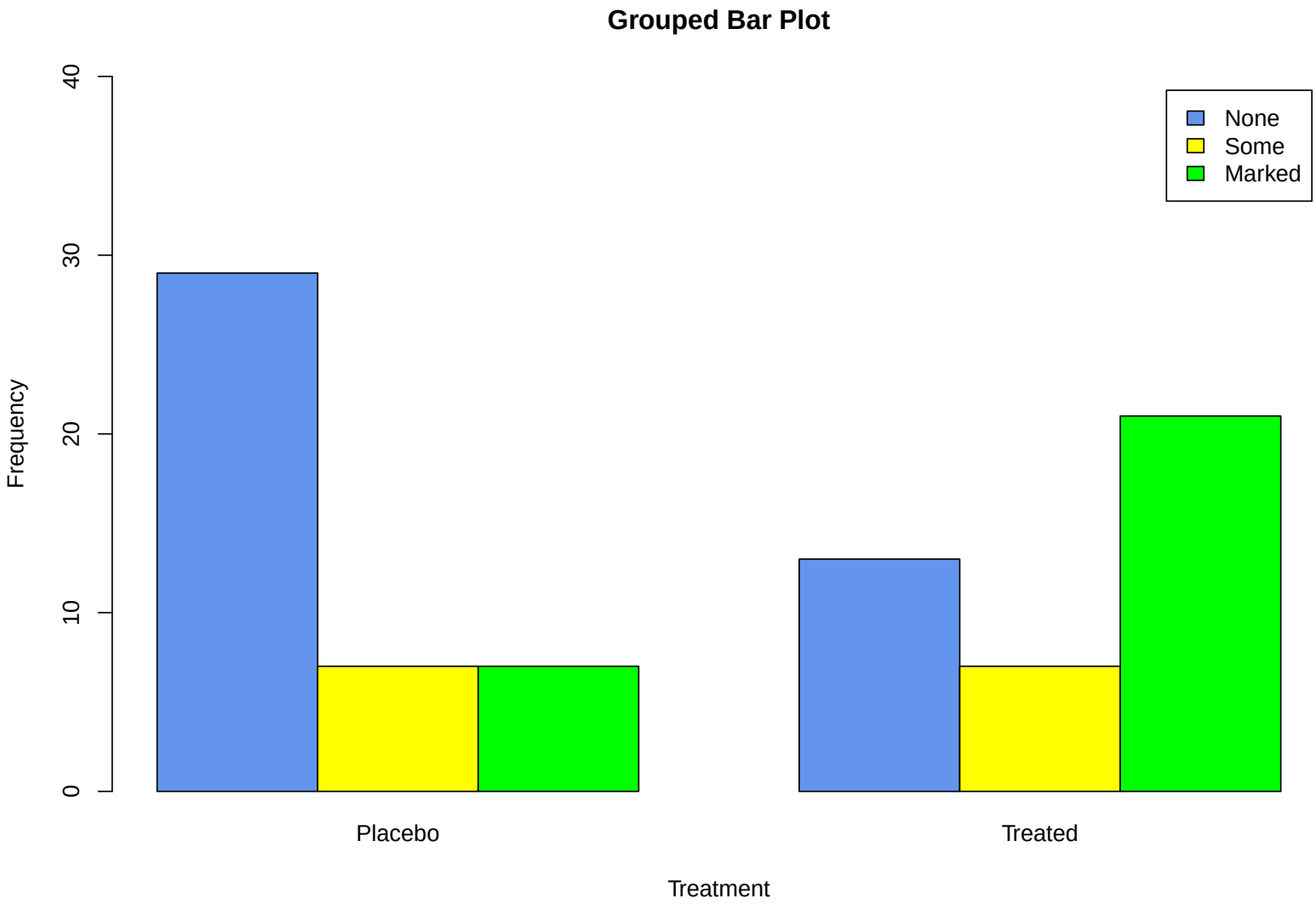
```
        Placebo Treated
  None       29      13
  Some        7       7
  Marked      7      21
```

stacked barplot

stacked plot

```
counts <- table(Arthritis$Improved, Arthritis$Treatment)
barplot(counts, main = "Stacked Bar Plot",
        xlab = "Treatment",
        ylab = "Frequency", col = c("cornflowerblue",
        "yellow", "green"),
        legend = rownames(counts),ylim=c(0,60))
##args.legend = list(x = "topleft",inset=0.05)
```

grouped barplot

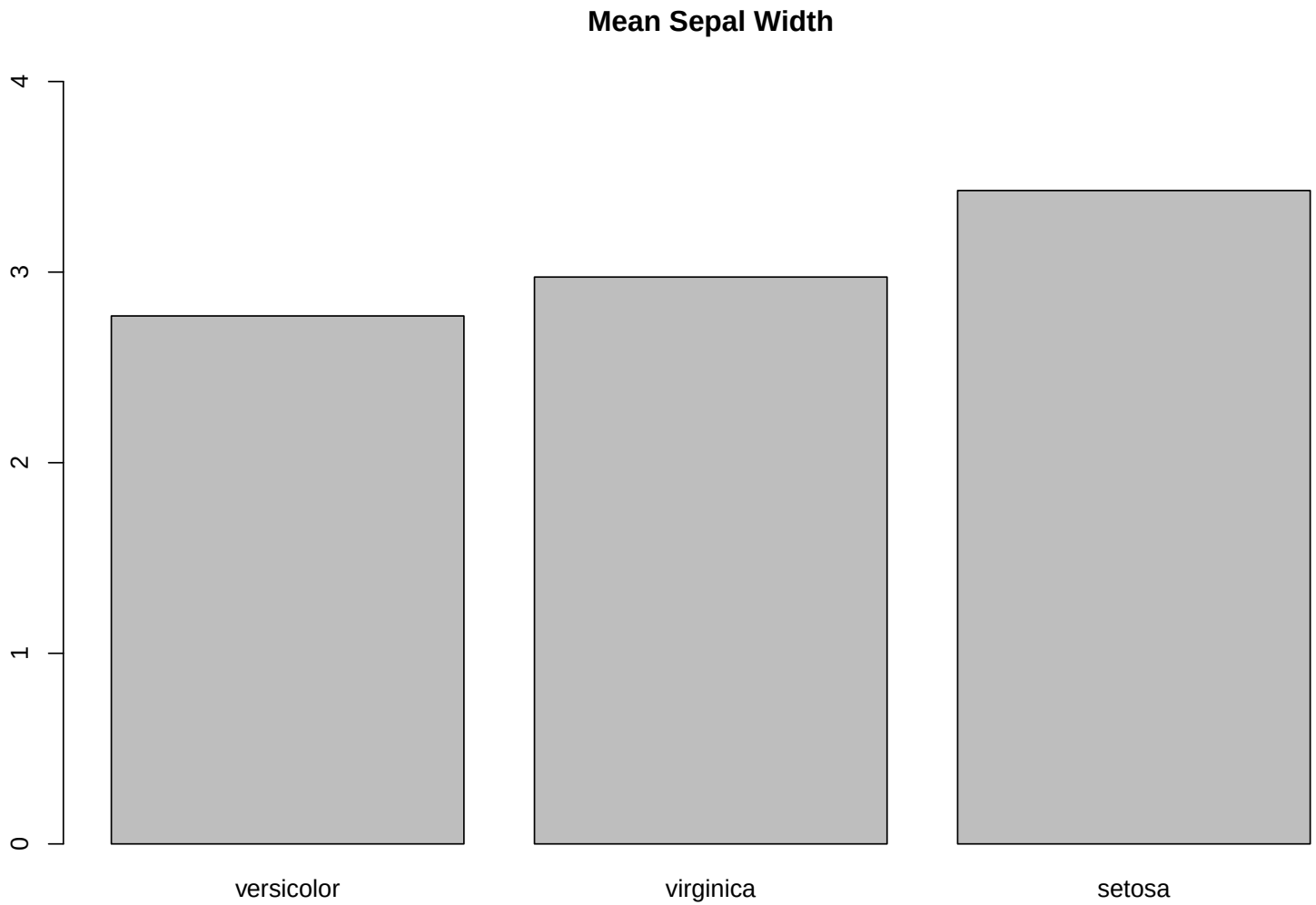


grouped barplot

```
counts <- table(Arthritis$Improved,
                Arthritis$Treatment)

barplot(counts, main = "Grouped Bar Plot",
        beside = TRUE,
        xlab = "Treatment",
        ylab = "Frequency", col = c("cornflowerblue",
                "yellow", "green"),
        legend = rownames(counts),
        ylim=c(0,40))
```

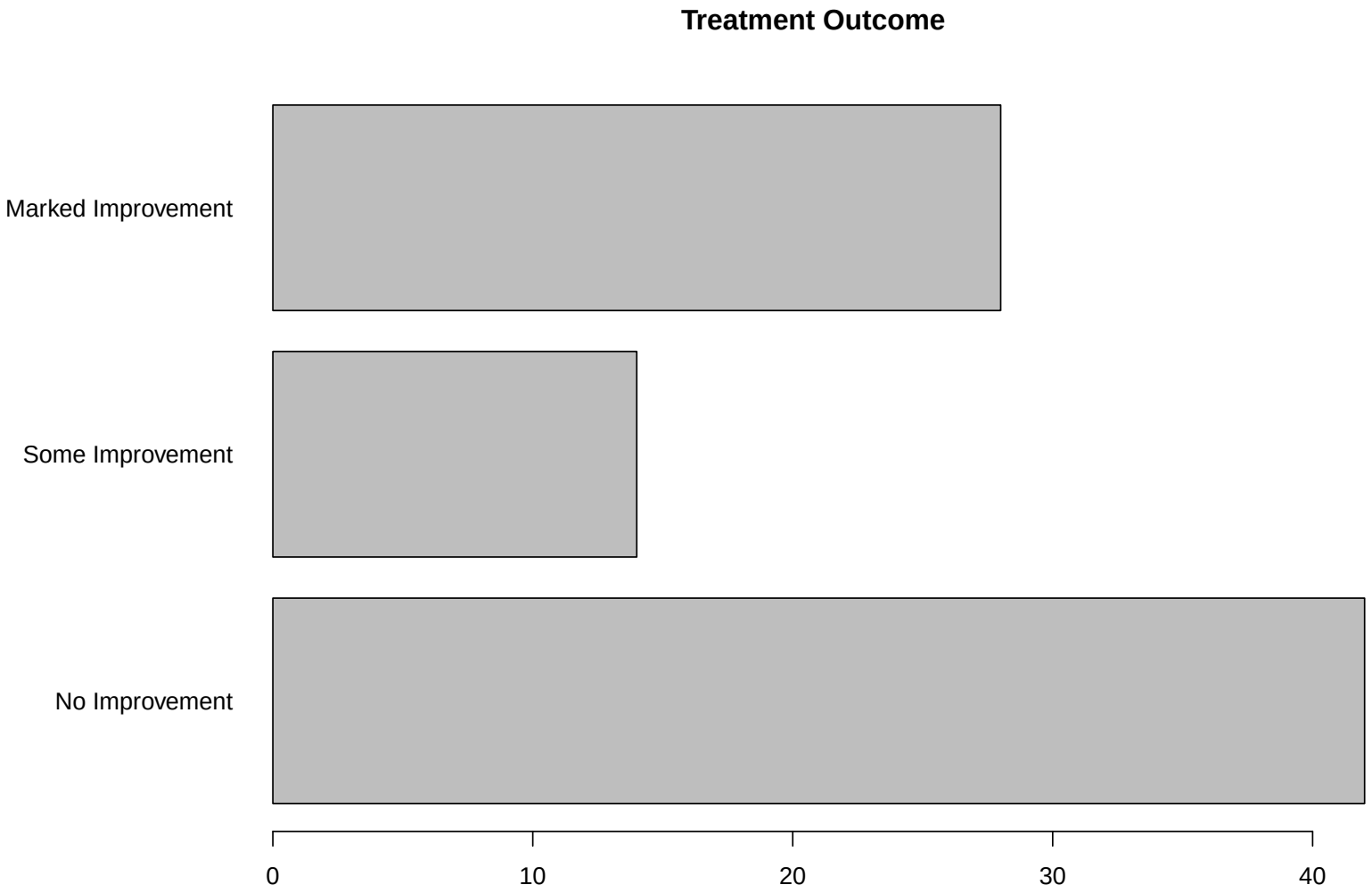
Mean bar plots



Mean bar plots

```
means<-tapply(iris$Sepal.Width,iris$Species,mean)
means<-means[order(means)]
x<-names(means)
barplot(means[1:3],names.arg=x,ylim=c(0,4))
title("Mean Sepal Width")
```

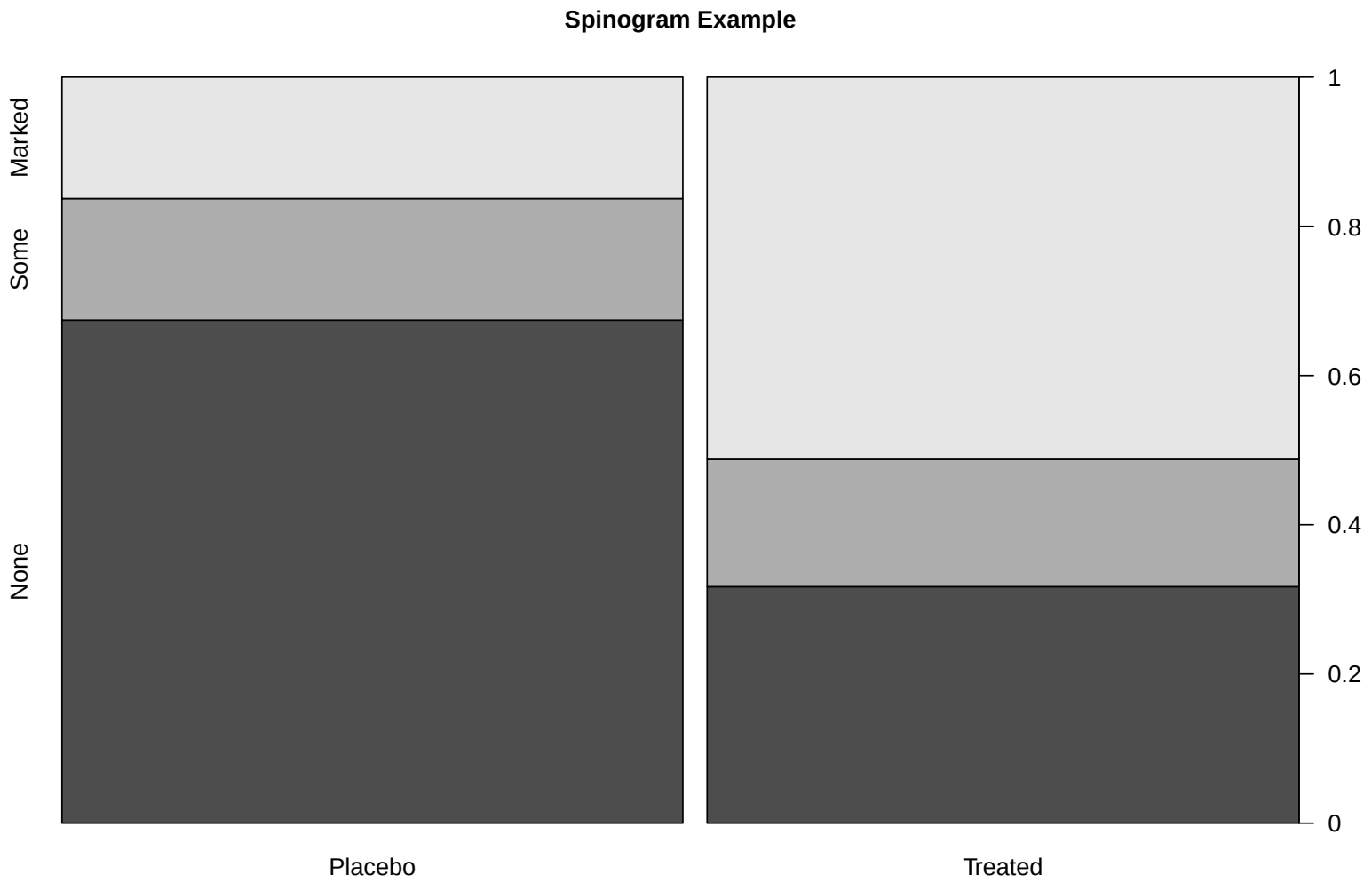
Fitting labels in bar plots



Fitting labels in bar plots

```
par(mar = c(5, 10, 4, 2))
par(las = 2)
counts <- table(Arthritis$Improved)
barplot(counts, main = "Treatment Outcome", horiz = TRUE,
                names.arg = c("No Improvement",
               "Some Improvement", "Marked Improvement"))
```

Spinograms


Spinogram Example
Marked
Some
None
1
0.8
0.6
0.4
0.2
0
Placebo
Treated

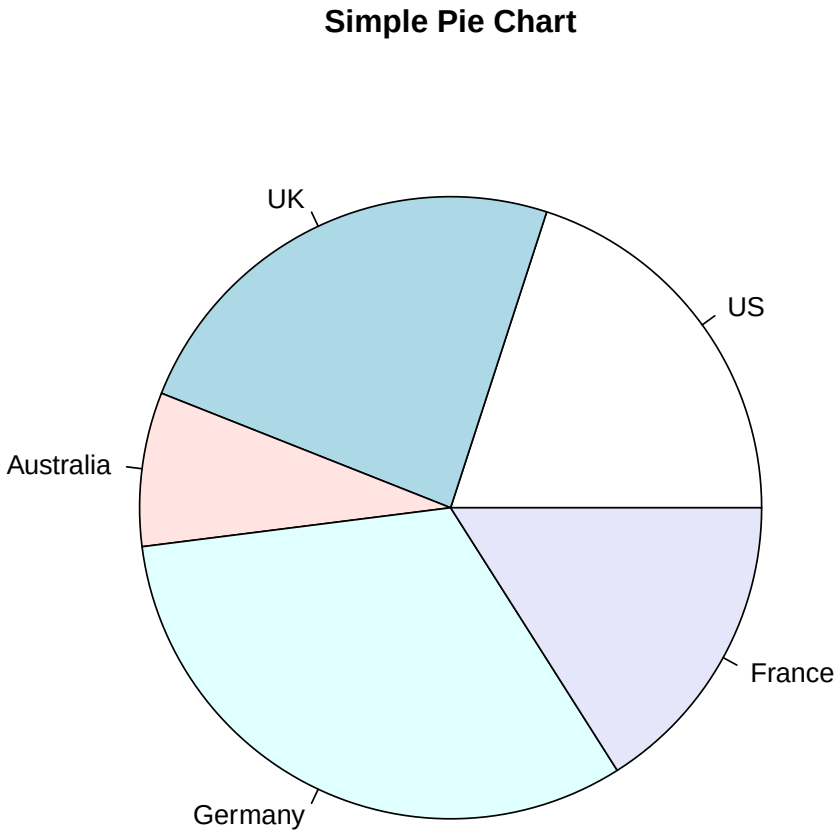
Spinograms

```
counts <- table(Arthritis$Treatment, Arthritis$Improved)
spine(counts, main = "Spinogram Example")
```

summary

- `table,barplot`
- `horiz=T`
- `beside=T table(x,y)`
- `names.arg = c("No Improvement","Some Improvement", "Marked Improvement")`
- Spinograms: `spine`

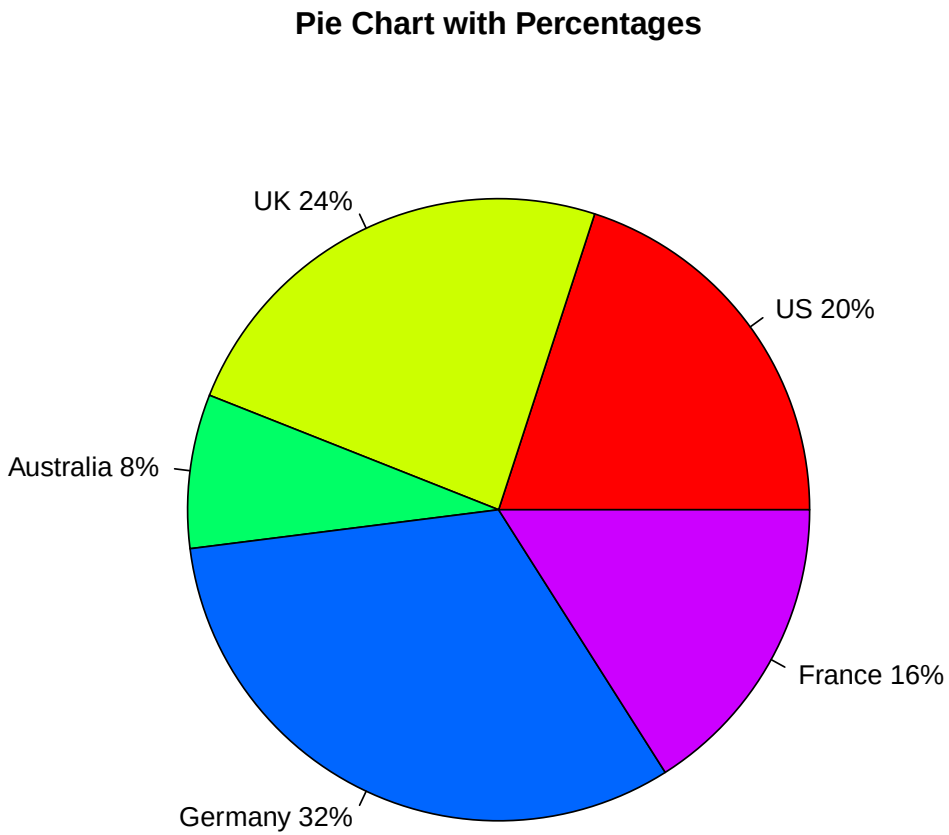
pie plot 1



pie plot 1

```
slices <- c(10, 12, 4, 16, 8)
lbls <- c("US", "UK", "Australia", "Germany", "France")
pie(slices, labels = lbls, main = "Simple Pie Chart")
```

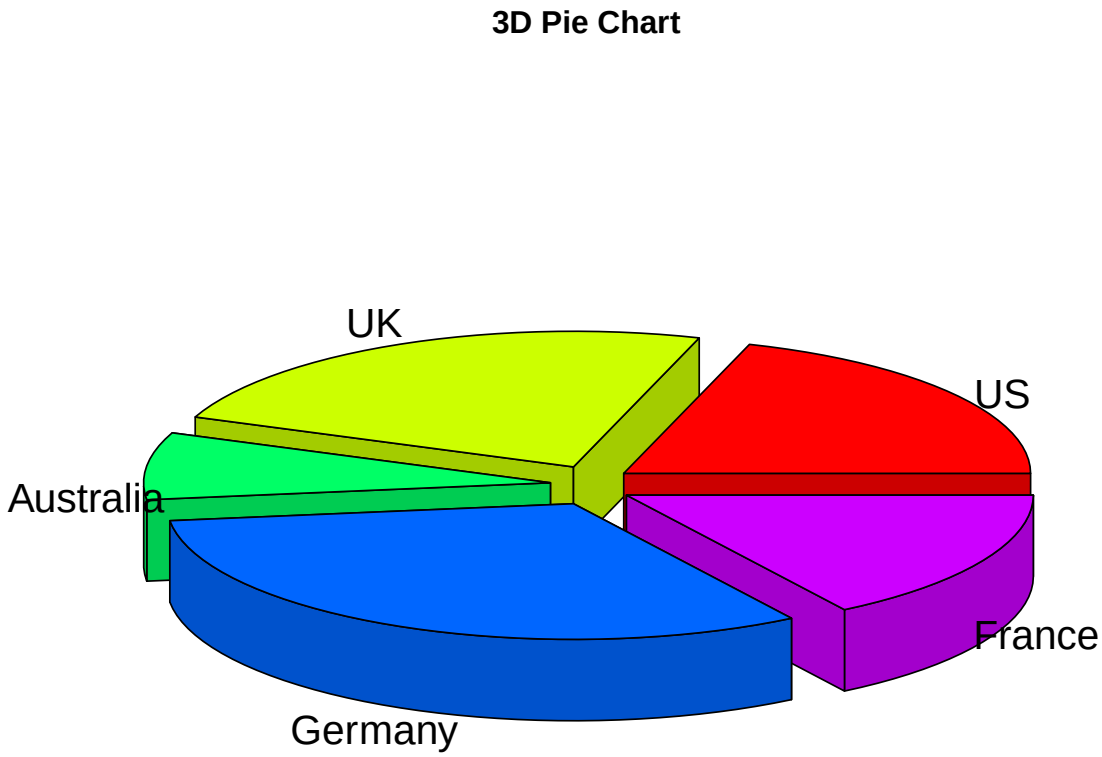
pie plot 2


Pie Chart with Percentages
UK 24%
US 20%
Australia 8%
France 16%
Germany 32%

pie plot 2

```
slices <- c(10, 12, 4, 16, 8)
lbls <- c("US", "UK", "Australia", "Germany", "France")
pct <- round(slices/sum(slices) * 100)
lbls2 <- paste(lbls, " ", pct, "%", sep = "")
pie(slices, labels = lbls2, col = rainbow(length(lbls)),
    main = "Pie Chart with Percentages")
```

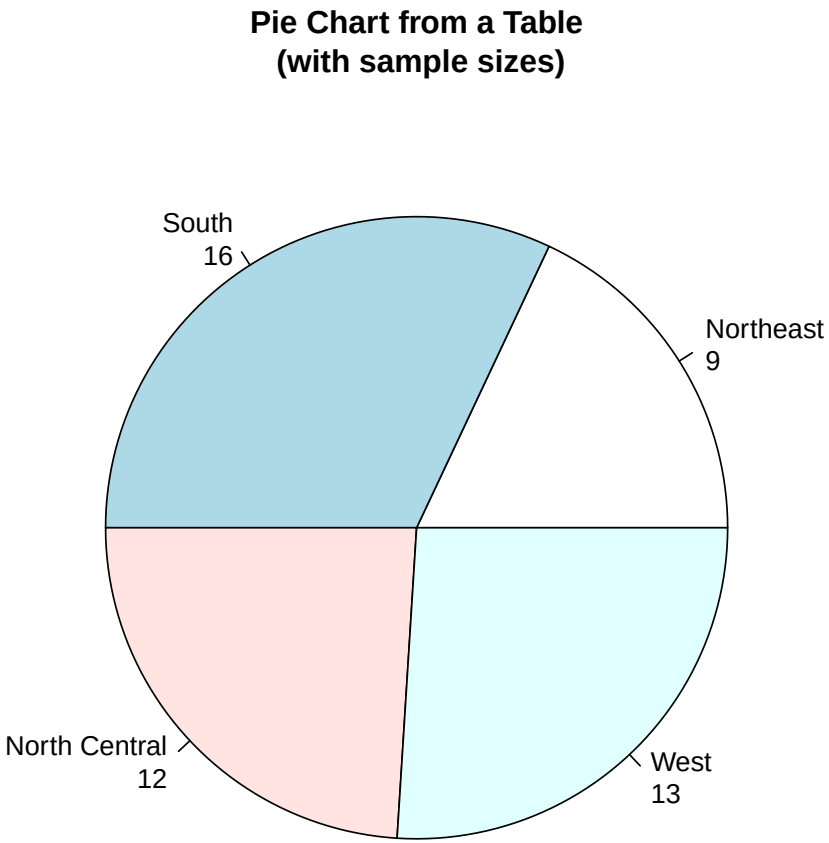
pie plot 3: 3D



pie plot 3: 3D

```
slices <- c(10, 12, 4, 16, 8)
lbls <- c("US", "UK", "Australia", "Germany", "France")
library(plotrix)
pie3D(slices, labels = lbls, explode = 0.1,
      main = "3D Pie Chart ")
```

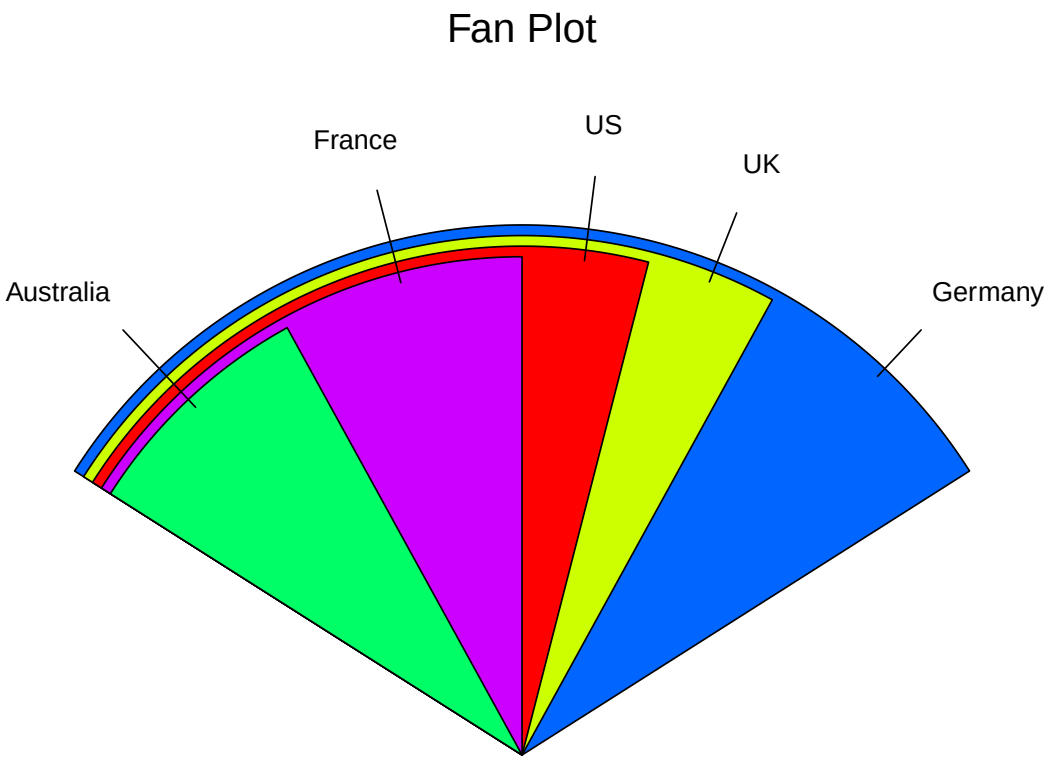
pie plot 4: from table



pie plot 4: from table

```
mytable <- table(state.region)
lbls <- paste(names(mytable), "\n", mytable, sep = "")
pie(mytable, labels = lbls,
    main = "Pie Chart from a Table\n (with sample sizes)")
```

fan plot



fan plot

```
library(plotrix)
slices <- c(10, 12, 4, 16, 8)
lbls <- c("US", "UK", "Australia", "Germany", "France")
fan.plot(slices, labels = lbls, main = "Fan Plot")
```

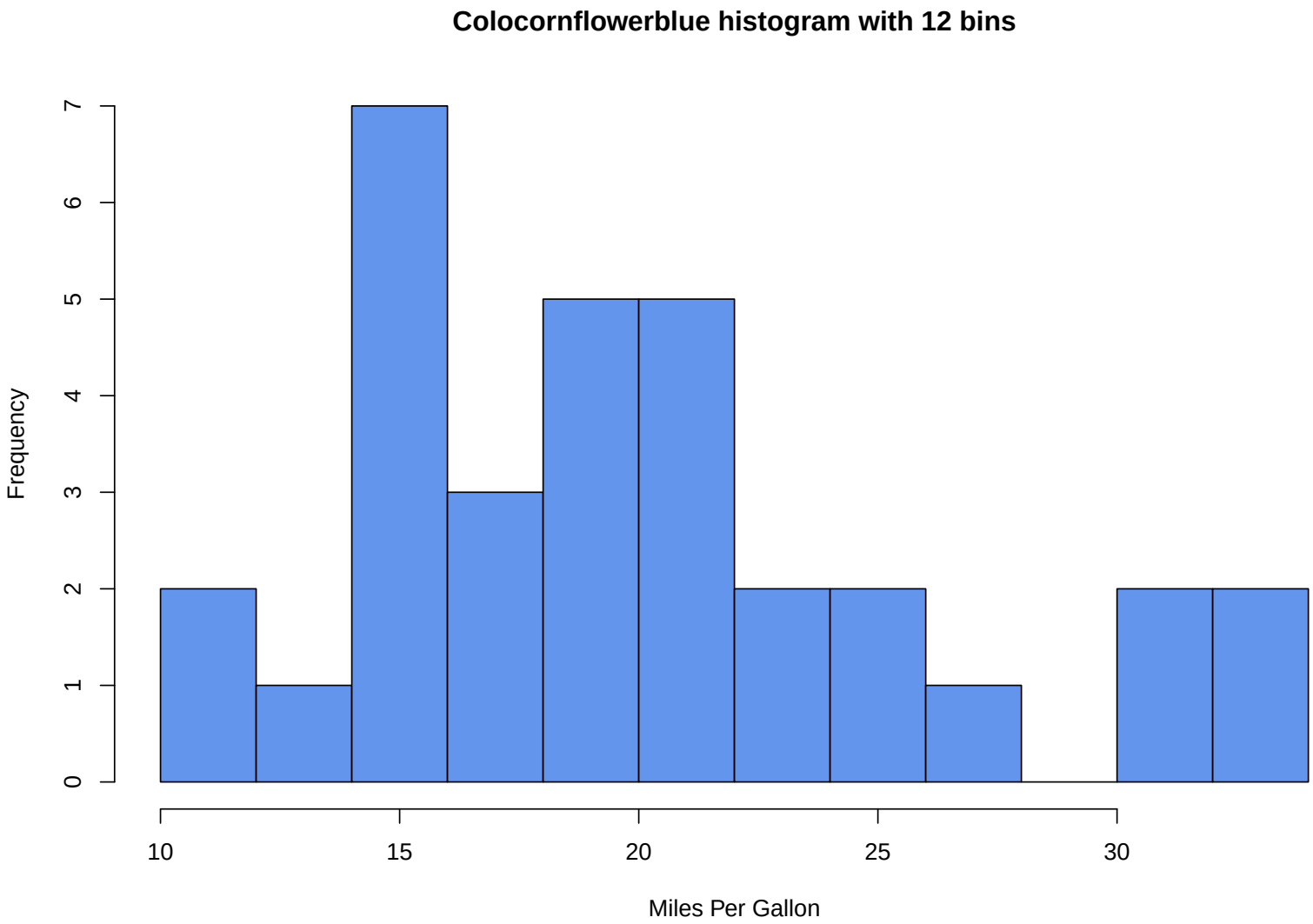
pie plot summary

1. popular and but accurate
2. `pie(x,labels=...)`
3. `pie3D(x,labels=...,explode = 0.1)`
4. `table`
5. `fan.plot` (plotrix)

histogram: frequencies

```
dev.new()
hist(mtcars$mpg)
```

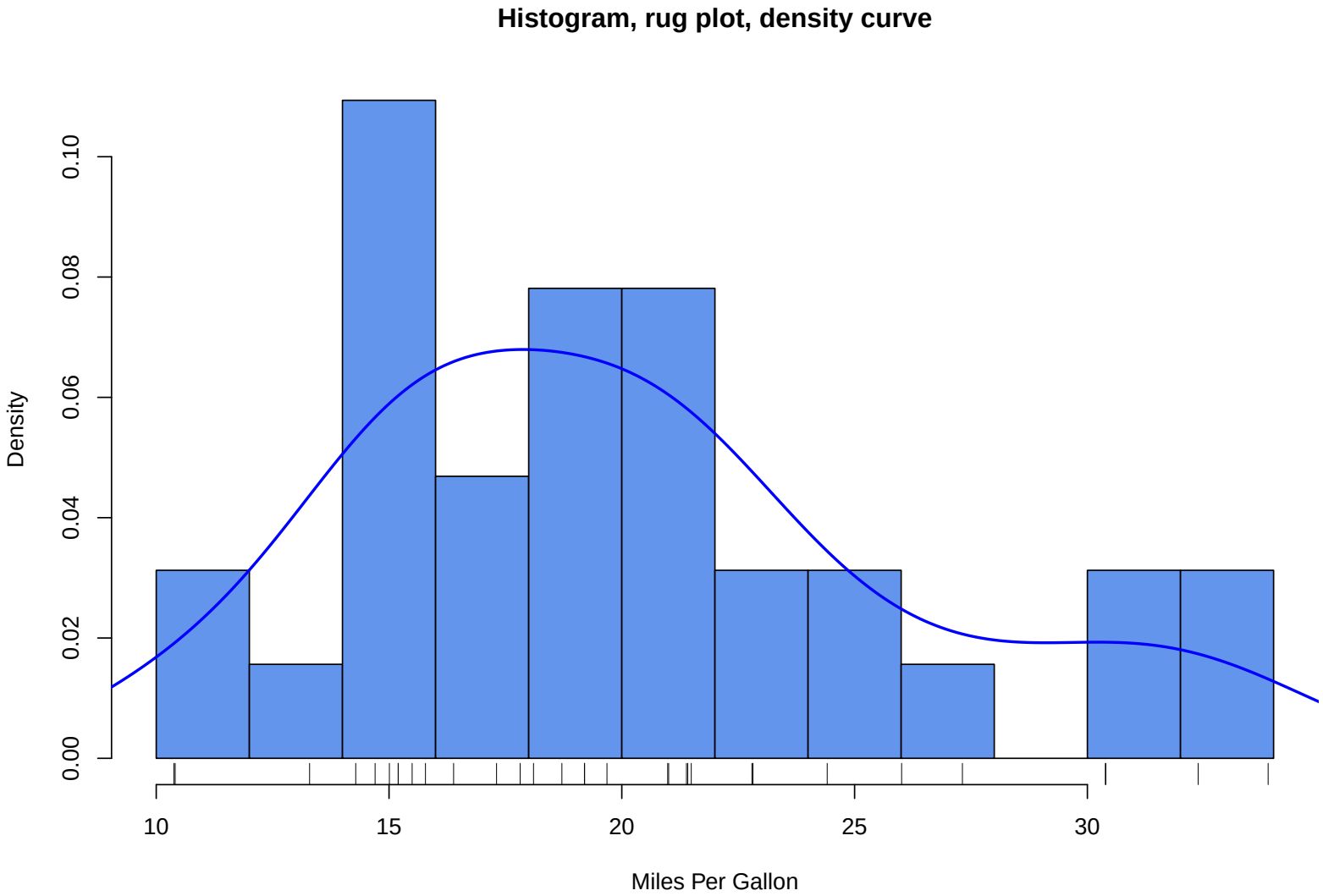
histogram: frequencies



histogram: frequencies

```
hist(mtcars$mpg)
hist(mtcars$mpg, breaks = 12, col = "cornflowerblue",
     xlab = "Miles Per Gallon",
     main = "Colocornflowerblue histogram with 12 bins")
```

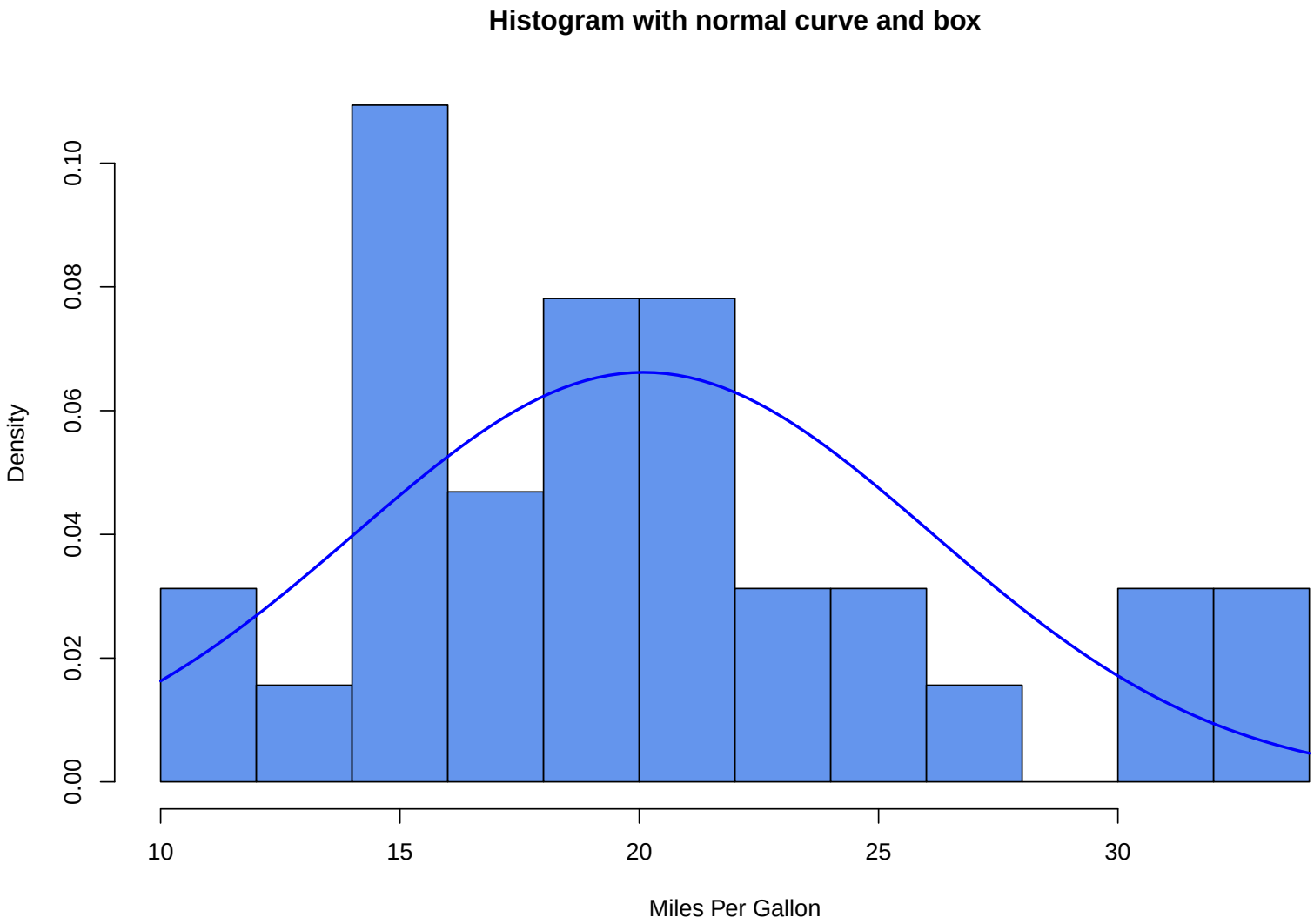
histogram: probability densities



histogram: probability densities

```
hist(mtcars$mpg, freq = FALSE, breaks = 12, col = "cornflow
     xlab = "Miles Per Gallon",
     main = "Histogram, rug plot, density curve")
rug(jitter(mtcars$mpg))
lines(density(mtcars$mpg), col = "blue", lwd = 2)
```

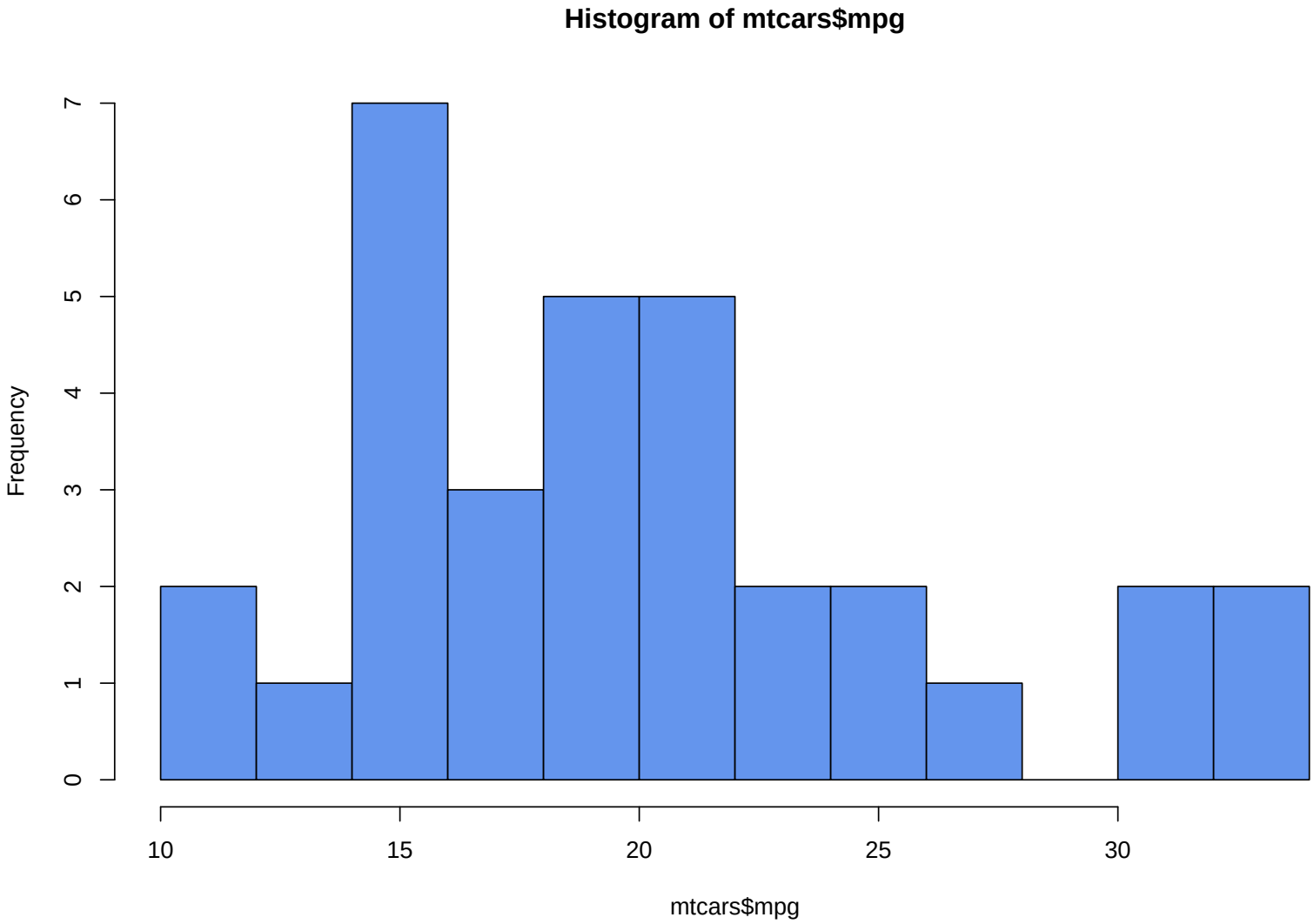
histogram with fitted curve



histogram with fitted curve

```
x <- mtcars$mpg
hist(mtcars$mpg, prob=TRUE,breaks = 12, col = "cornflowerbl
           xlab = "Miles Per Gallon",
    main = "Histogram with normal curve and box")
curve(dnorm(x, mean=mean(mtcars$mpg),
  sd=sd(mtcars$mpg)), add=TRUE,col = "blue", lwd = 2)
```

histogram: percentage

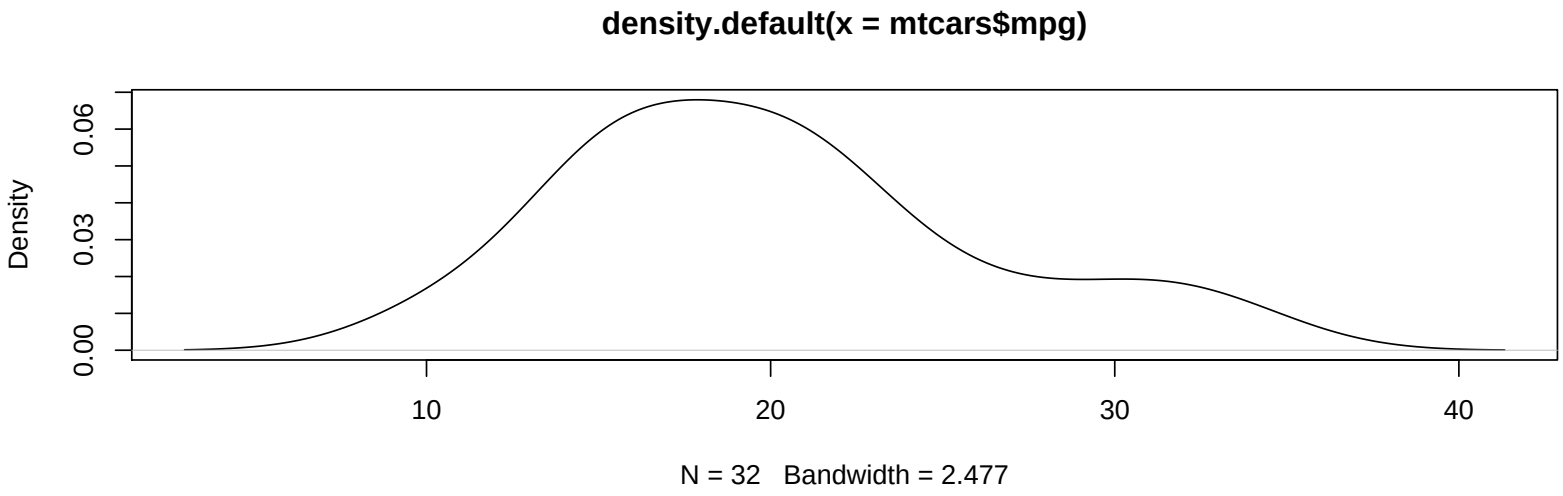


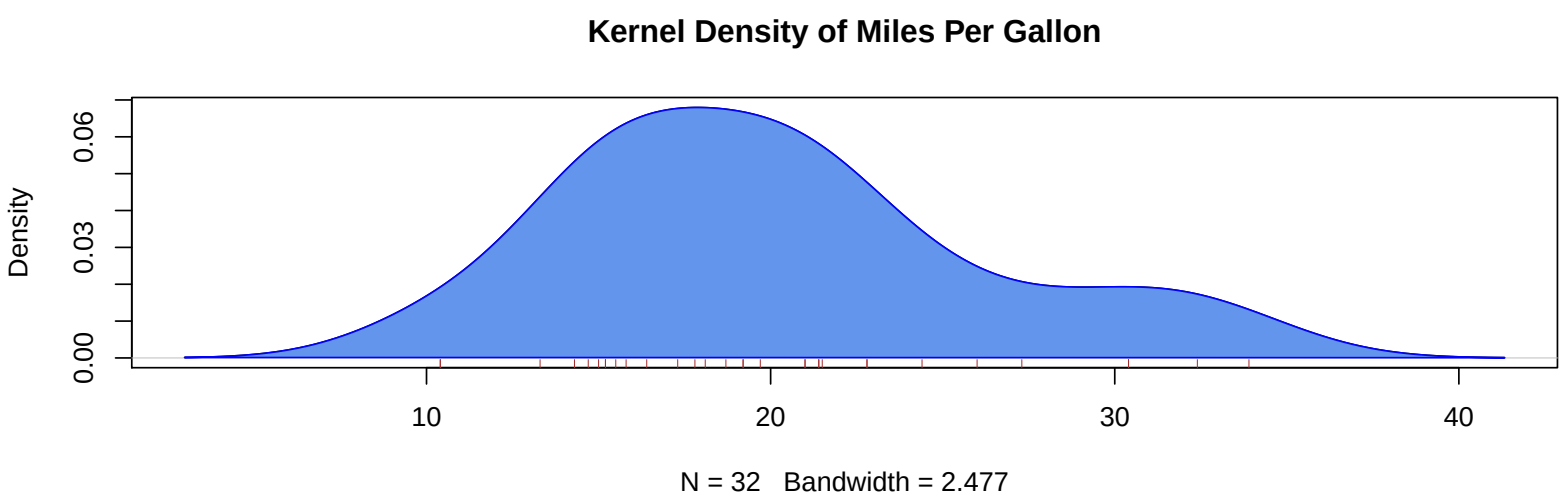

```
     [,1]
```

histogram: percentage

```
par(mfrow=c(1,1))
h = hist(mtcars$mpg,breaks=12,col='cornflowerblue')
# h$density
t(as.matrix(h$density)) %*% as.matrix
#(diff(h$breaks,1))#it just equals 1.
h$density <- h$counts/sum(h$counts)*100
plot(h,freq=F,ylab='Percentage',col='cornflowerblue',
     xlim=c(10,35),ylim=c(0,30),main=
       'histogram: Percentage')
```

kernal density function

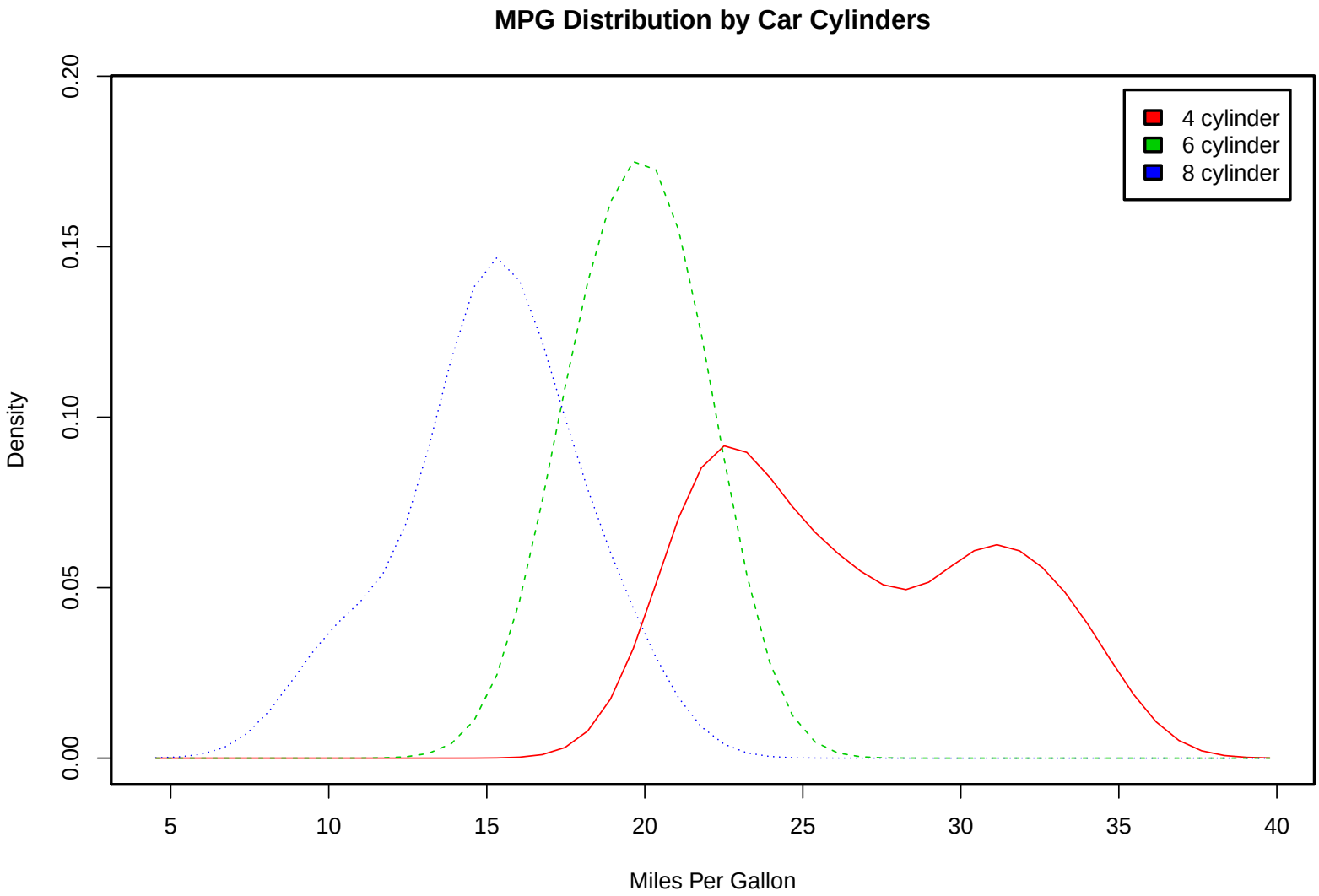




kernal density function

```
par(mfrow = c(1, 1))
d <- density(mtcars$mpg)
plot(d)
d <- density(mtcars$mpg)##bw="SJ", kernel="epanechnikov"
plot(d, main = "Kernel Density of Miles Per Gallon")
polygon(d, col = "cornflowerblue", border = "green")
rug(mtcars$mpg, col = "brown")
```

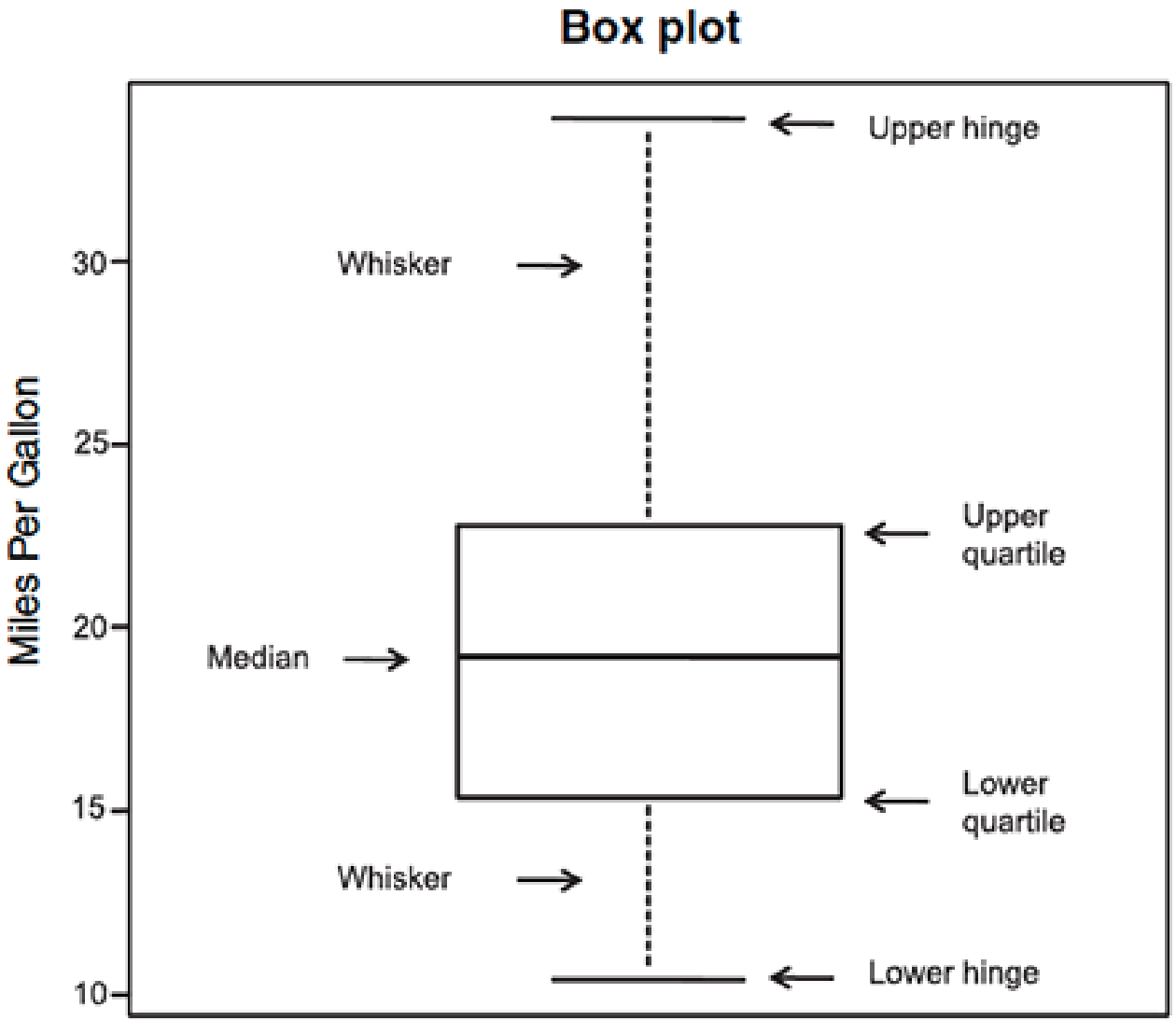
Comparing kernel density plots



Comparing kernel density plots

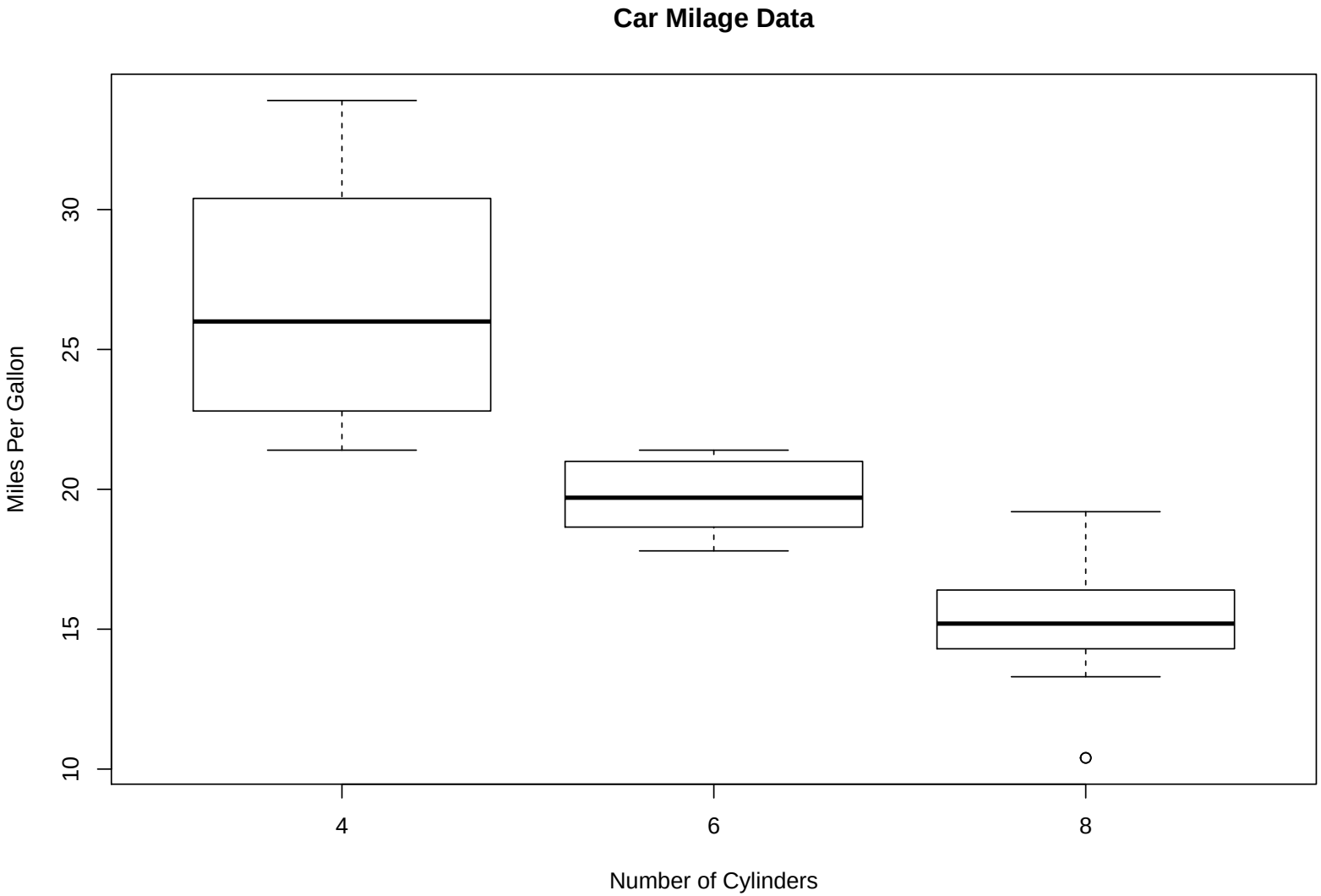
```
par(lwd = 2)
library(sm)
attach(mtcars)
cyl.f <- factor(cyl, levels = c(4, 6, 8),
                labels = c("4 cylinder", "6 cylinder", "8 c
sm.density.compare(mpg, cyl, xlab = "Miles Per Gallon")
title(main = "MPG Distribution by Car Cylinders")
colfill <- c(2:(2 + length(levels(cyl.f))))
# cat("Use mouse to place legend...", "\n\n")
legend('topright', levels(cyl.f), fill = colfill)
detach(mtcars) #locator(1)
par(lwd = 1)
```

box plot




```
[1] 10.40 15.35 19.20 22.80 33.90
```

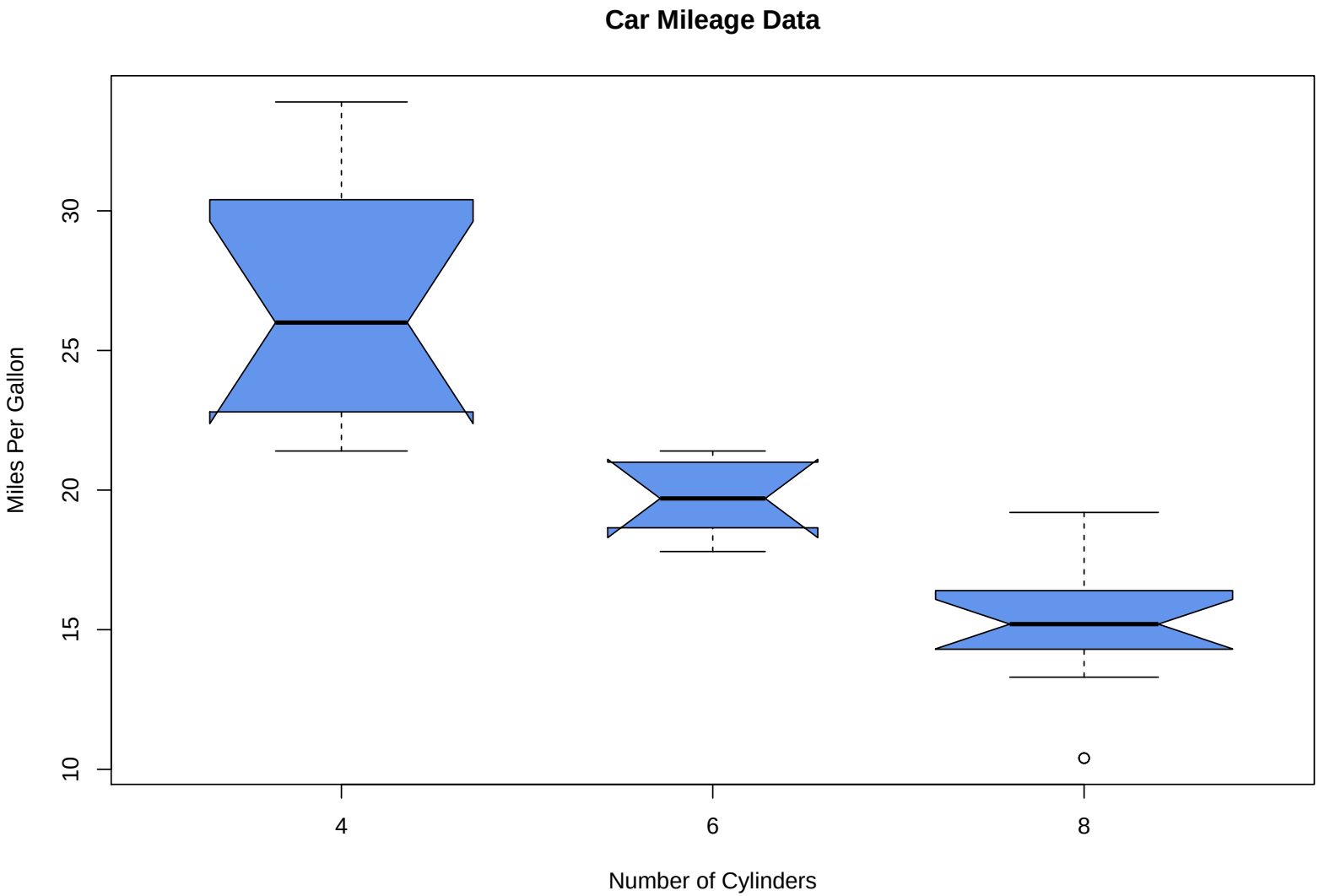
box plot


Car Milage Data
Miles Per Gallon
10
15
20
25
30
4
6
8
Number of Cylinders

box plot

```
boxplot(mpg ~ cyl, data = mtcars,
        main = "Car Milage Data",
        xlab = "Number of Cylinders",
        ylab = "Miles Per Gallon")
```

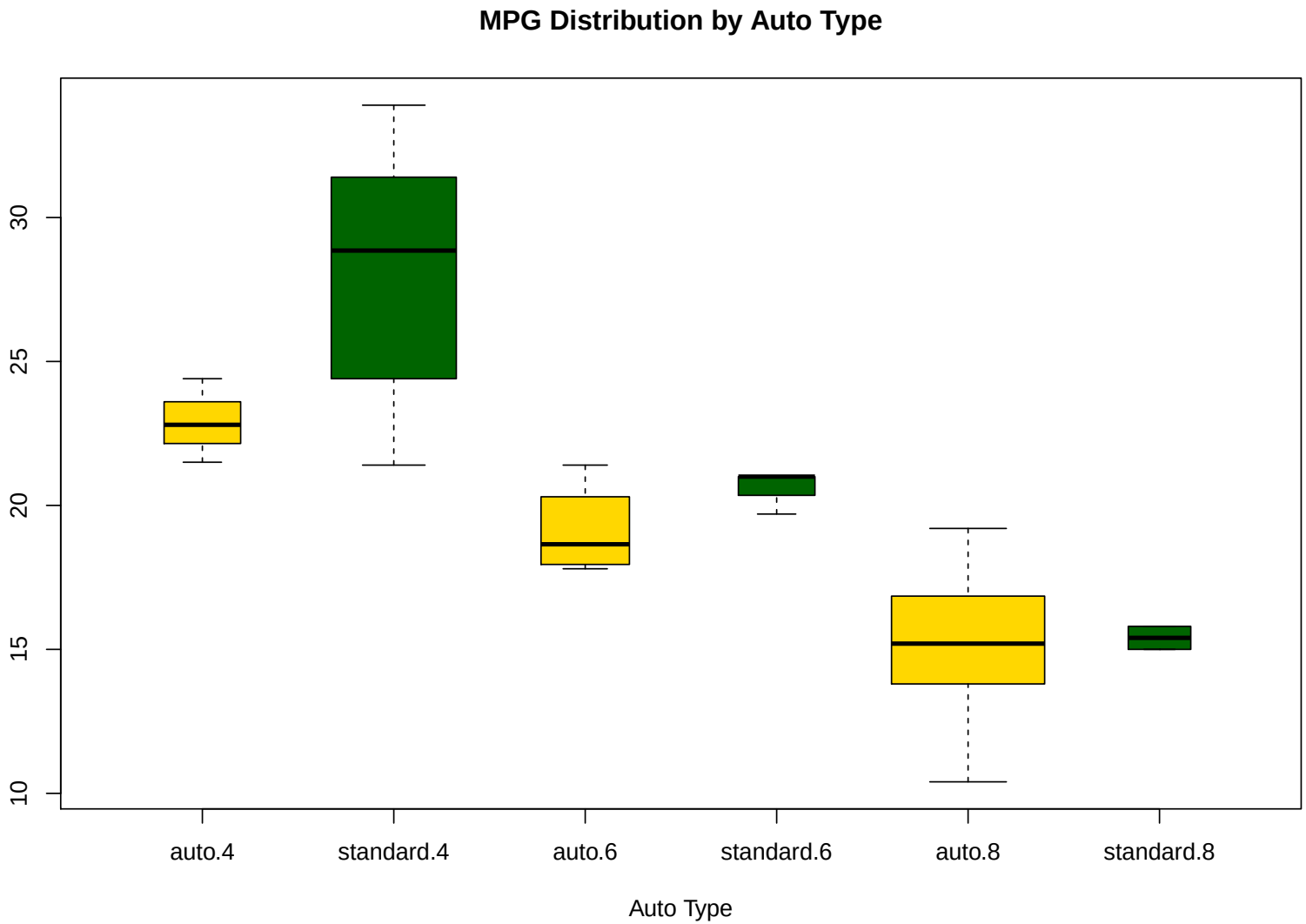
box plot


Car Mileage Data
Miles Per Gallon
10
15
20
25
30
4
6
8
Number of Cylinders

box plot

```
boxplot(mpg ~ cyl, data = mtcars, notch = T,
        varwidth = TRUE, col = "cornflowerblue",
        main = "Car Mileage Data",
        xlab = "Number of Cylinders",
        ylab = "Miles Per Gallon")
```

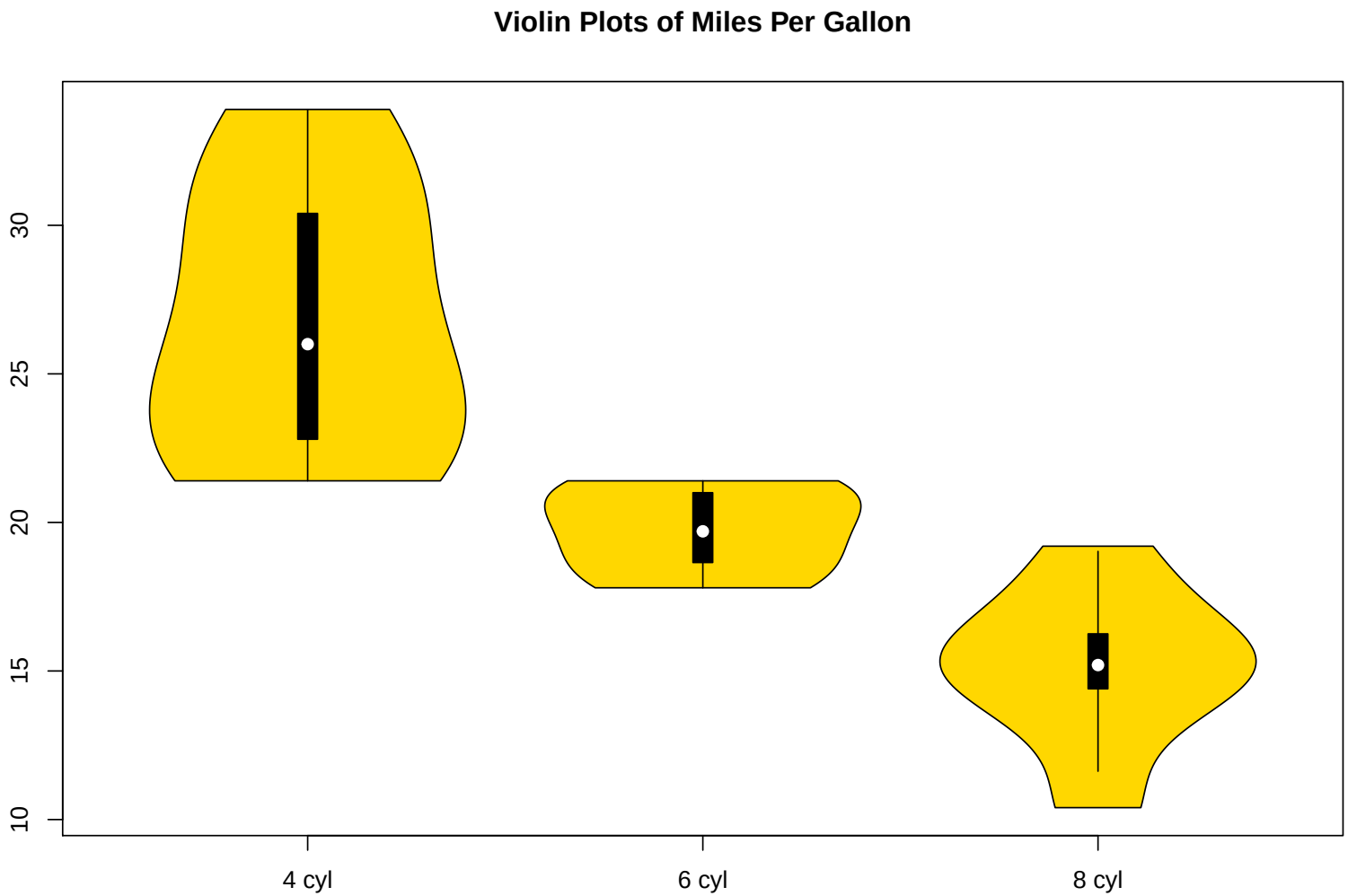
box plot


MPG Distribution by Auto Type
10
15
20
25
30
auto.4
standard.4
auto.6
standard.6
auto.8
standard.8
Auto Type

box plot

```
mtcars$cyl.f <- factor(mtcars$cyl, levels = c(4, 6,
                                              8), labels = c("4", "6", "8"))
mtcars$am.f <- factor(mtcars$am, levels = c(0, 1),
                                            labels = c("auto", "standard"))
boxplot(mpg ~ am.f * cyl.f, data = mtcars,
        varwidth = TRUE, col = c("gold", "darkgreen"),
        main = "MPG Distribution by Auto Type",
        xlab = "Auto Type")
```

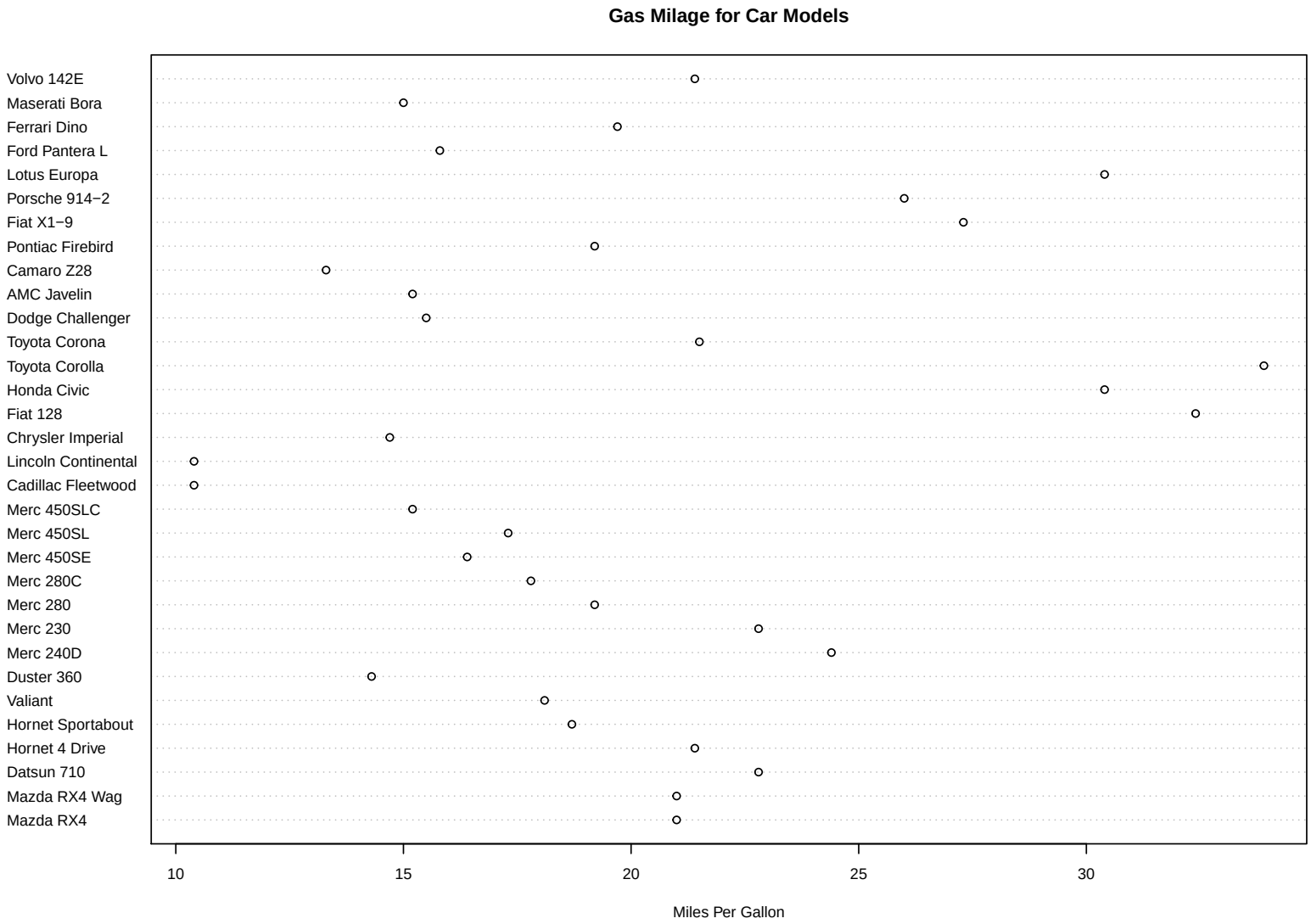
vilion plot



vilion plot

```
library(vioplot)
x1 <- mtcars$mpg[mtcars$cyl == 4]
x2 <- mtcars$mpg[mtcars$cyl == 6]
x3 <- mtcars$mpg[mtcars$cyl == 8]
vioplot(x1, x2, x3,
        names = c("4 cyl", "6 cyl", "8 cyl"),
        col = "gold")
title("Violin Plots of Miles Per Gallon")
```

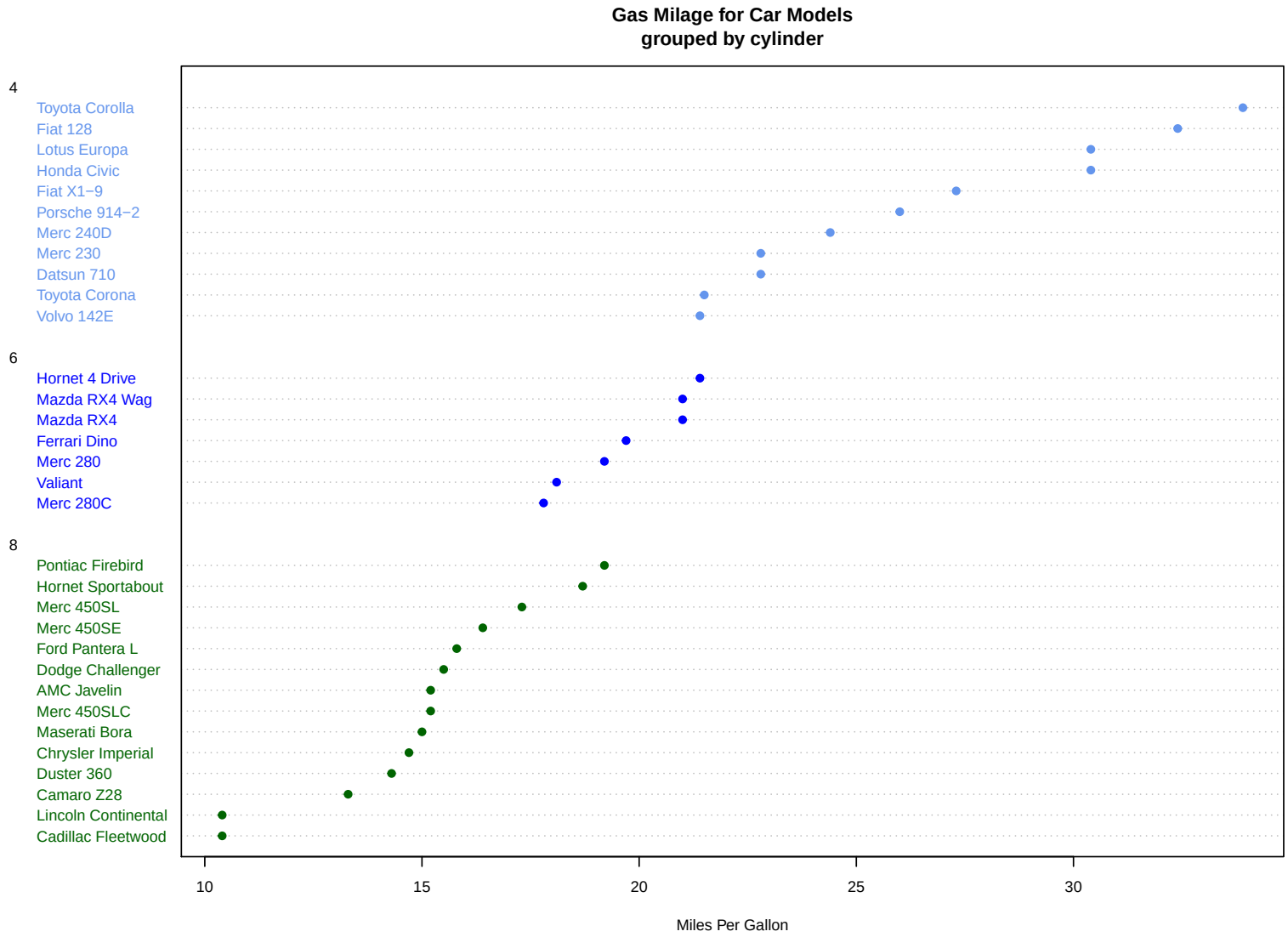
dot chart


Gas Milage for Car Models
Volvo 142E
Maserati Bora
Ferrari Dino
Ford Pantera L
Lotus Europa
Porsche 914−2
Fiat X1−9
Pontiac Firebird
Camaro Z28
AMC Javelin
Dodge Challenger
Toyota Corona
Toyota Corolla
Honda Civic
Fiat 128
Chrysler Imperial
Lincoln Continental
Cadillac Fleetwood
Merc 450SLC
Merc 450SL
Merc 450SE
Merc 280C
Merc 280
Merc 230
Merc 240D
Duster 360
Valiant
Hornet Sportabout
Hornet 4 Drive
Datsun 710
Mazda RX4 Wag
Mazda RX4
10
15
20
25
30
Miles Per Gallon

dot chart

```
dotchart(mtcars$mpg, labels = row.names(mtcars),
         cex = 0.7,
         main = "Gas Milage for Car Models",
         xlab = "Miles Per Gallon")
```

dot chart


Gas Milage for Car Models
grouped by cylinder
4
Toyota Corolla
Fiat 128
Lotus Europa
Honda Civic
Fiat X1−9
Porsche 914−2
Merc 240D
Merc 230
Datsun 710
Toyota Corona
Volvo 142E
6
Hornet 4 Drive
Mazda RX4 Wag
Mazda RX4
Ferrari Dino
Merc 280
Valiant
Merc 280C
8
Pontiac Firebird
Hornet Sportabout
Merc 450SL
Merc 450SE
Ford Pantera L
Dodge Challenger
AMC Javelin
Merc 450SLC
Maserati Bora
Chrysler Imperial
Duster 360
Camaro Z28
Lincoln Continental
Cadillac Fleetwood
10
15
20
25
30
Miles Per Gallon

dot chart

```
x <- mtcars[order(mtcars$mpg), ]
x$cyl <- factor(x$cyl)
x$color[x$cyl == 4] <- "cornflowerblue"
x$color[x$cyl == 6] <- "blue"
x$color[x$cyl == 8] <- "darkgreen"
dotchart(x$mpg, labels = row.names(x), cex = 0.7,
         pch = 19, groups = x$cyl,
         color = x$color,
         main = "Gas Milage for Car Models\n
         grouped by cylinder",
         xlab = "Miles Per Gallon")
```

categorical and quantitative data

- categorical data:

bar chart*, pie chart

- quantitative data:

histogram, box plot, dot plot