chapter 6

Simple plots using R

Instructor: Li, Han

Contents

- box plot
- histogram
- density plot
- bar plot
- pie plot
- dot char

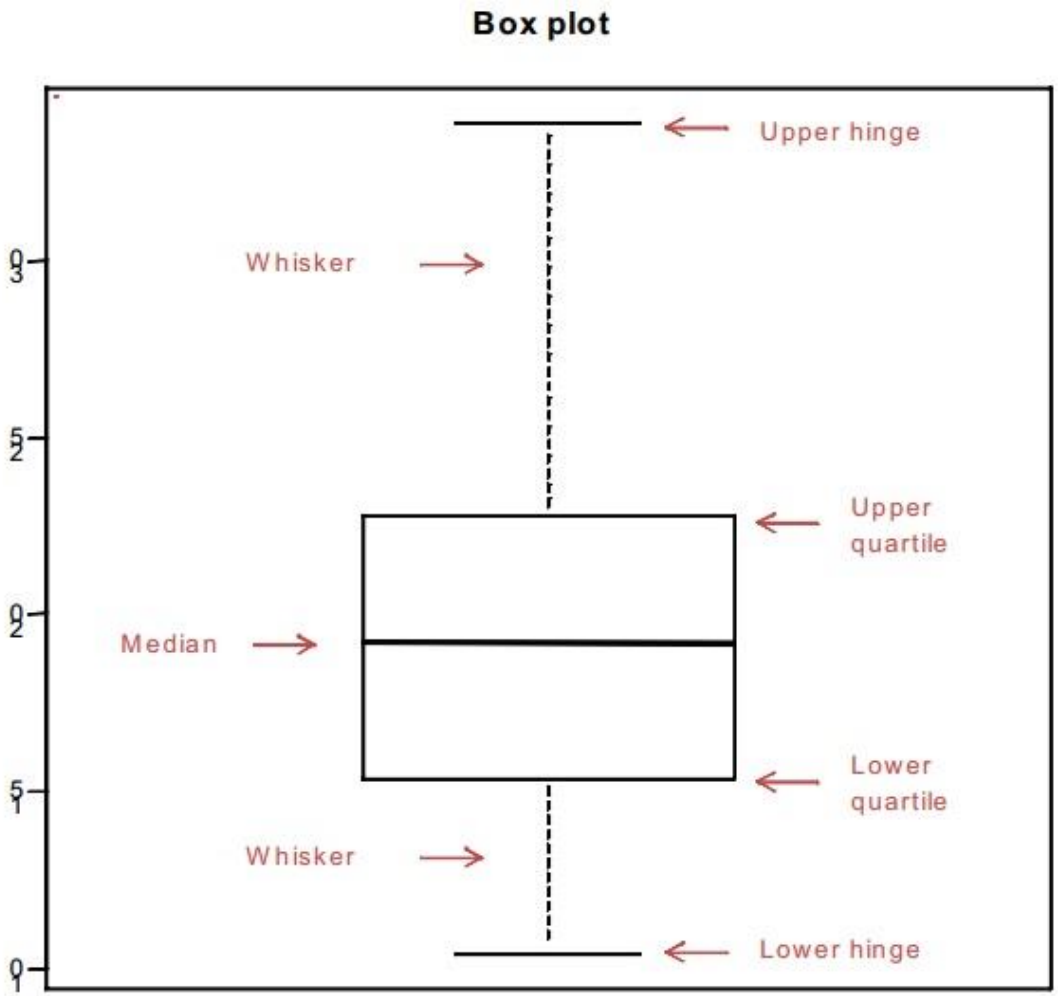
data example

	mpg	cyl	am
Mazda RX4	21.0	6	1
Mazda RX4 Wag	21.0	6	1
Toyota Corolla	33.9	4	1
Merc 280C	17.8	6	0
Ford Pantera L	15.8	8	1
Ferrari Dino	19.7	6	1
...	...	...	...

Box plot

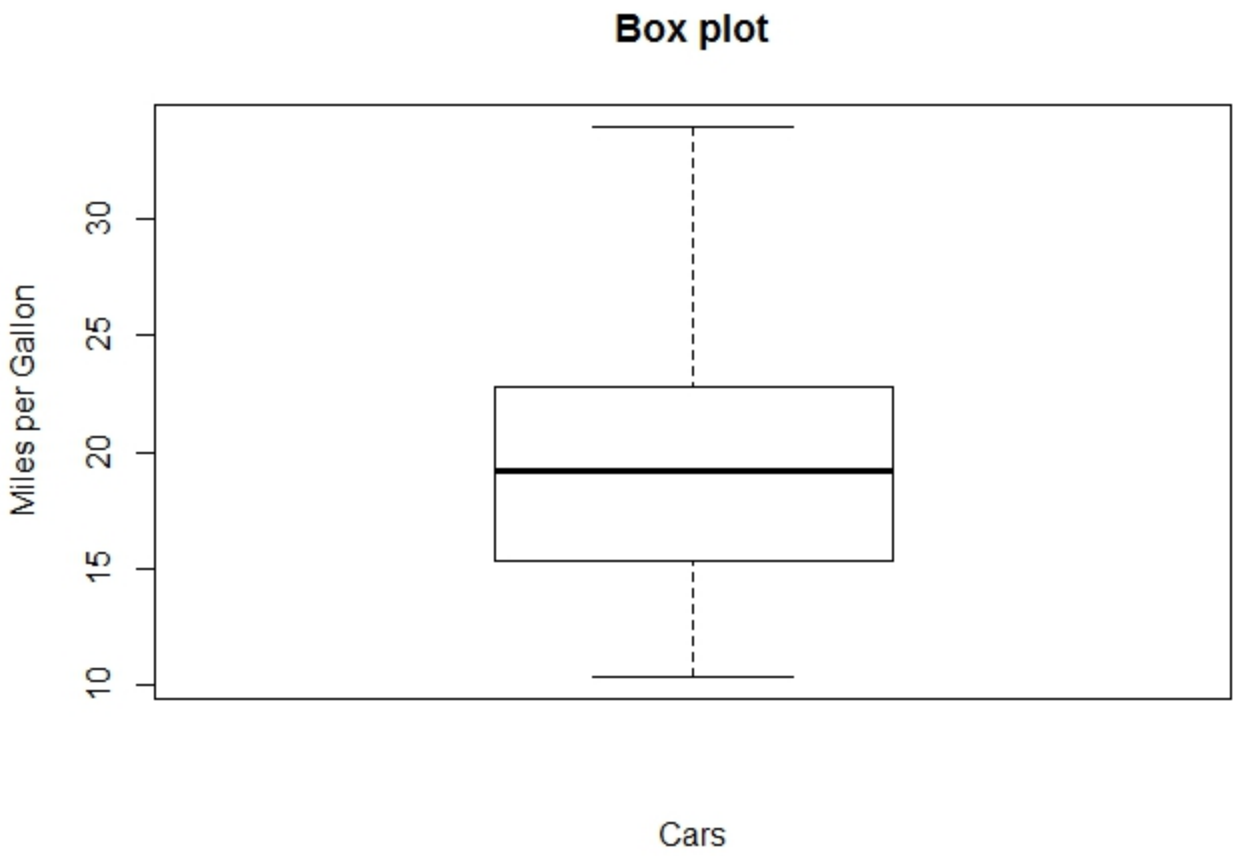
A “box-and-whiskers” plot describes the distribution of a continuous variable by plotting its five-number summary:

- maximum
- upper quartile (75th percentile)
- Median (50th percentile)
- lower quartile (25th percentile)
- minimum


Box plot
Upper hinge
Whisker
Upper quartile
Median
Lower quartile
Whisker
Lower hinge

Example 1 (box plot)

```
attach(mtcars)
boxplot(mpg, main="Box plot", xlab="Cars", ylab="Miles per Gallon")
```

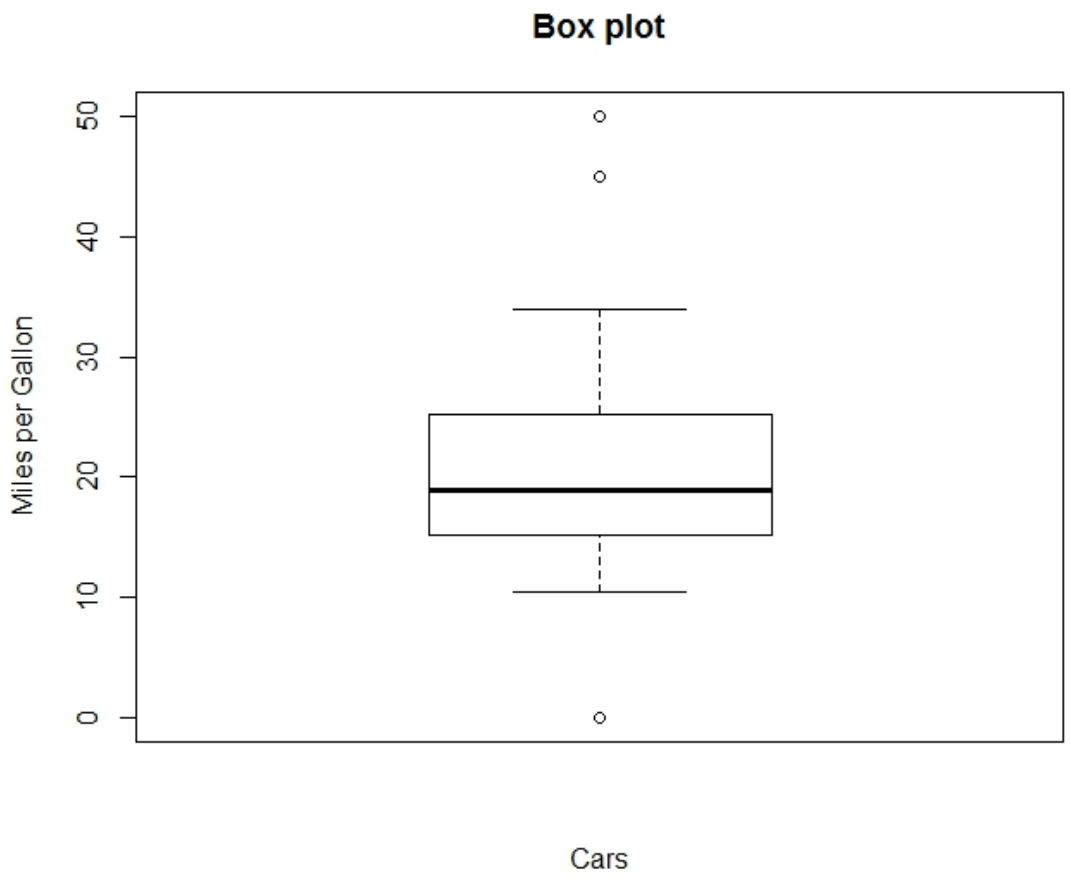

Box plot
Miles per Gallon
10
15
20
25
30
Cars

If the data have outliers

We change the data to have outliers by setting

```
mpg <- mtcars$mpg
mpg[1,1]=50
mpg[2,1]=45
mpg[32,1]=0
```

Then the box plot has some changes.


Box plot
Miles per Gallon
0
10
20
30
40
50
Cars

Box plots by group

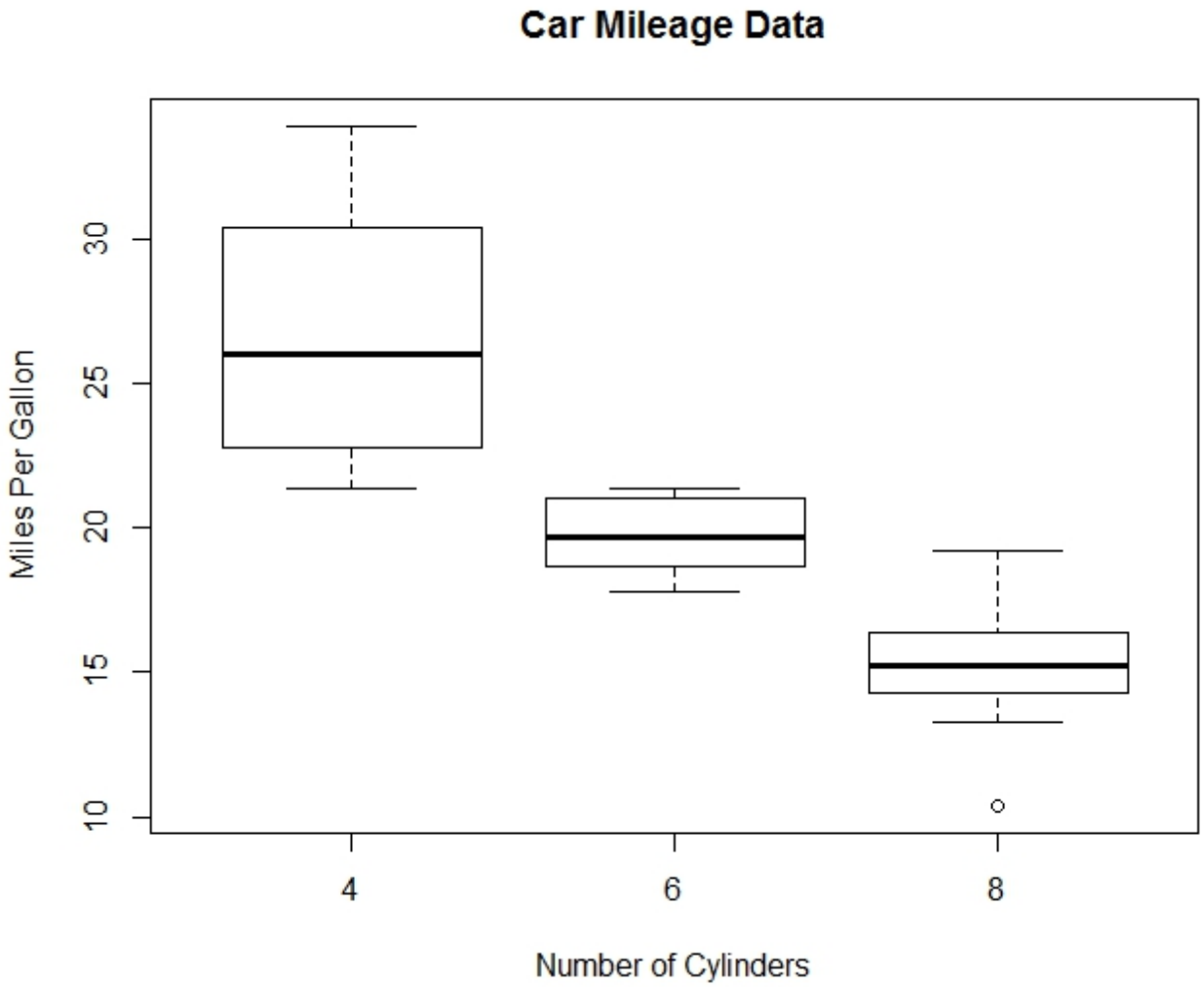
Box plots can be created for multiple variables by group. The format is

boxplot(formula, data=dataframe, main=…)

formula: y ~ A, y is the whole data, A is a categorical variable, the elements of y and A are one-to-one match. Then we have box plots for each category of A.

Example 2(box plot by group)

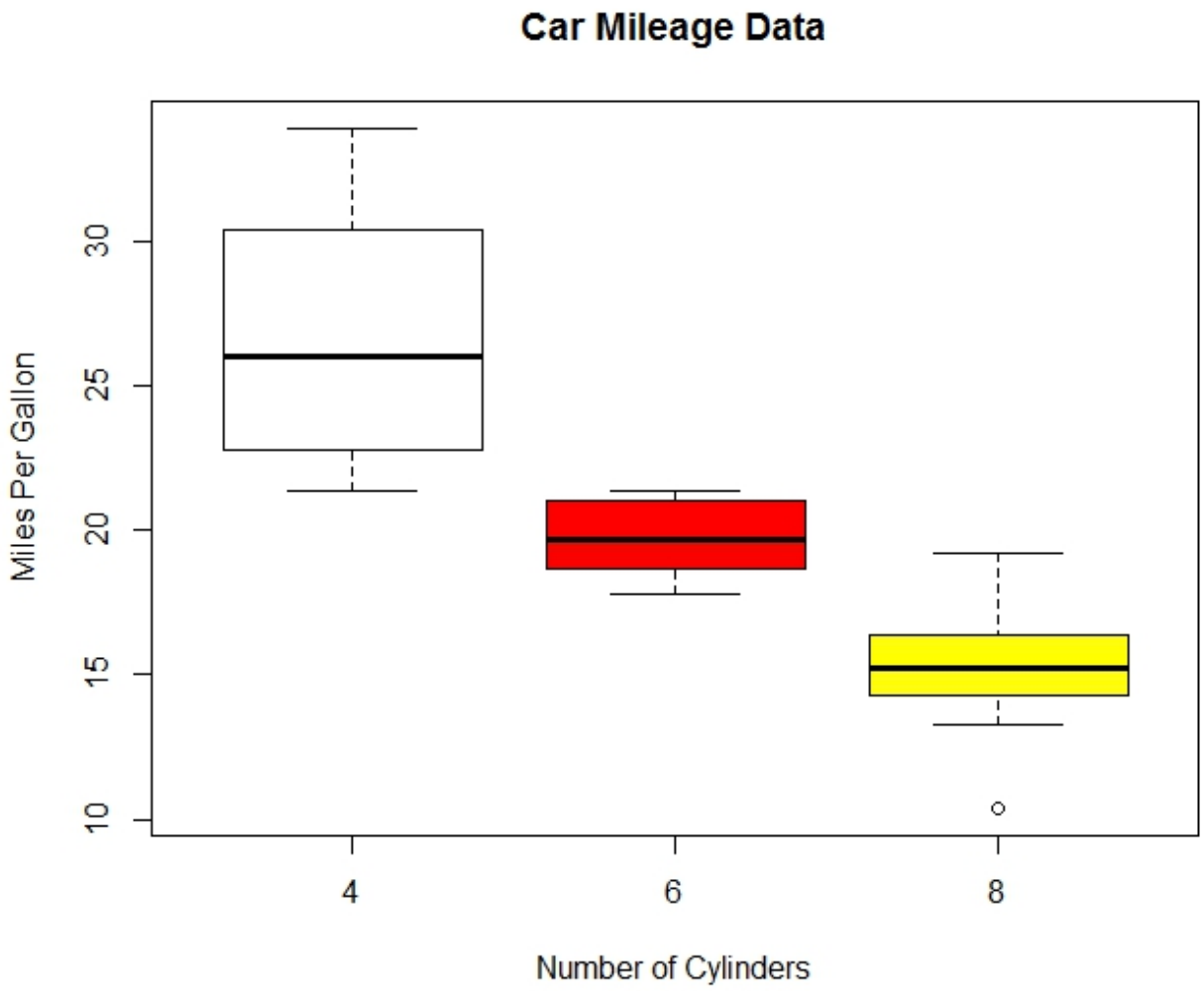
```
mpg <- mtcars$mpg
cyl <- mtcars$cyl
boxplot(mpg ~ cyl, data=mtcars,
main="Car Mileage Data",
xlab="Number of Cylinders",
ylab="Miles Per Gallon")
```


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

Change the color of the plot

Example 3

```
boxplot(mpg ~ cyl, data=mtcars,
main="Car Mileage Data",
xlab="Number of Cylinders",
ylab="Miles Per Gallon",
col=c("white","red","yellow"))
```


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

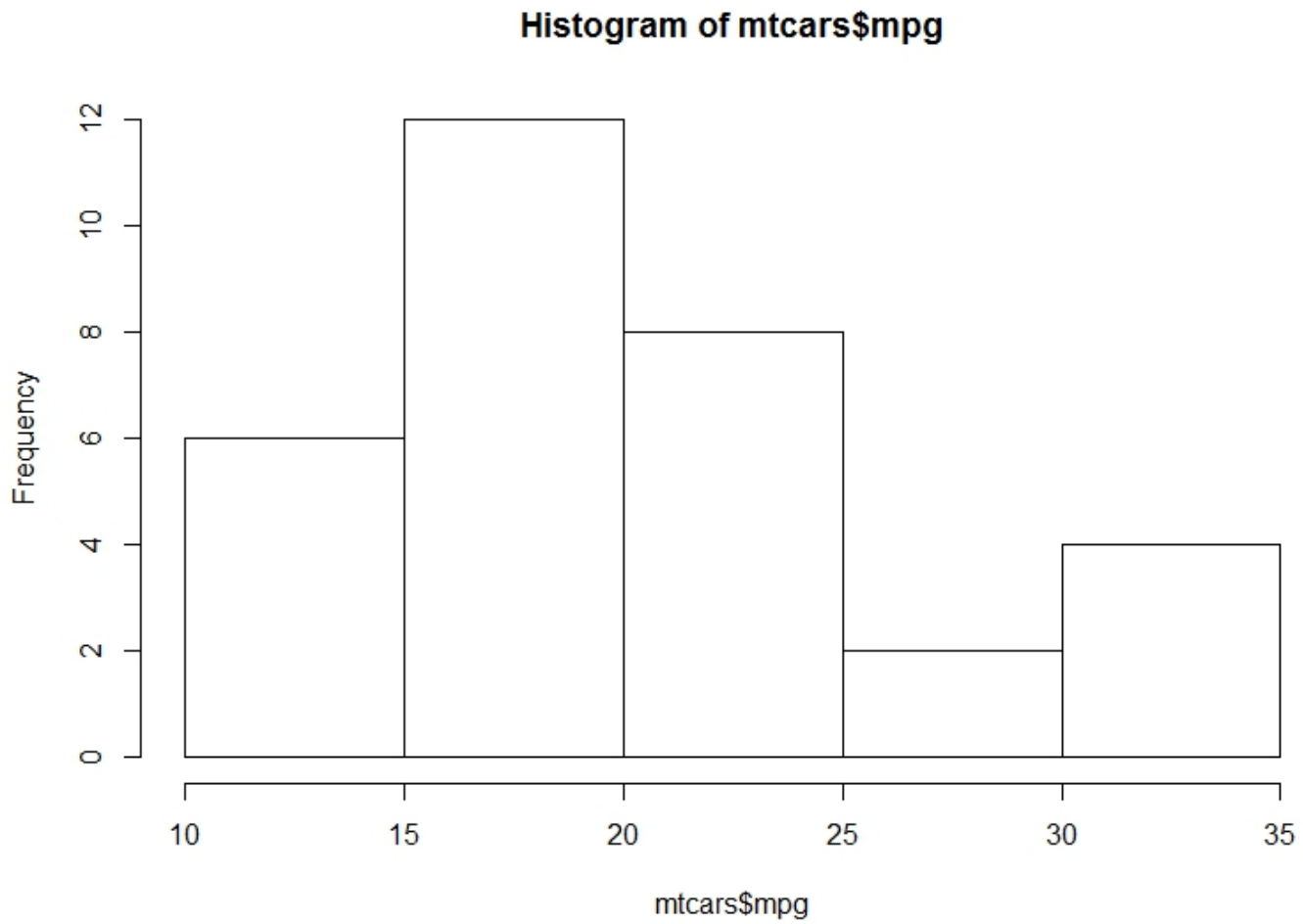
Histogram

Histogram displays the distribution of a continuous variable by dividing it into several bins.

hist(x, breaks=number of bins, main=…)

Example 4 (histogram)

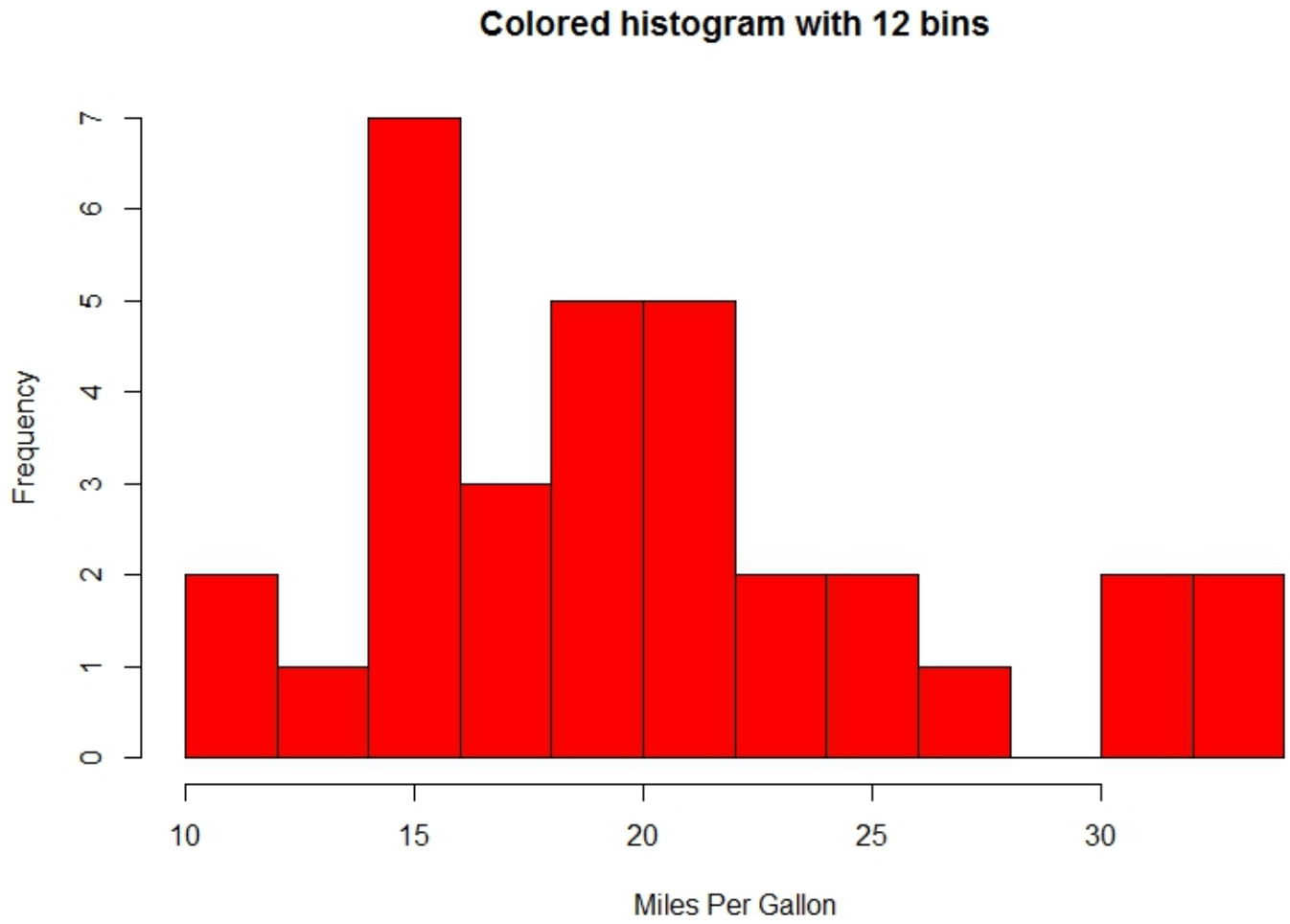
hist(mtcars$mpg)


Histogram of mtcars$mpg
Frequency
0
2
4
6
8
10
12
10
15
20
25
30
35
mtcars$mpg

Specify the number of bins

Example 5

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


Colored histogram with 12 bins
Frequency
0
1
2
3
4
5
6
7
10
15
20
25
30
Miles Per Gallon

Density plot

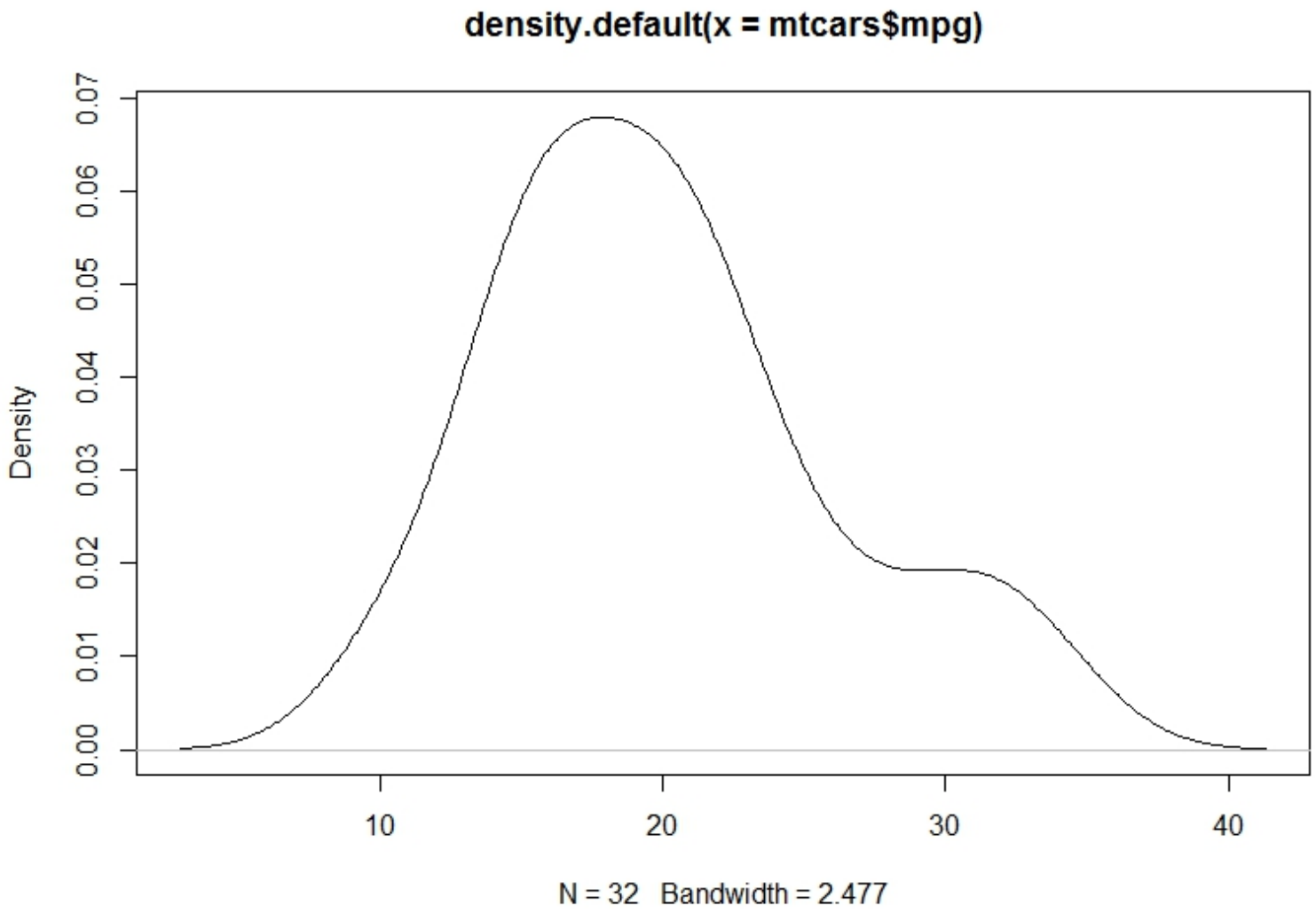
Density plot is a nonparametric method for estimating the probability density function of a random variable.

Compare with histogram, which is discrete, density plot is a continous approximation of the distribution of the data.

Example 6

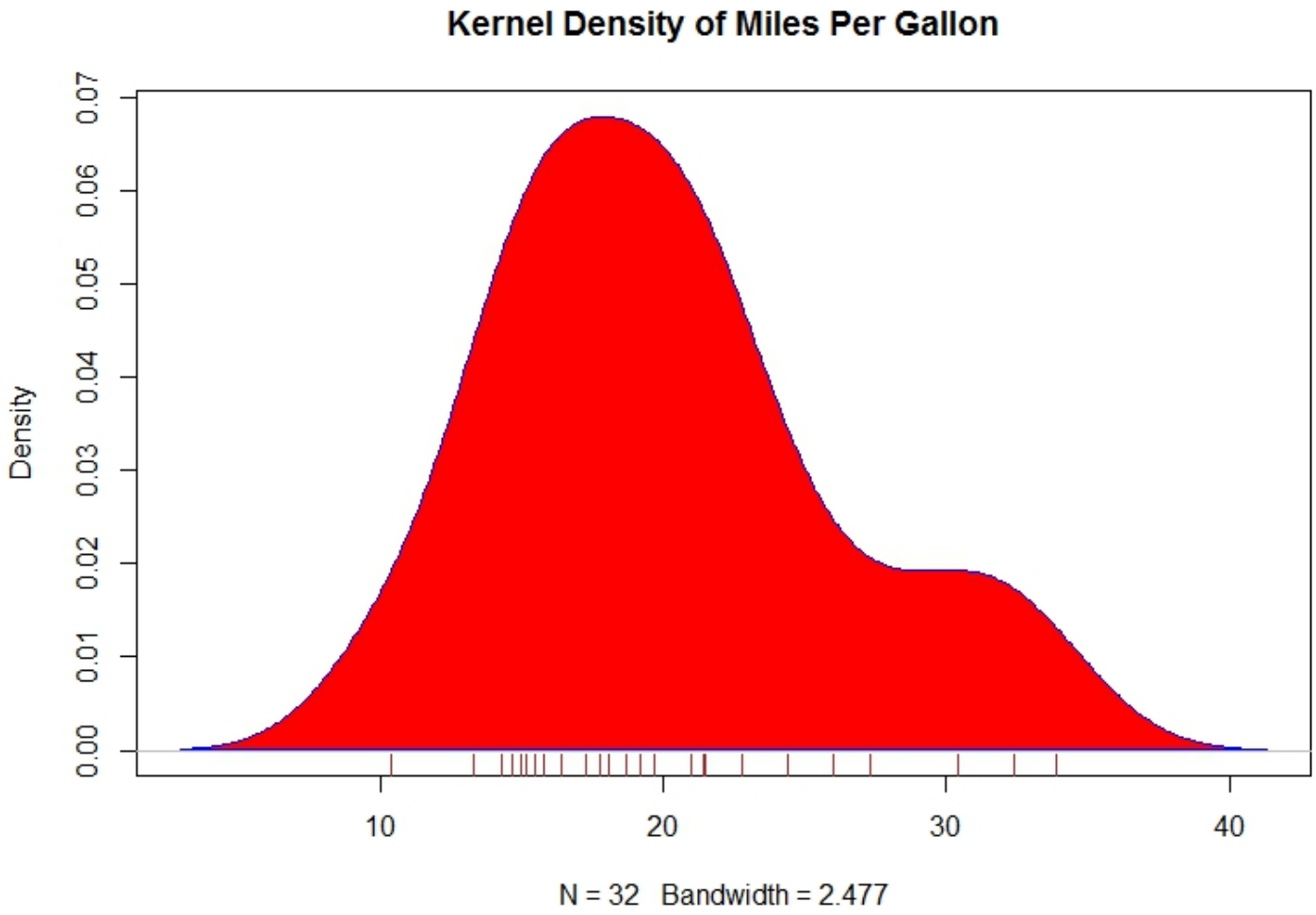
```
plot(density(mtcars$mpg))
```

If the plot already exists, we use lines(density(x)) to impose a density line in the original plot.


density.default(x = mtcars$mpg)
Density
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
10
20
30
40
N = 32 Bandwidth = 2.477

Example 7

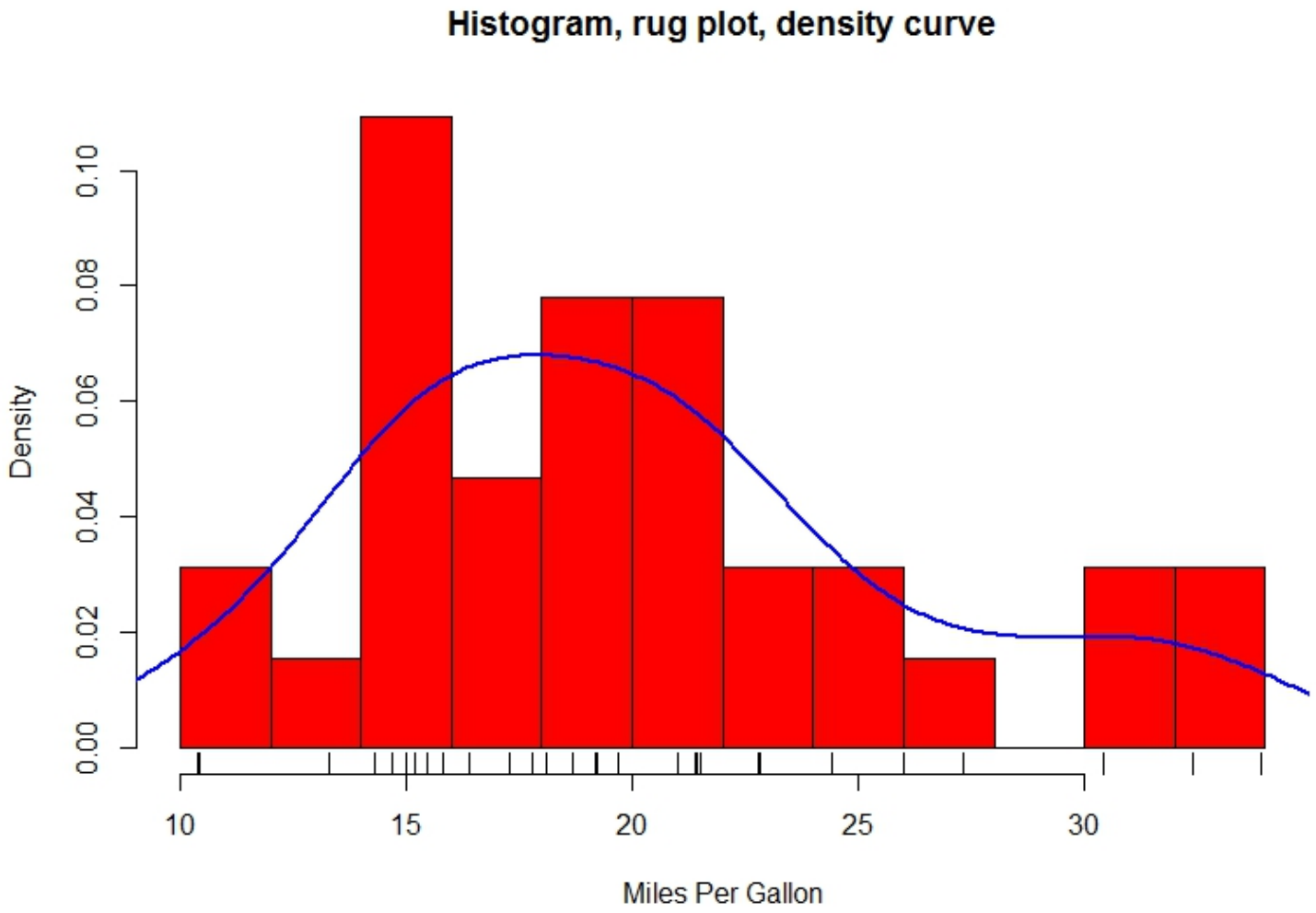
```
d <- density(mtcars$mpg)
plot(d, main="Kernel Density of Miles Per Gallon")
polygon(d, col="red", border="blue")
rug(mtcars$mpg, col="brown")
```


Kernel Density of Miles Per Gallon
Density
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
10
20
30
40
N = 32 Bandwidth = 2.477

Add density plot to histogram

Example 8

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


Histogram, rug plot, density curve
Density
0.00
0.02
0.04
0.06
0.08
0.10
10
15
20
25
30
Miles Per Gallon

Bar plot

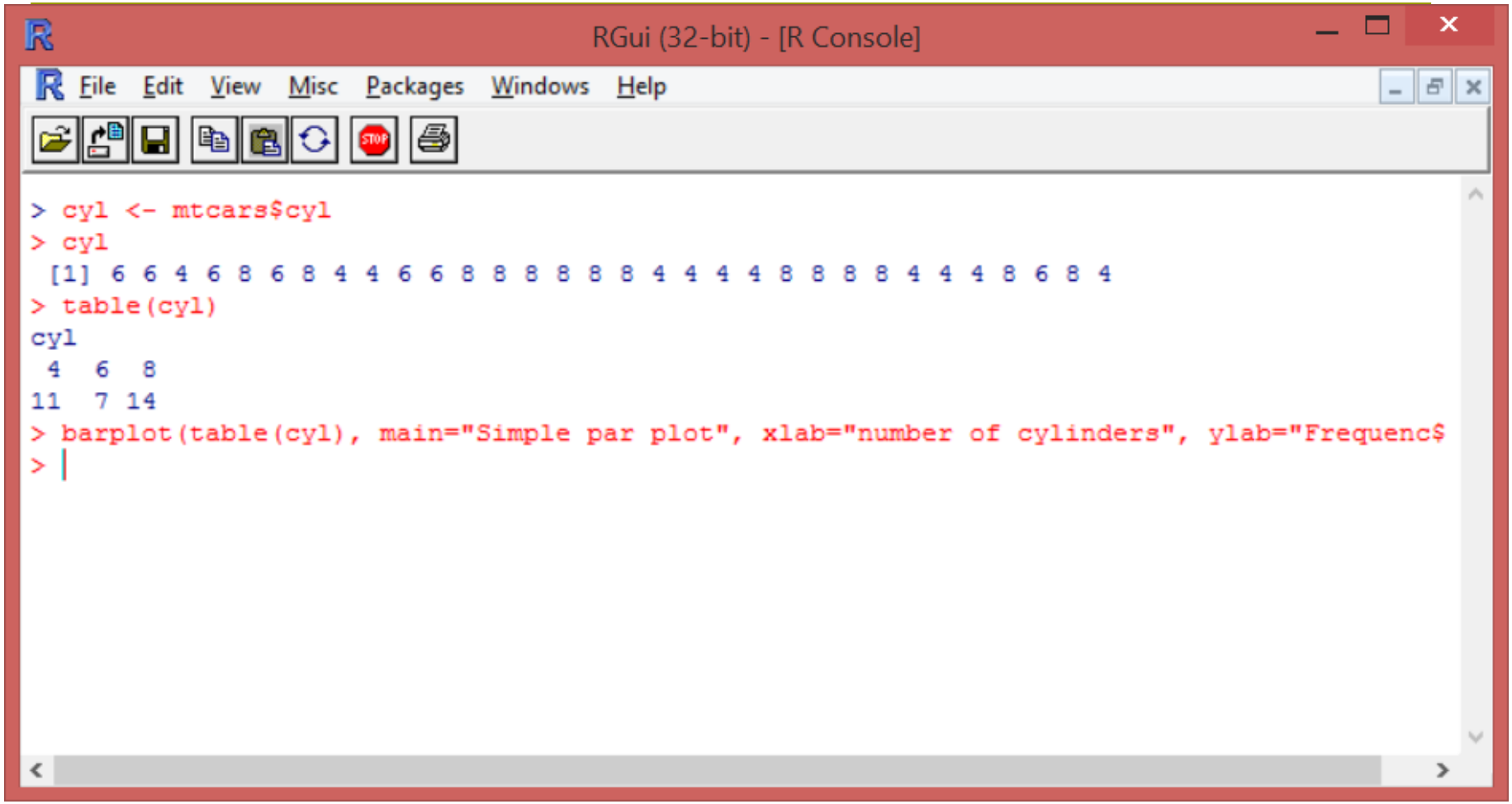
Bar plot displays the distribution (frequencies) of a categorical variable through vertical or horizontal bars.

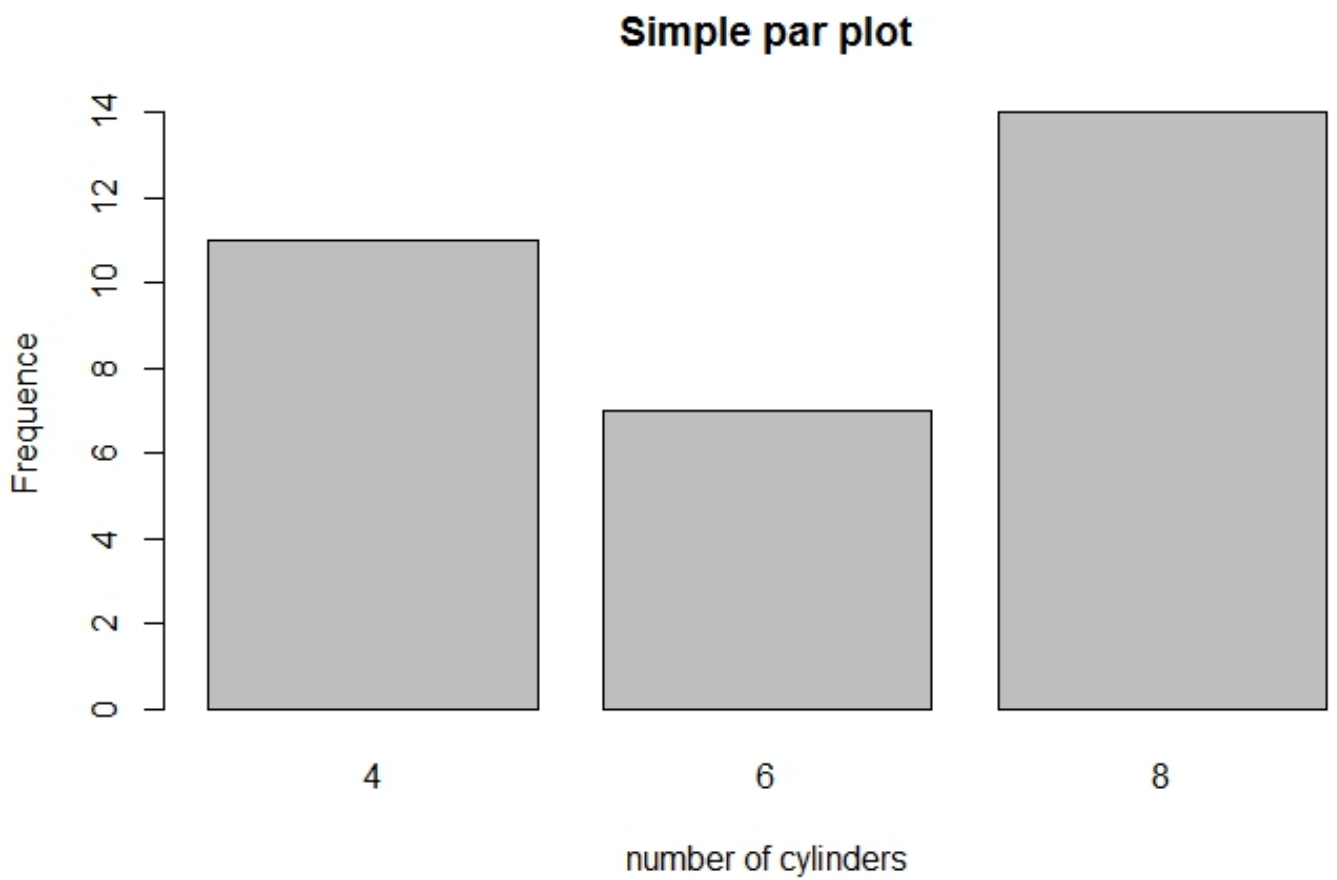
barplot(x)

Example 8 (bar plot)

```
cyl <- mtcars$cyl
cyl
table(cyl)
barplot(table(cyl), main="Simple par plot", xlab="number of cylinders", ylab="Frequence")
```

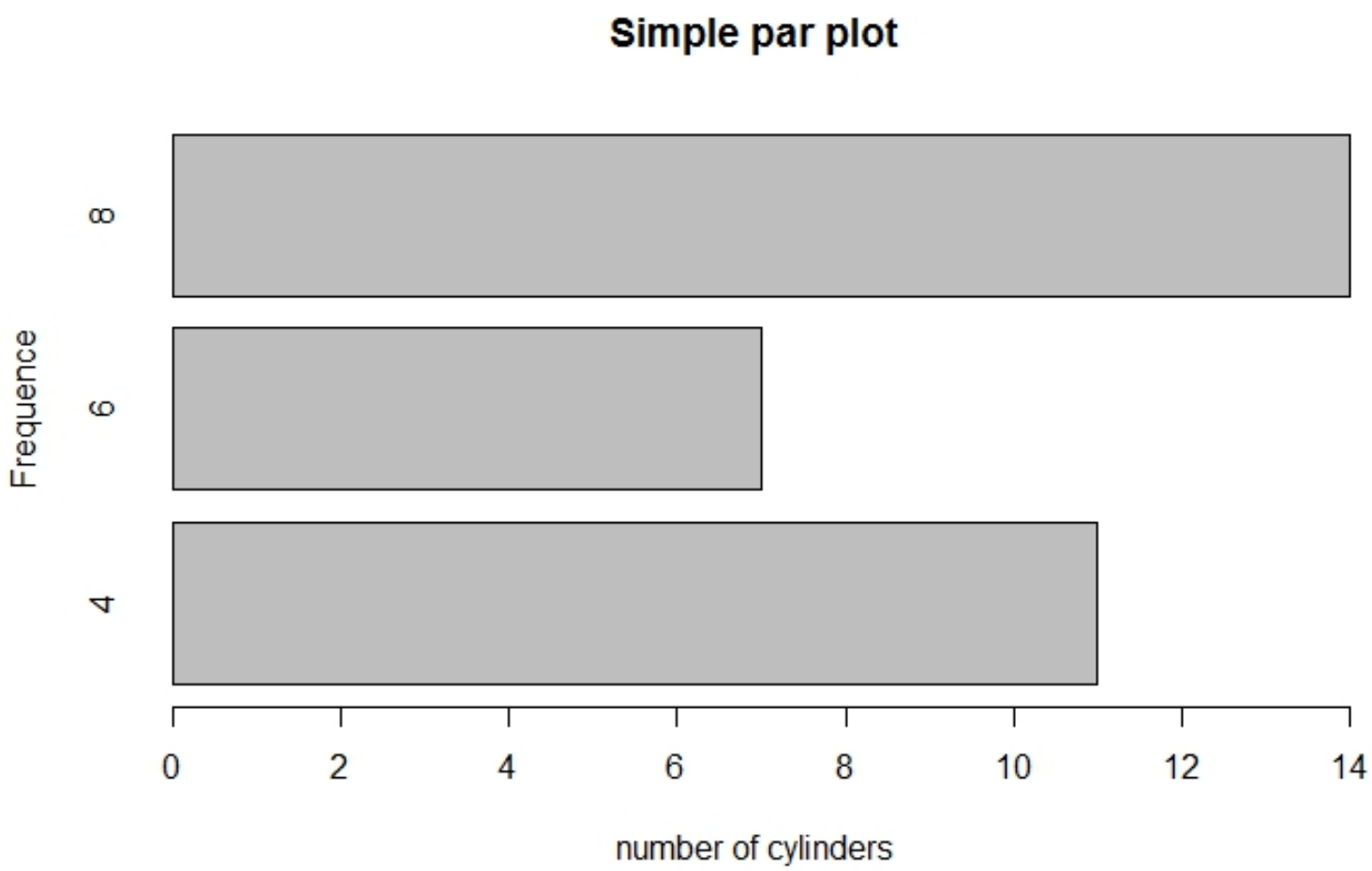
table(x): summarize the counts in each category


RGui (32-bit) - [R Console]
File Edit View Misc Packages Windows Help
> cyl <- mtcars$cyl
> cyl
[1] 6 6 4 6 8 6 8 4 4 6 6 8 8 8 8 8 8 4 4 4 4 8 8 8 8 4 4 4 8 6 8 4
> table(cyl)
cyl
4 6 8
11 7 14
> barplot(table(cyl), main="Simple par plot", xlab="number of cylinders", ylab="Frequenc$
>


Simple par plot
Frequence
0
2
4
6
8
10
12
14
4
6
8
number of cylinders

Example 9

```
barplot(table(cyl), horiz=TRUE,
main="Simple par plot", xlab="number of cylinders",
ylab="Frequence")
```


Simple par plot
Frequence
8
6
4
0
2
4
6
8
10
12
14
number of cylinders

Multiple bar plots

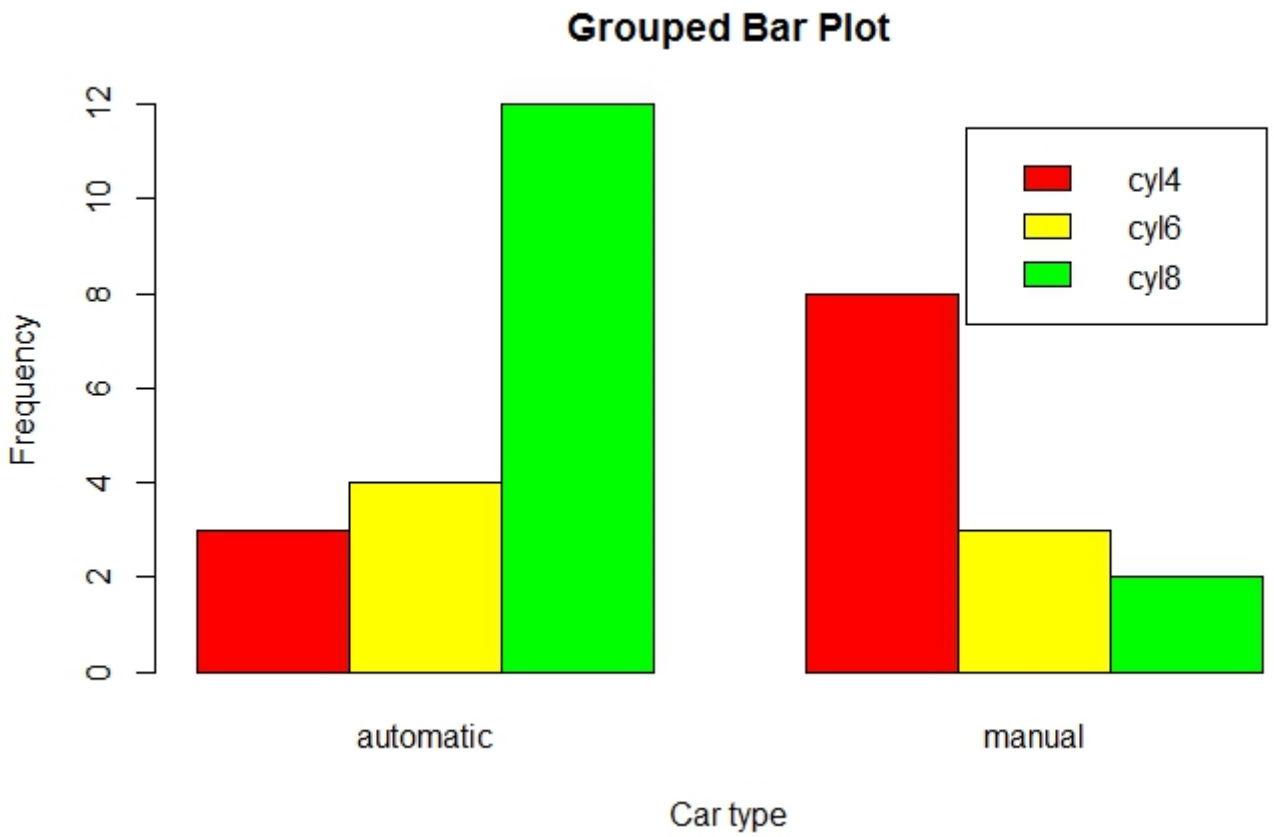
am= 0, automatic,

am=1, manual.

```
counts <- table(mtcars$cyl, mtcars$am)
rownames(counts)=c("cyl4","cyl6","cyl8")
colnames（counts)=c("automatic","manual")
```

Example 10

```
barplot(counts,
main="Grouped Bar Plot",
xlab="Car type", ylab="Frequency",
col=c("red", "yellow", "green"),
legend=rownames(counts), beside=TRUE)
```

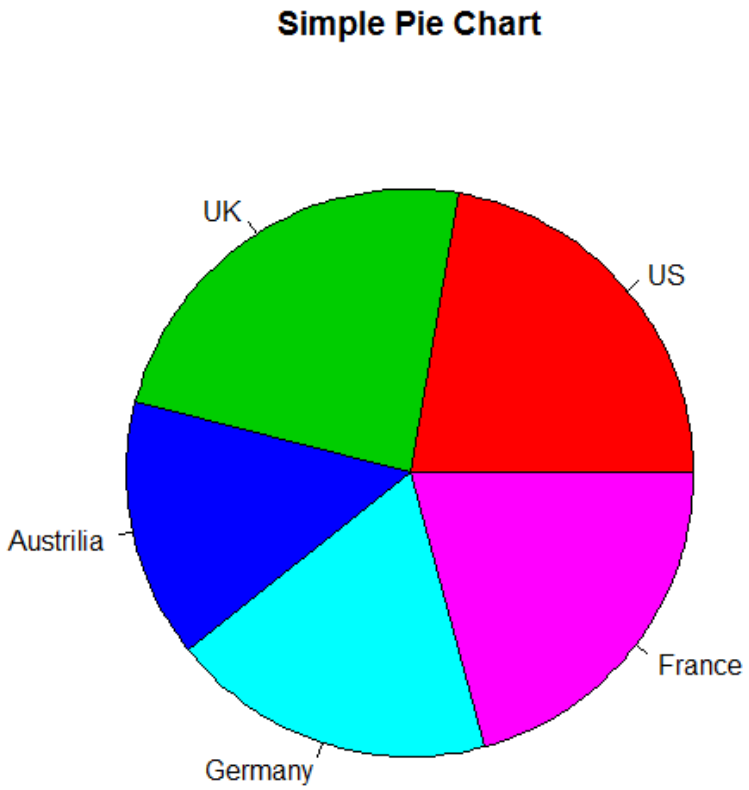

Grouped Bar Plot
Frequency
0
2
4
6
8
10
12
cyl4
cyl6
cyl8
automatic
manual
Car type

Pie chart

pie(x, labels=?,...)

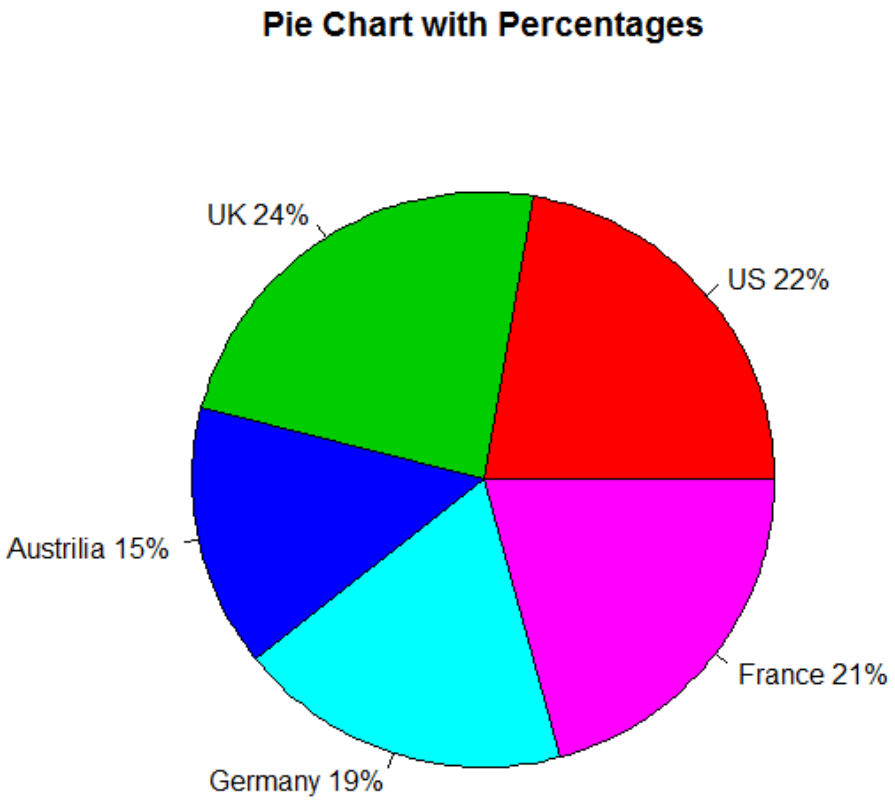
Example 11 (pie plot)

```
users <- c(3000, 3200, 2000, 2500, 2800)
country <- c("US", "UK", "Austrilia", "Germany", "France")
pie(users, labels=country, col=2:6, main="Simple Pie Chart")
```


Simple Pie Chart
UK
US
Austrilia
France
Germany

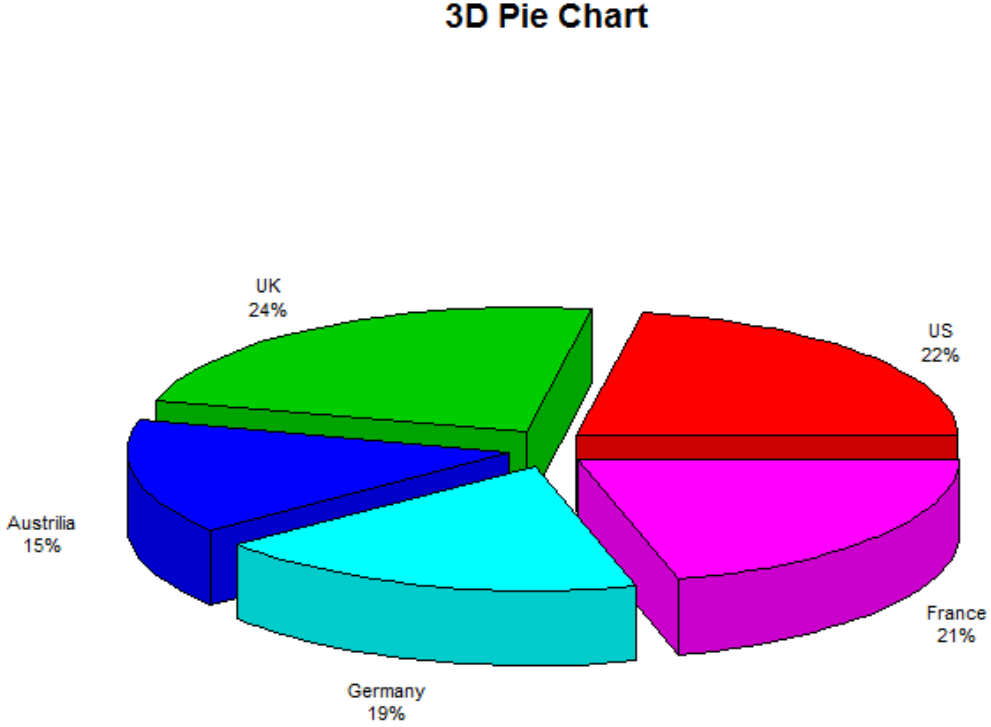
Example 12

```
#caculate the percentage
pct <- round(users/sum(users)*100)

#use paste to combine the characters
country2=paste(country, " ", pct, "%", sep="")

#pie plot
pie(users, labels=country2, col=2:6, main="Pie Chart with Percentages")
```


Pie Chart with Percentages
UK 24%
US 22%
Austrilia 15%
France 21%
Germany 19%

Example 13 (3D pie plot)

```
library(plotrix)
country3=paste(country, "\n", pct, "%", sep="")
pie3D(users, labels=country3, explode=0.1, col=2:6,
labelcex=0.7, main="3D Pie Chart")
```


3D Pie Chart
UK
24%
US
22%
Austrilia
15%
France
21%
Germany
19%

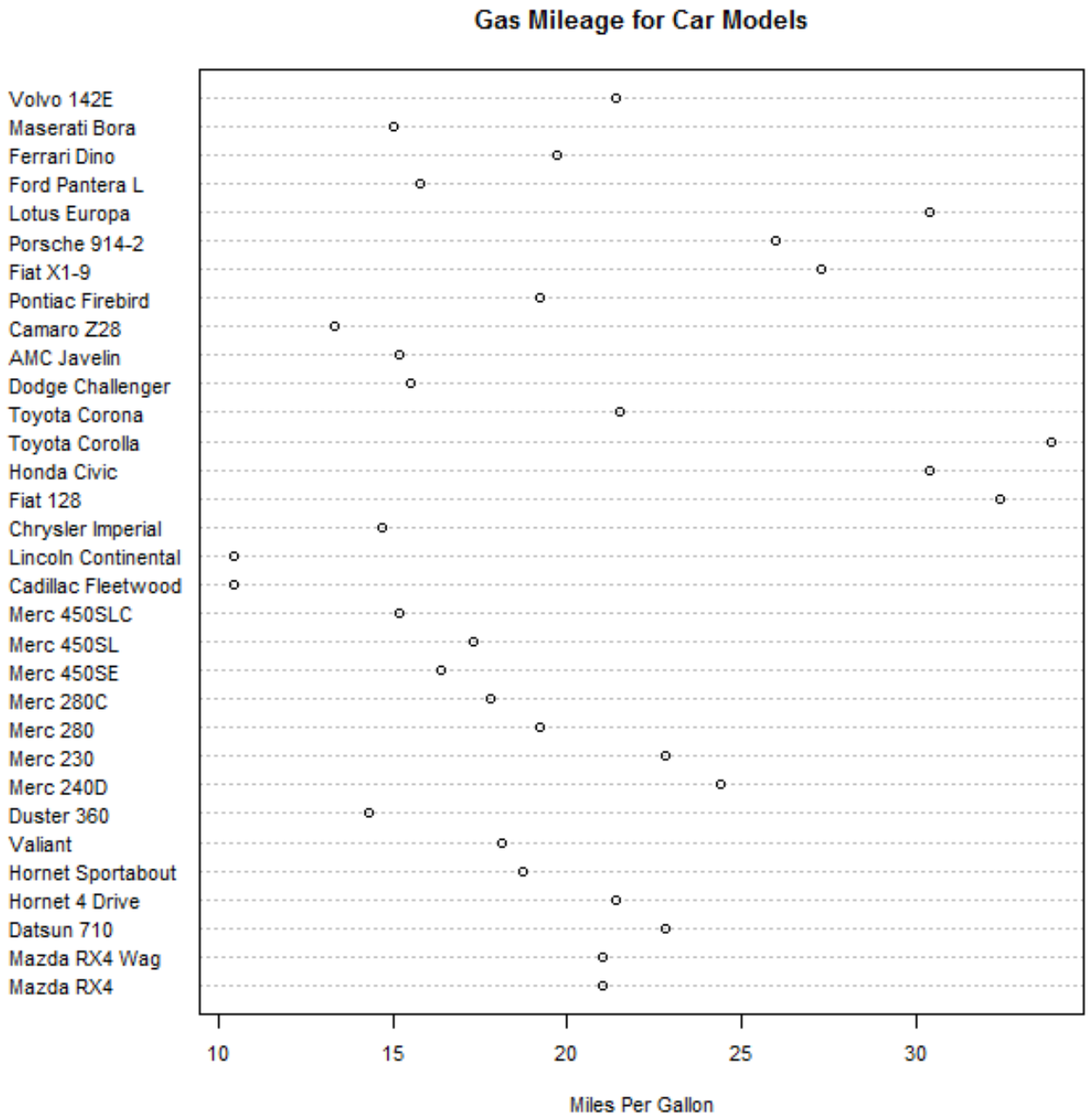
Dot plot

Dot plots provide a method of plotting a large number of labeled values on a simple horizontal scale.

dotchart(x, labels=?,...)

Example 14 (dot plot)

```
attach(mtcars)
dotchart(mpg, labels=row.names(mtcars), cex=.7,
main="Gas Mileage for Car Models",
xlab="Miles Per Gallon")
```


Gas Mileage 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

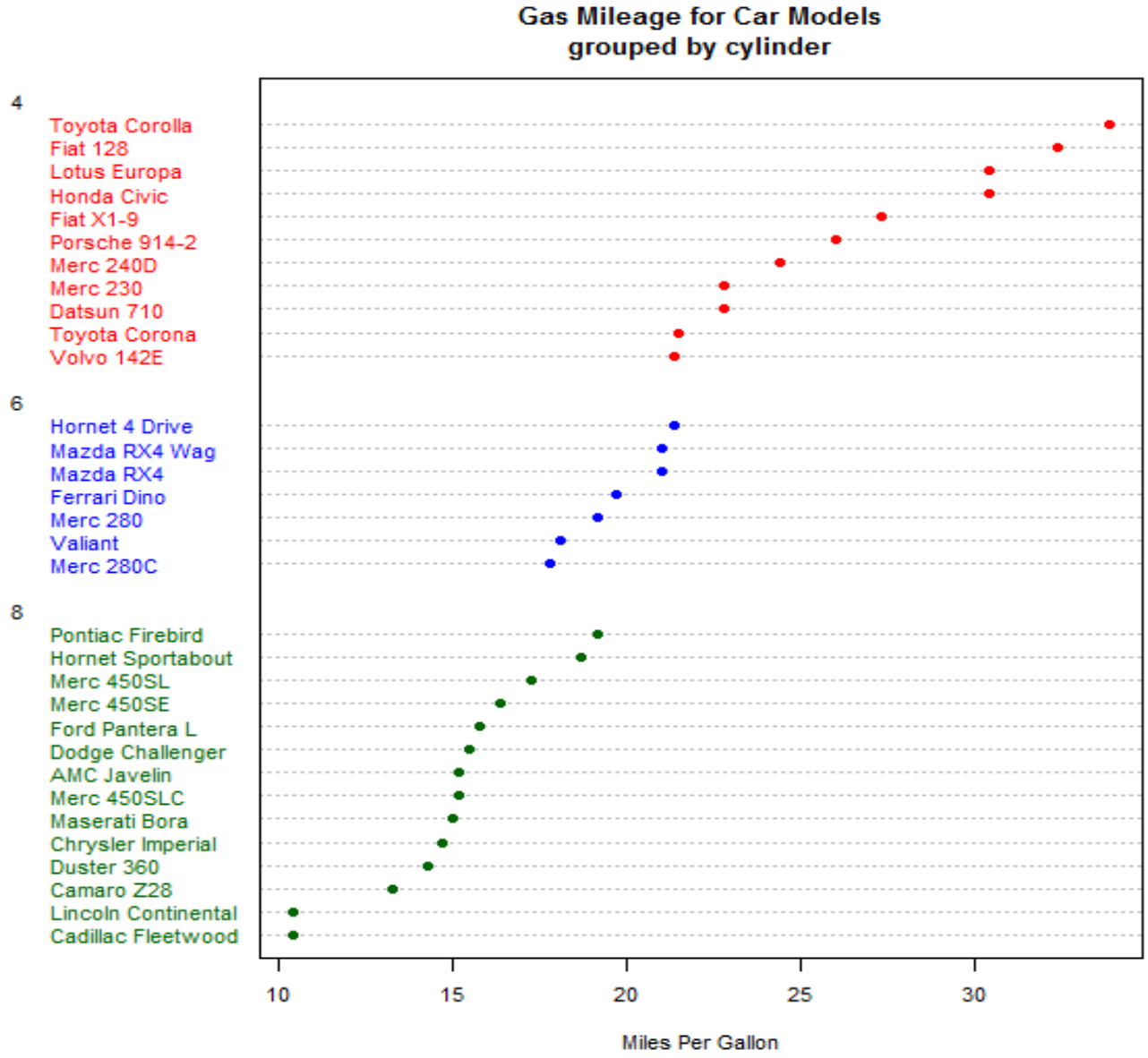
Dot plot by group

Suppose we would like to sort the mpg values in an increasing order, and then plot them in different colors based on the number of their cyl values.

Example 15

```
x <- mtcars[order(mtcars$mpg),]
x$cyl <- factor(x$cyl)
x$color[x$cyl==4] <- "red"
x$color[x$cyl==6] <- "blue"
x$color[x$cyl==8] <- "darkgreen"
```

```
dotchart(x$mpg,
labels = row.names(x),
cex=.7,
groups = x$cyl,
gcolor = "black",
color = x$color,
pch=19,
main = "Gas Mileage for Car Models\ngrouped by cylinder",
xlab = "Miles Per Gallon")
```


Gas Mileage 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

Try the examples and enjoy plotting!