Intro to plotting in R

Martin Aryee

Graphics in R

- Introduction to plotting in R
- Basic plotting functions and common options
  - Scatter (XY) plots
  - histograms
  - bar charts
- A more complex example using the dog data

Example Data

Age (years)
Weight (kg)
Height (cm)
Gender: 1=Male, 2=Female

```
> load("for_lec6.rda")
> head(people)
  age weight height gender
1  21   65.6  174.0      1
2  23   71.8  175.3      1
3  28   80.7  193.5      1
4  23   72.6  186.5      1
5  22   78.8  187.2      1
6  21   74.8  181.5      1
```

Basic plotting functions

- `plot()`
- `hist()`
- `barplot()`
- `points()`
- `lines()`

plot()

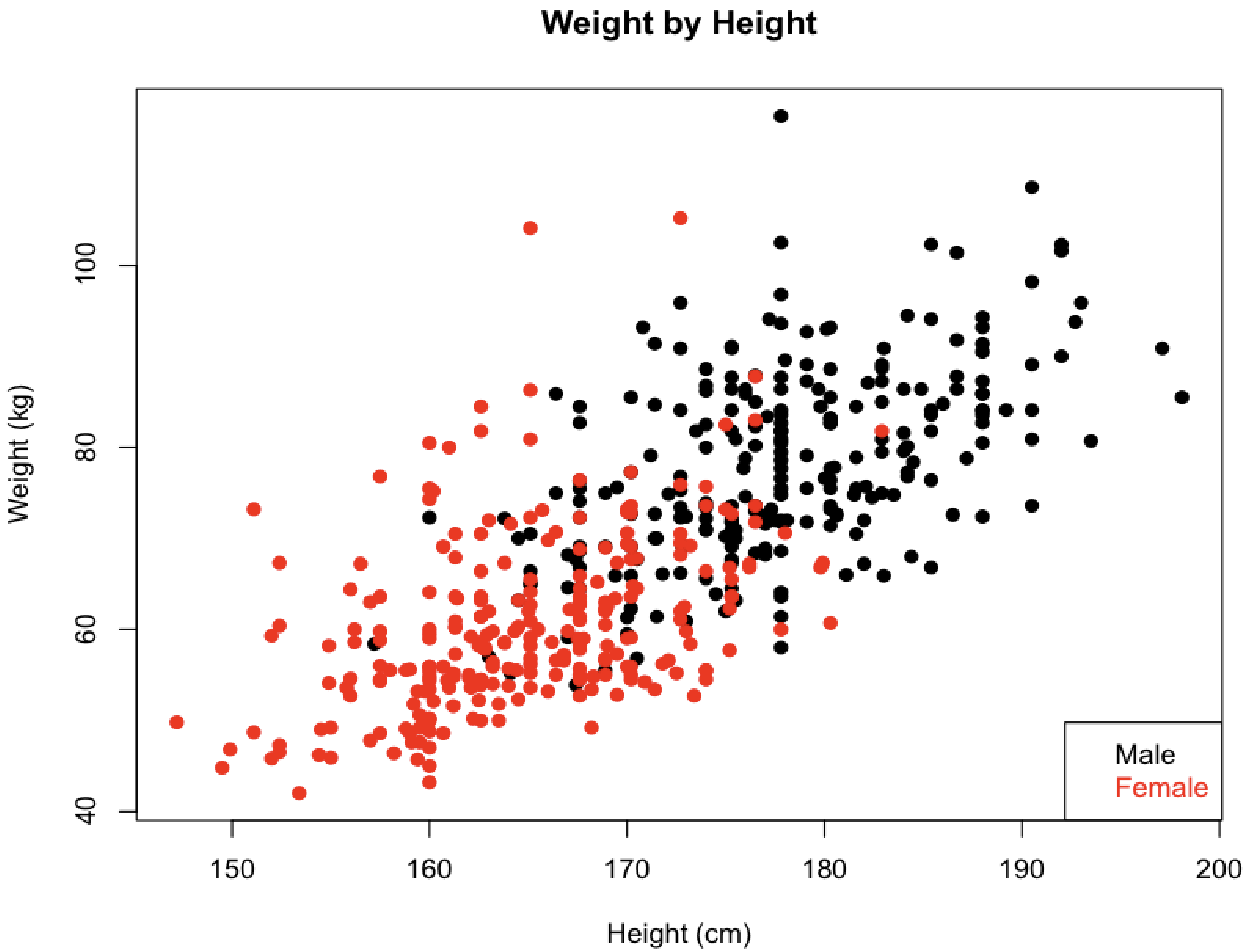
- Can be used for two-dimensional scatter (XY) plots
- Takes two vectors as input

```
> x <- c(2,4,6,8,10)
> y <- c(1.5,3,7,8,15)
> plot(y ~ x)
> plot(x, y) ## equivalent to line above
```

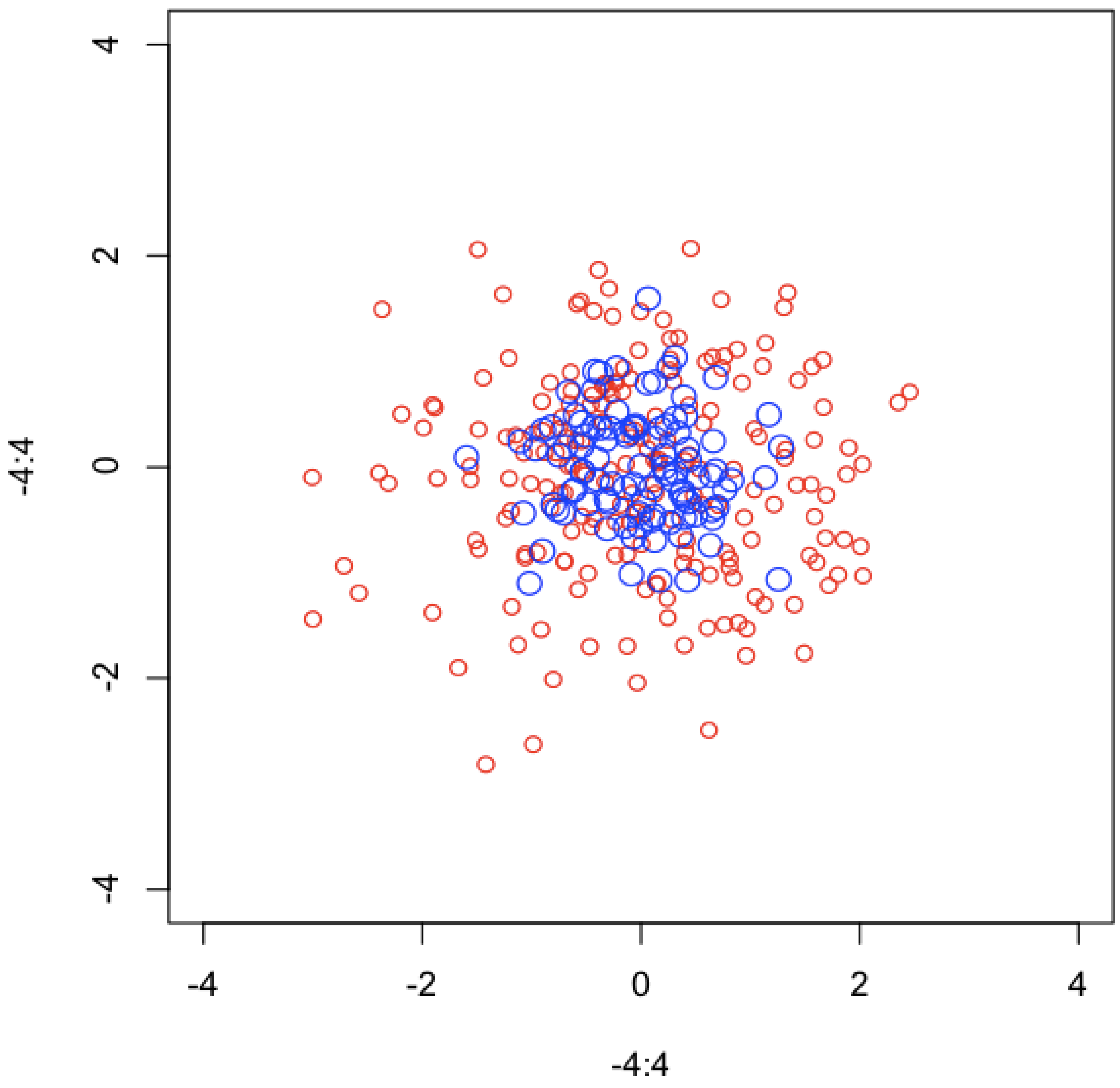
Basic formatting options

- `main`: The title
- `xlab, ylab`: X/Y axis labels
- `xlim, ylim`: X/Y axis range
- `type` : points or lines
- `pch`: point type (circle, square, filled circle etc.)
- Many more! Use `help()`

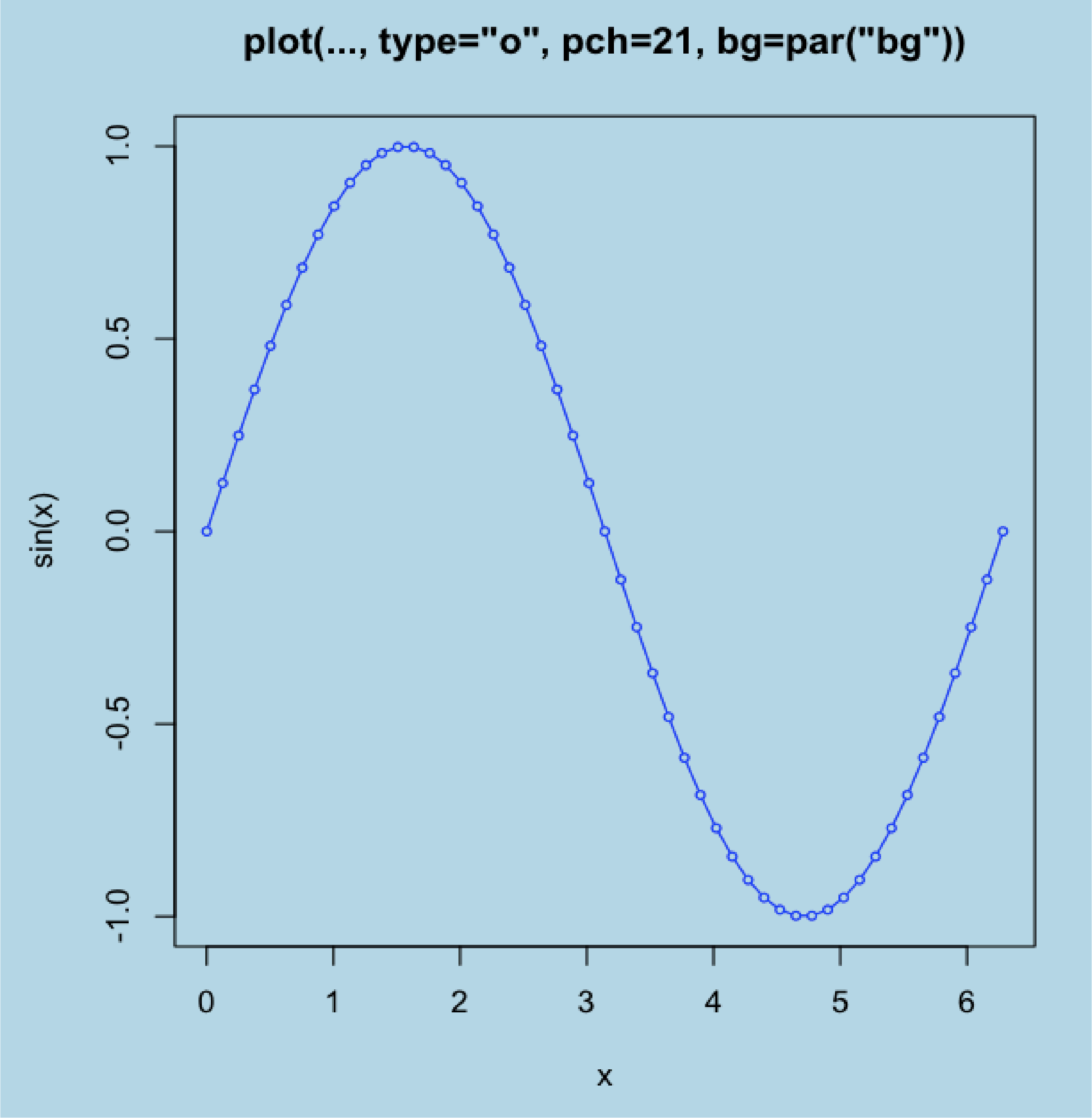
`plot()` example



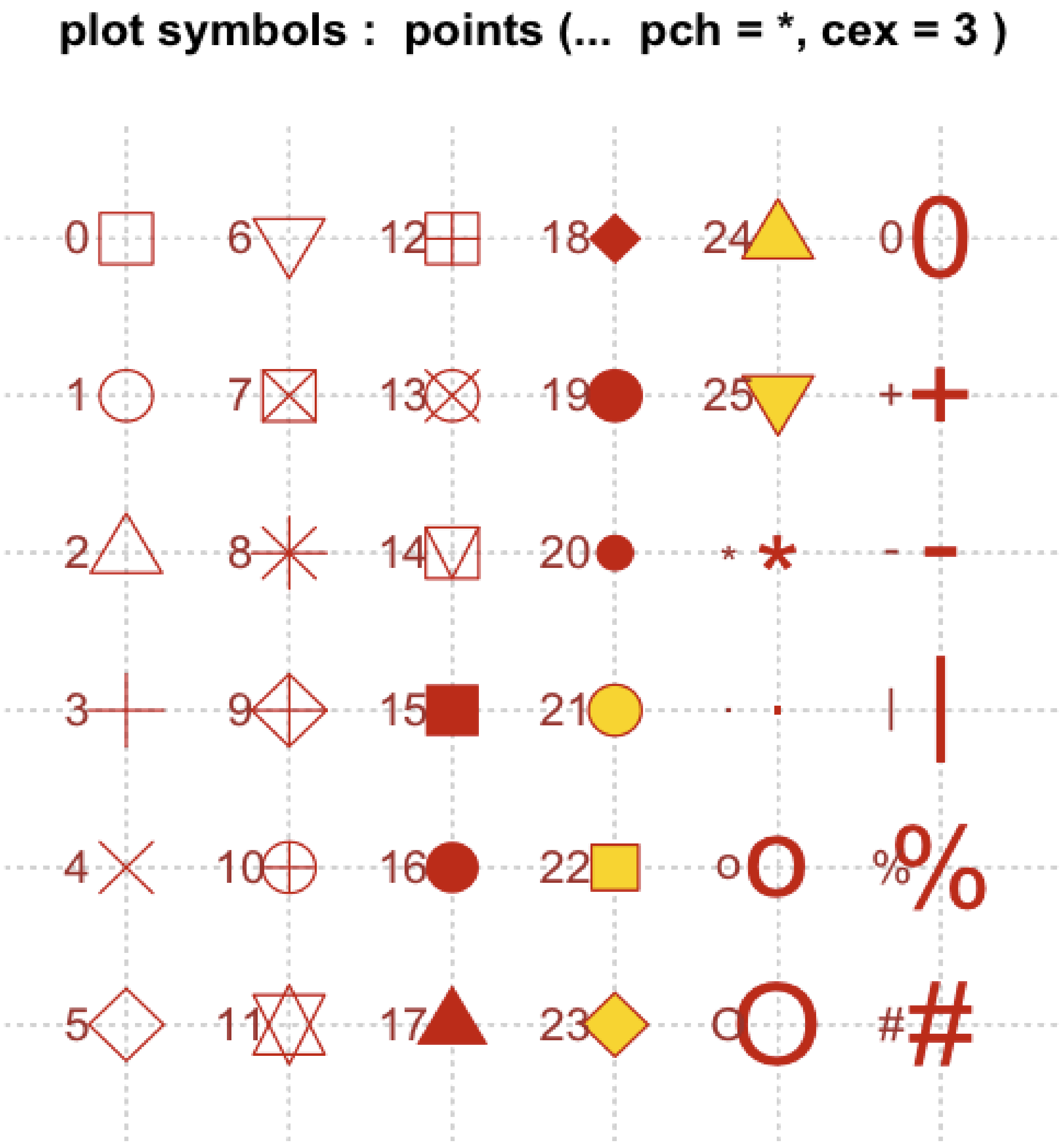
example(points)



example(points)



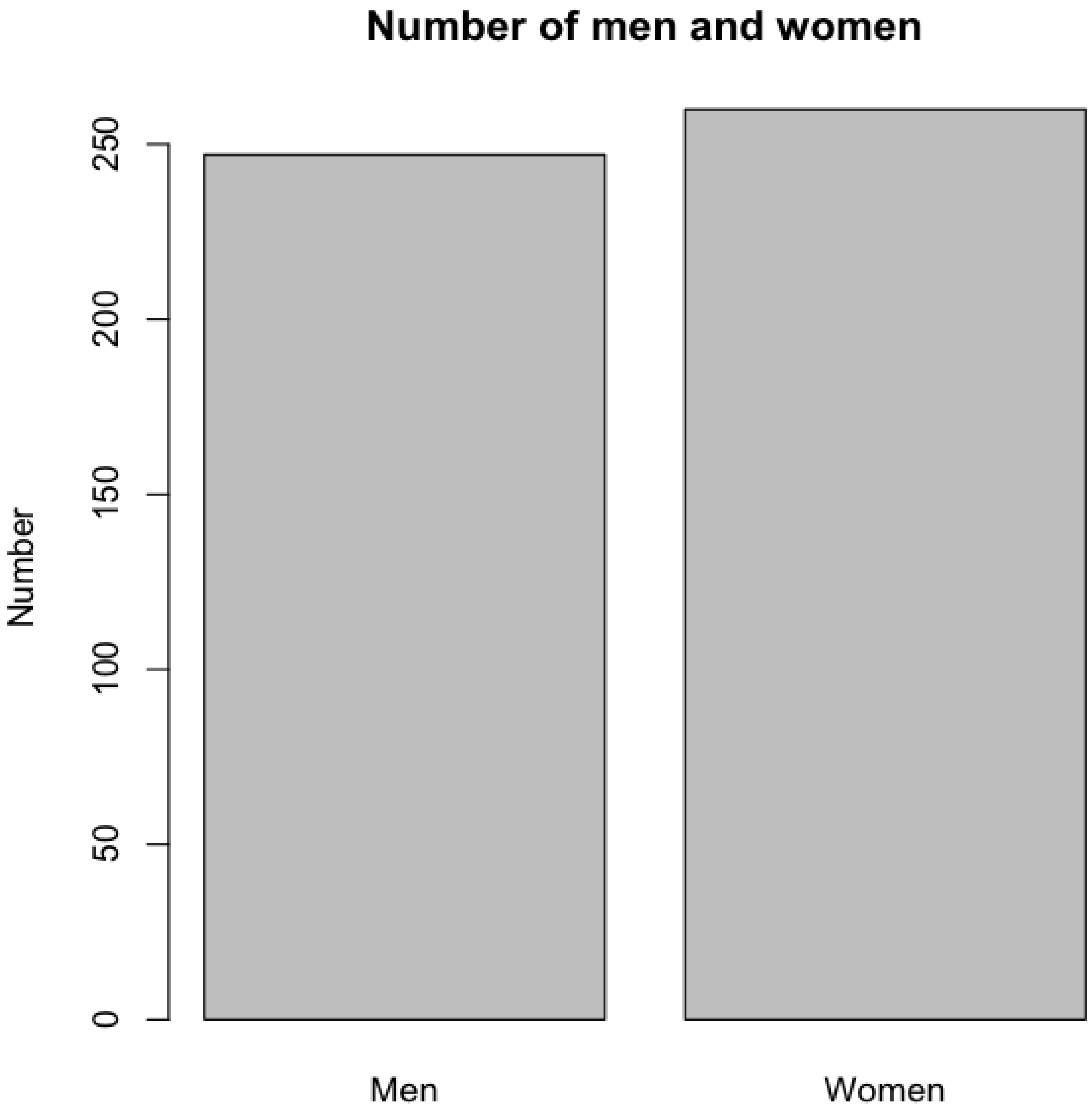
`example(points)`



`barplot()`

- Produces bar charts
- Common options include:
  - `names.arg`: the bar labels
  - `horiz`: make the bars horizontal (`TRUE/FALSE`)
  - `main, xlim, xlab etc.` function as in `plot()`

barplot()



`hist()`

- Produces histograms
- Common options include:
  - `breaks`: Specifies how to bin values
  - `main, xlim, xlab etc.` function as in `plot()`

Example: BMI

- Body Mass Index
- BMI = weight / height^2

```
> bmi = people$weight / (people$height/100)^2
```

`lines()`

- Draw straight lines
- Specify start and end coordinates
- Main options include:
  - `x`: a vector of start and end x-coordinates (e.g. `x = c(1, 4)`)
  - `y`: a vector of start and end y-coordinates

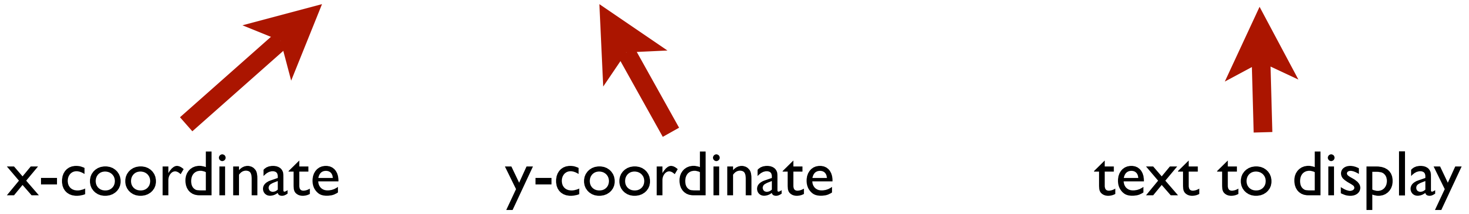
`abline()`

- A simple way to draw straight lines
- Horizontal lines:
  `abline(h=10)`
- Vertical lines
  `abline(v=5)`

`text()`

- Add text to a plot

```
> text(21, 67, "Ideal BMI range")
```



x-coordinate

y-coordinate

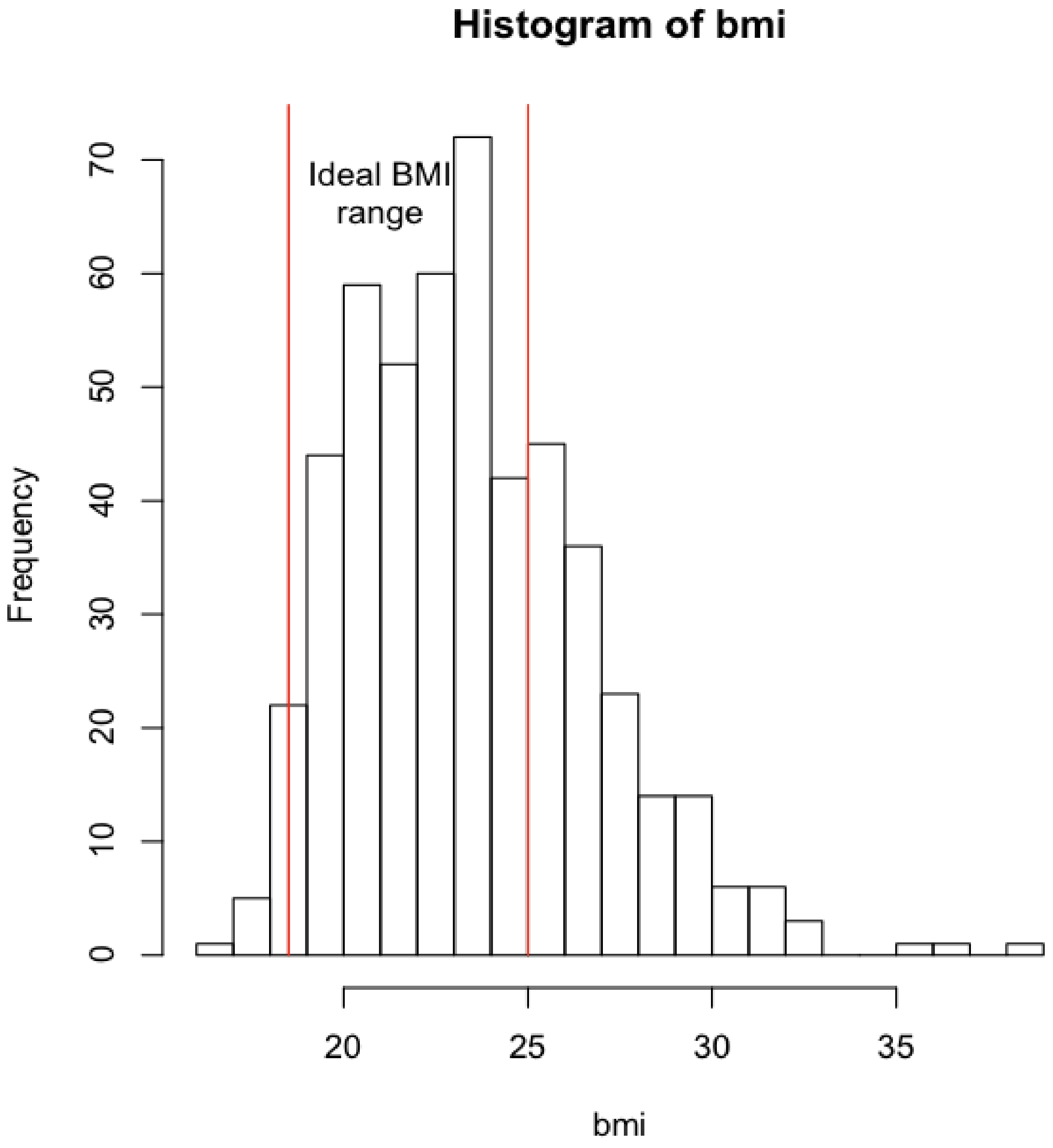
text to display

- Can use `\n` for a new line.
  e.g. `"Ideal\nBMI range"`

Example:

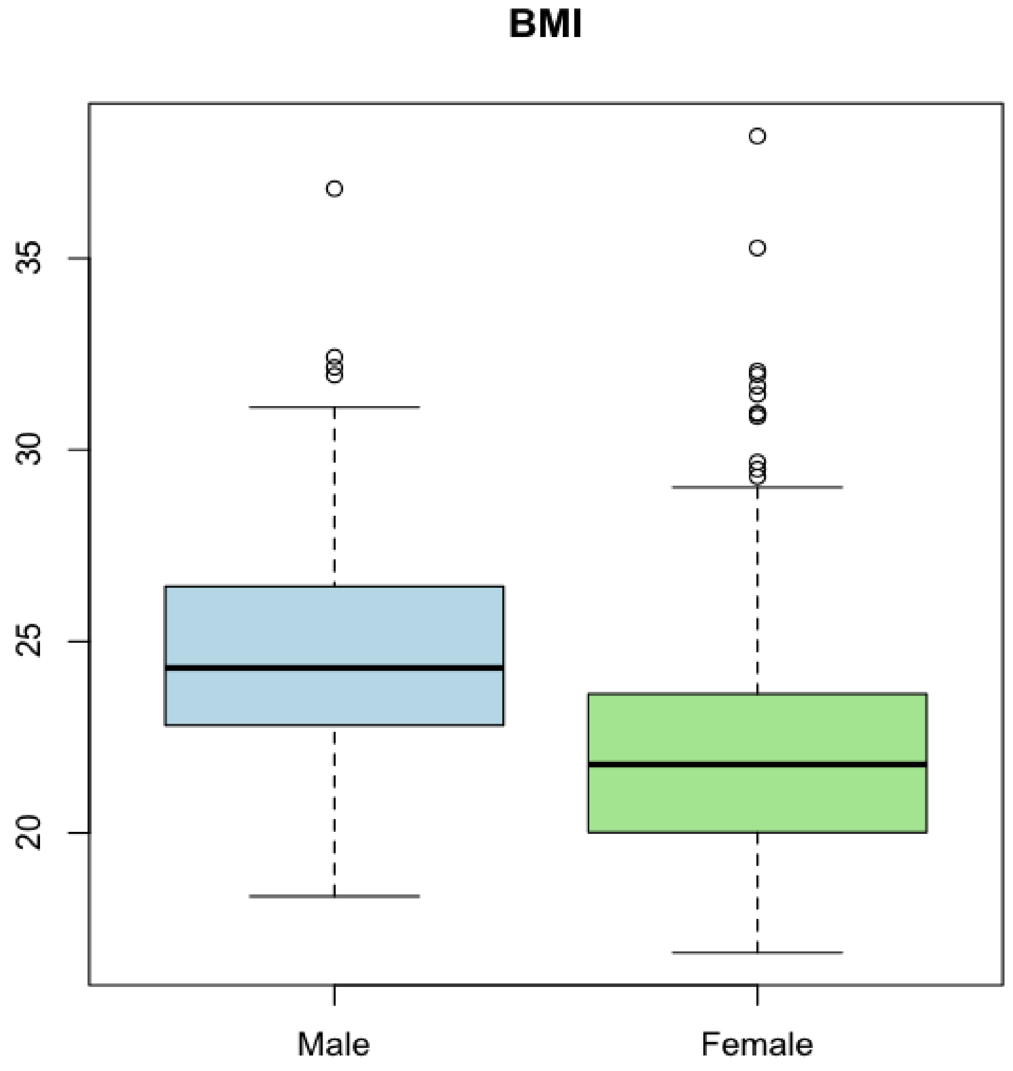
`hist(), abline() and text()`

```
> hist(bmi, breaks=20)
> abline(v=18.5, col="red")
> abline(v=25, col="red")
> text(21, 67, "Ideal BMI\nrange")
```



boxplot()

- Visual summary of:
  - Median
  - Quartiles (Q1, Q3)
  - Outliers



```
> boxplot(bmi ~ people$gender, main="BMI",
      names=c("Male", "Female"), col=c("lightblue", "lightgreen"))
```

An aside:
Subsetting a data frame

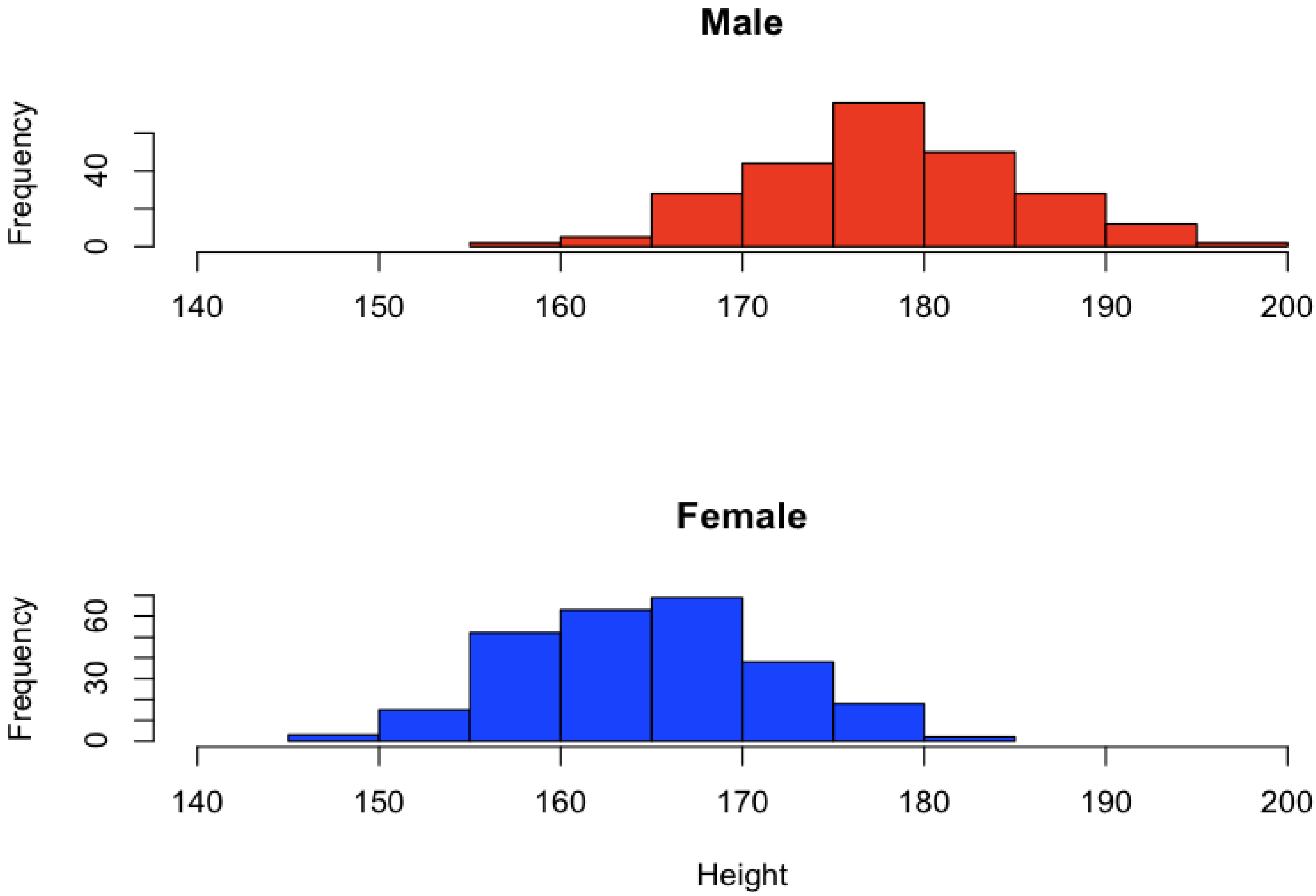
```
m = subset(people, gender==1)
```

A data frame

The inclusion criteria

```
f = subset(people, gender==2)
```

Multiple plots



Multiple plots

- Set the number of rows and columns in the plot window.

- E.g. 2 rows, 1 column:
  `par(mfrow=c(2,1))`

Example:
Had a dog?

Original data set:

```
> head(dat)
  id age sex height weight dog_0 dog_1 dog_2 dog_3 dog_4 dog_5 dog_6 dog_7 dog_8 ...
1  1  40   F   63.5  134.5    no    no   yes   yes    no    no   yes   yes    no ...
2  2  36   M   65.6  191.6    no   yes    no    no    no   yes  <NA>   yes    no ...
3  3  69   M   68.2  170.0    no   yes   yes  <NA>  <NA>   yes   yes   yes   yes ...
4  4  56   F   62.9  134.5    no    no    no    no  <NA>  <NA>   yes    no   yes ...
5  5  66   F   63.7  133.4    no  <NA>  <NA>    no   yes   yes    no    no    no ...
6  6  84   M   70.8  200.6    no   yes   yes    no   yes    no    no    no    no ...
```

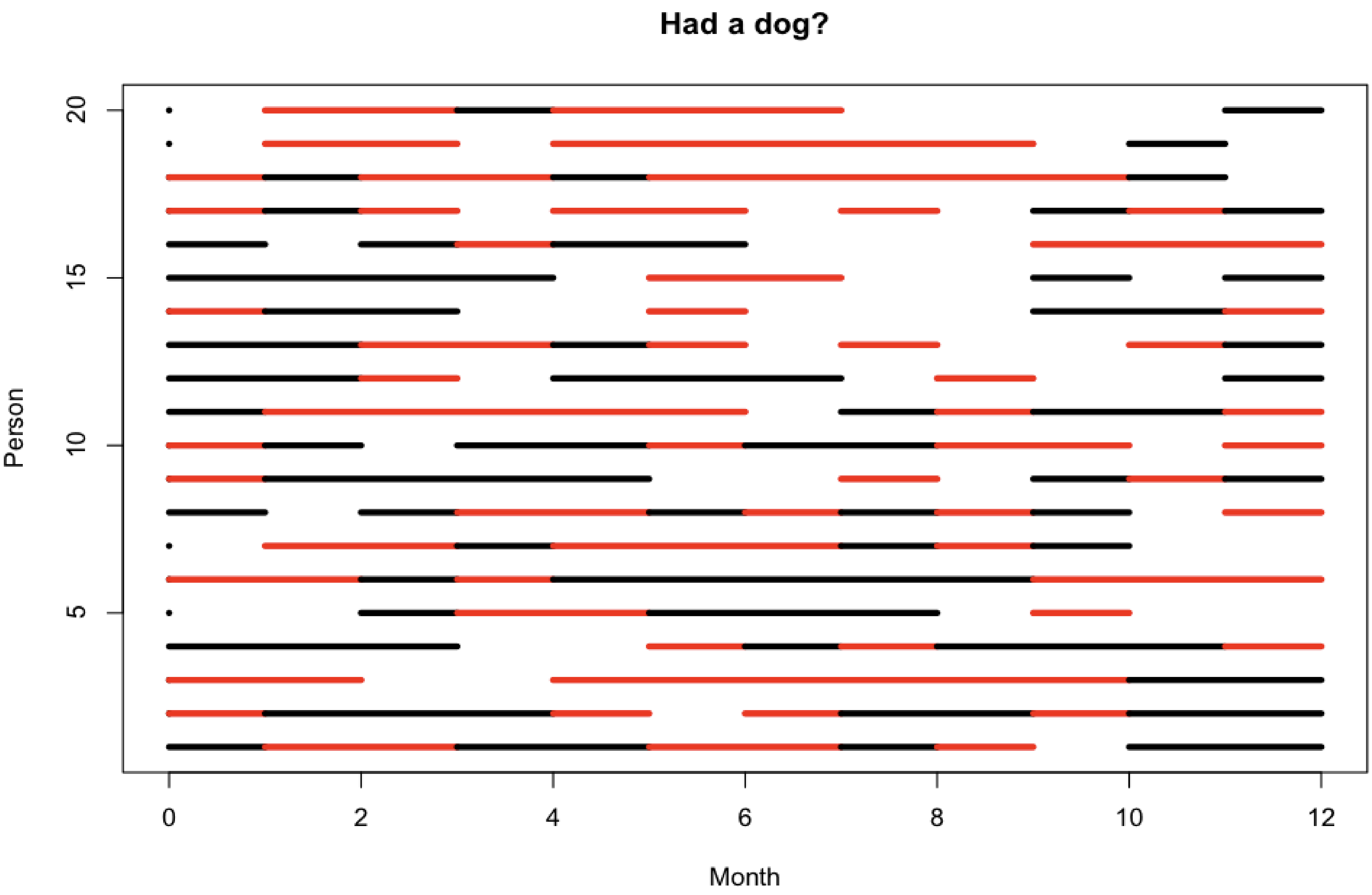
Data reformatted for plotting:

```
   > out2[1:10,]
   id exp start end
1   1   1     0   1
2   1   2     1   3
3   1   1     3   5
4   1   2     5   7
5   1   1     7   8
6   1   2     8   9
7   1   1    10  12
8   2   1     0   0
9   2   2     0   1
10  2   1     1   4
```

```
> Indexes[1:2]
$`1`
[1] 1 2 3 4 5 6 7

$`2`
[1]  8  9 10 11 12 13 14 15
```

Had a dog in the past month?
1 - No
2 - Yes


Had a dog?
Person
5
10
15
20
Month
0
2
4
6
8
10
12

Plotting the data

- Set up an empty plot window with title and labels
- Loop through the data one person at a time to construct their plot.

Plotting the data:

The empty plot window:

```
plot(0 ~ 0, type = "n",
  ylim = c(1,20), # first 20 people
  xlim = c(0,12), # 12 months
  main = "Had a dog?",
  xlab = "Month",
  ylab = "Person")
```

Plotting the data:

Loop through the data, processing one person at a time:

```
for(i in 1:20) { # first 20
    Index = Indexes[[i]]
    tmp = out2[Index,]
    for(j in 1:nrow(tmp)) {
        lines(x = as.numeric(tmp[j,c("start", "end")]),
            y = c(i,i), col = tmp$exp[j], lwd = 4)
    }
}
```

```
   > out2[1:10,]
   id exp start end
1   1   1     0   1
2   1   2     1   3
3   1   1     3   5
4   1   2     5   7
5   1   1     7   8
6   1   2     8   9
7   1   1    10  12
8   2   1     0   0
9   2   2     0   1
10  2   1     1   4
```

```
> Indexes[1:2]
$`1`
[1] 1 2 3 4 5 6 7

$`2`
[1]  8  9 10 11 12 13 14 15
```

Summary

- Many, many plot types and options
- Use `help(), example()` and the google.