Introduction to R

Session 5 – Basic Graphics

Statistical Consulting Centre

consulting@stat.auckland.ac.nz
The Department of Statistics
The University of Auckland

19 July, 2017

SCIENCE
DEPARTMENT OF STATISTICS

Exploring your data

- Examine for patterns, relationships, structures, and other features.
- Do this using graphs, tables and summary statistics.
- Patterns are more easily seen in graphs.
- Remember:

  "R is a free software environment for statistical computing and *graphics*"

Designing graphs

Two important considerations when designing a graph:

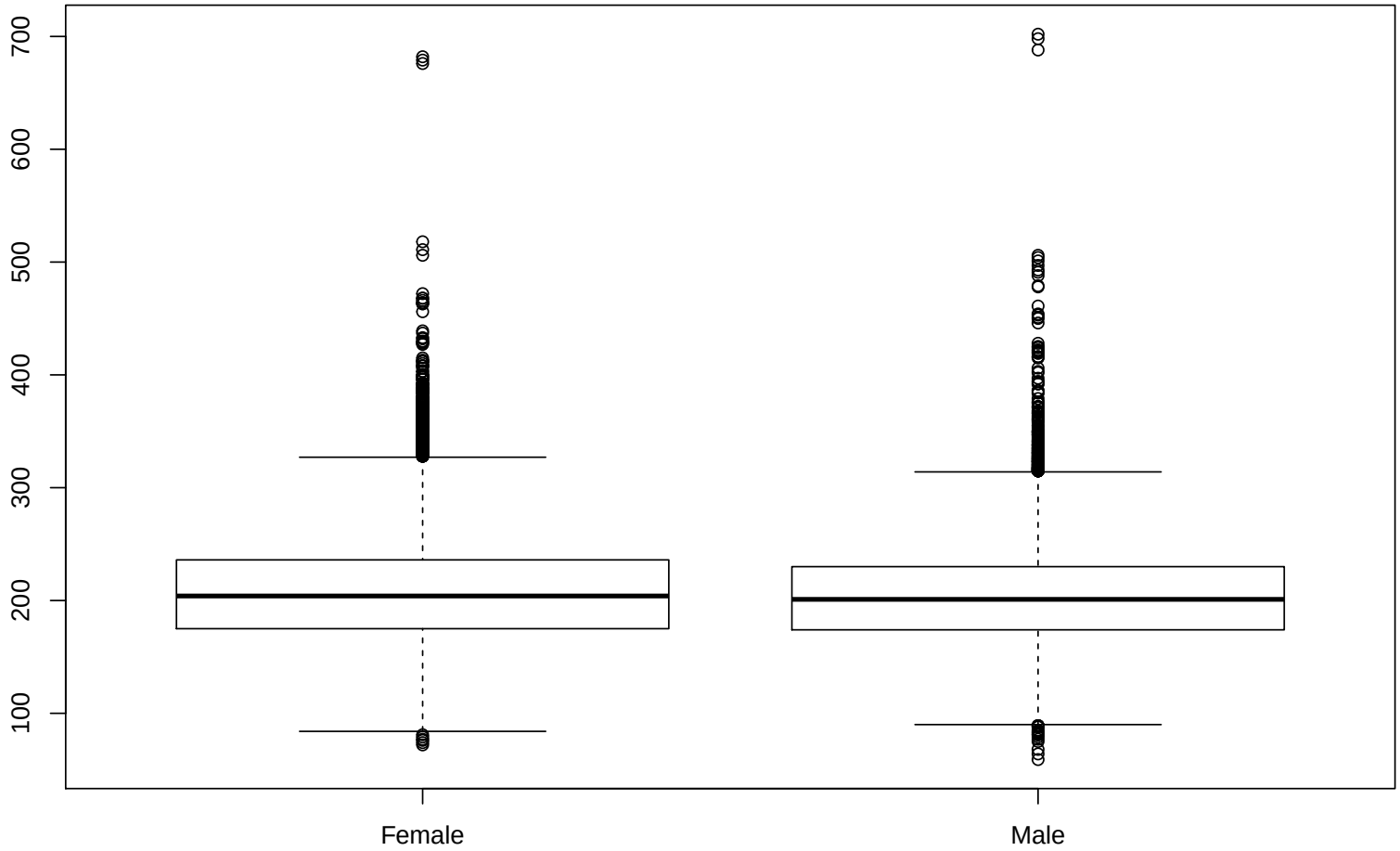
- It should have something to say.
  - Decide what information you want your graph to display.
  - Graphical software is capable of producing any number of different graphs, so generating graphs with meaningless information is quite simple to do!
- It should be easy to interpret.
  - Simplicity is key!
  - Sufficiently large fonts, all axes labelled, clearly defined legends, etc.

Some functions for drawing graphs

- Usually, we use `plot()` to create a graph and then `lines()`, `points()`, `text()`, and other commands to annotate it.
- `plot()` does appropriate things for different types of variables (see later examples).
- Variables of type `character` are not suitable for plotting (and analysis). Convert such variables to type `factor`.
- There are no "erase" or "undo" functions, so store your commands in an `R` script.

Boxplots

```
with(combined.long.df, plot(Sex, Cholesterol))
```



Boxplots

```
with(combined.long.df, plot(as.character(Sex), Cholesterol))
```

```
## Warning in xy.coords(x, y, xlabel, ylabel, log):
## NAs introduced by coercion

## Warning in min(x): no non-missing arguments to
## min; returning Inf

## Warning in max(x): no non-missing arguments to
## max; returning -Inf

## Error in plot.window(...): need finite 'xlim' values
```

Boxplots

Sex varaible in `combined.long.df` has to be `factor`.

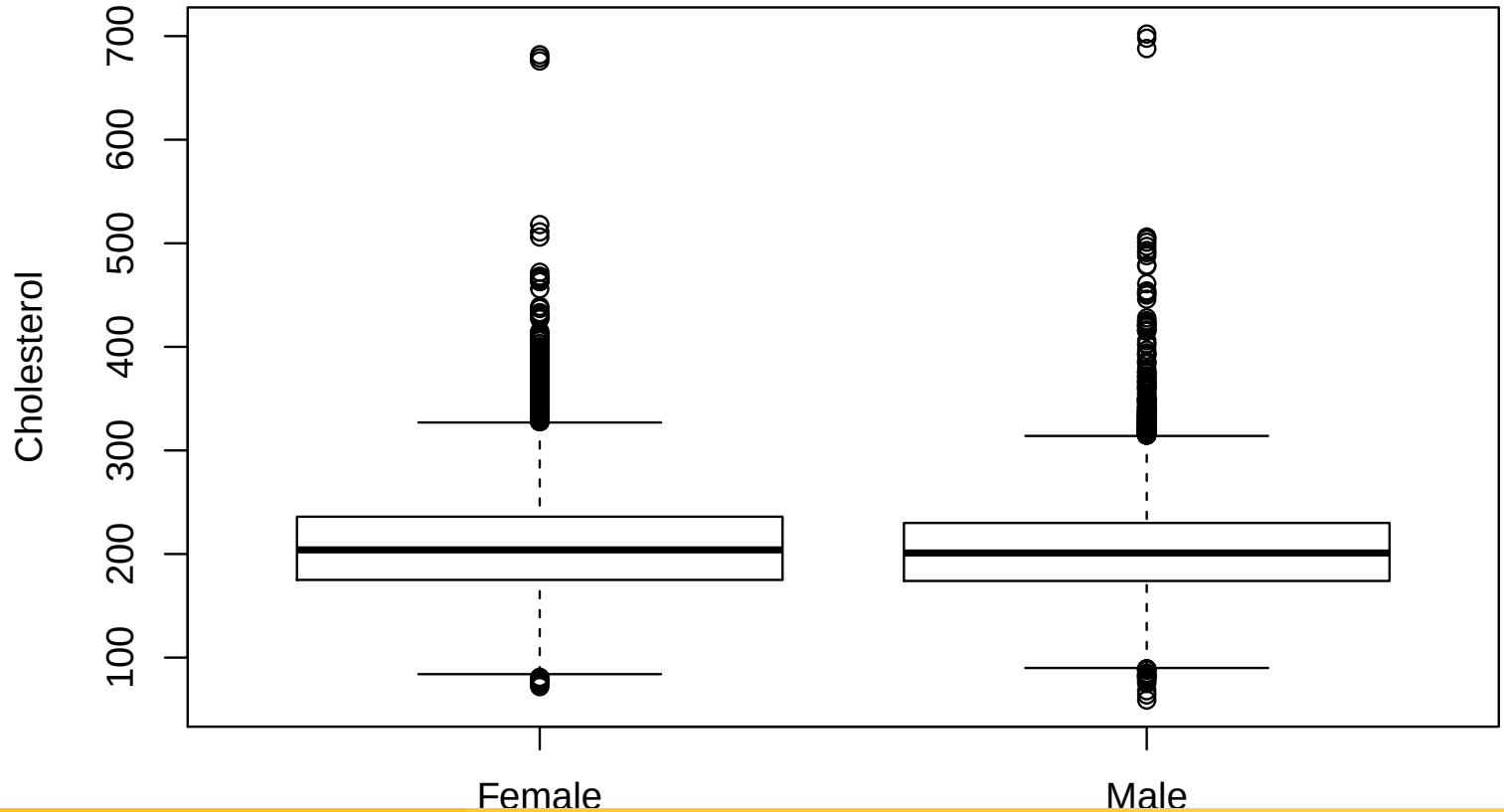
```
class(combined.long.df$Sex)
```

```
## [1] "factor"
```

Formula interface

The plot() command can also be used this way:

```
with(combined.long.df, plot(Cholesterol ~ Sex))
```

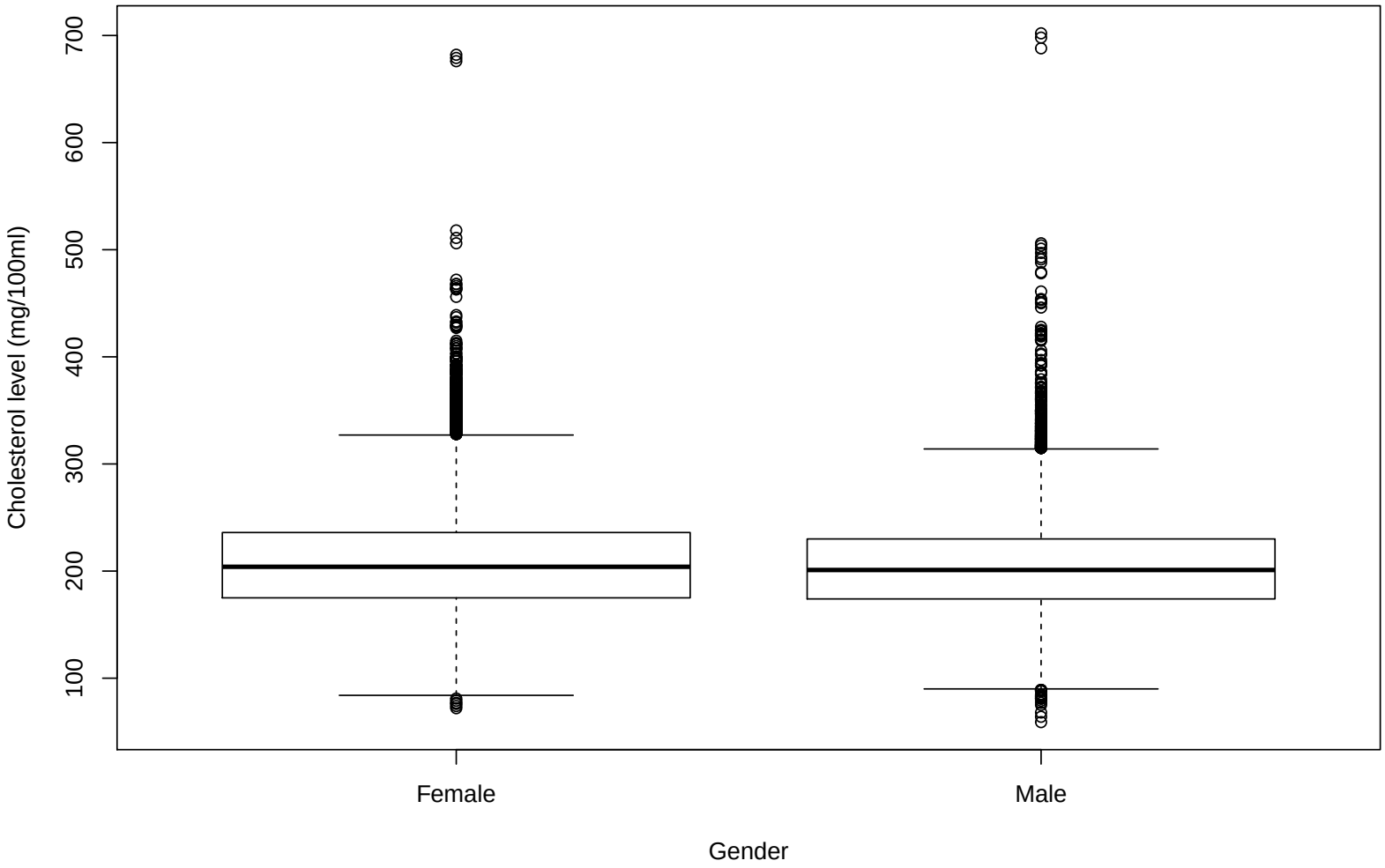


Axes need labels

Label axes with units.

```
with(combined.long.df,
     plot(Sex, Cholesterol, xlab = "Gender",
          ylab = "Cholesterol level (mg/100ml)"))
```

xlab and ylab

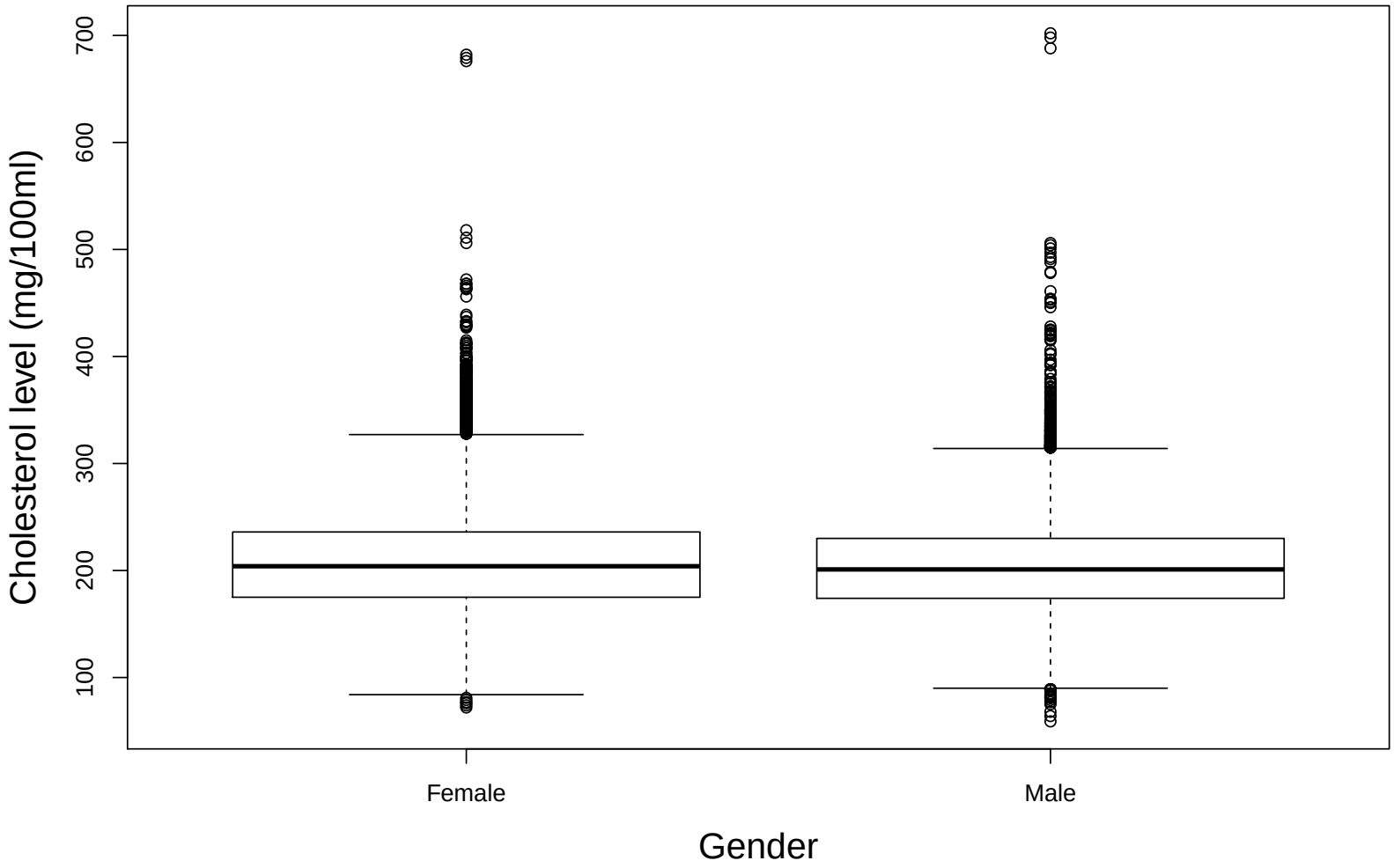


Axes labels are too small

Label axes with units, large enough to read.

```
with(combined.long.df,
     plot(Sex, Cholesterol, xlab = "Gender",
          ylab = "Cholesterol level (mg/100ml)",
          cex.lab = 1.5))
```

cex.lab = 1.5

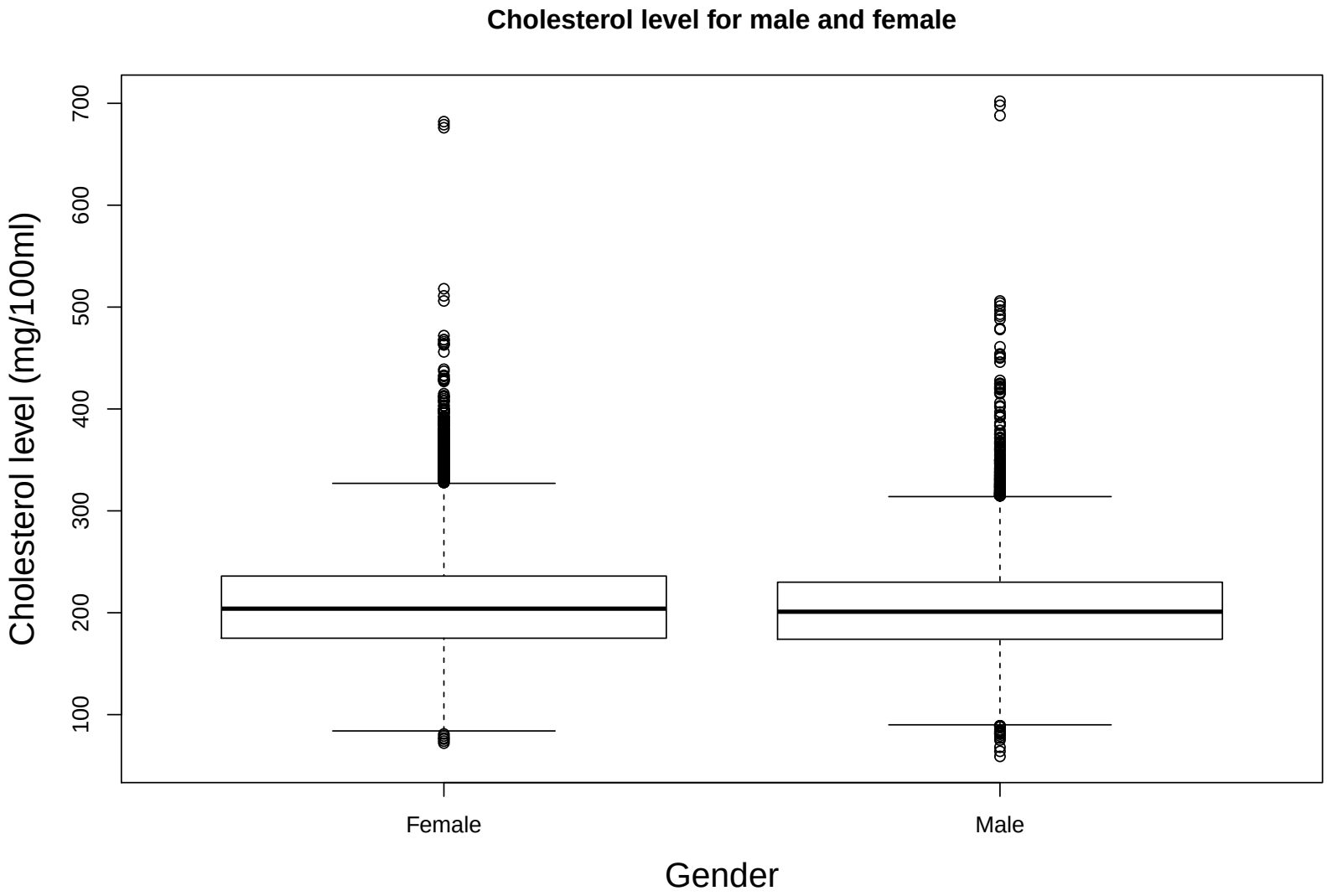

Cholesterol level (mg/100ml)
100
200
300
400
500
600
700
Female
Male
Gender

Title

Sometimes you need a title.

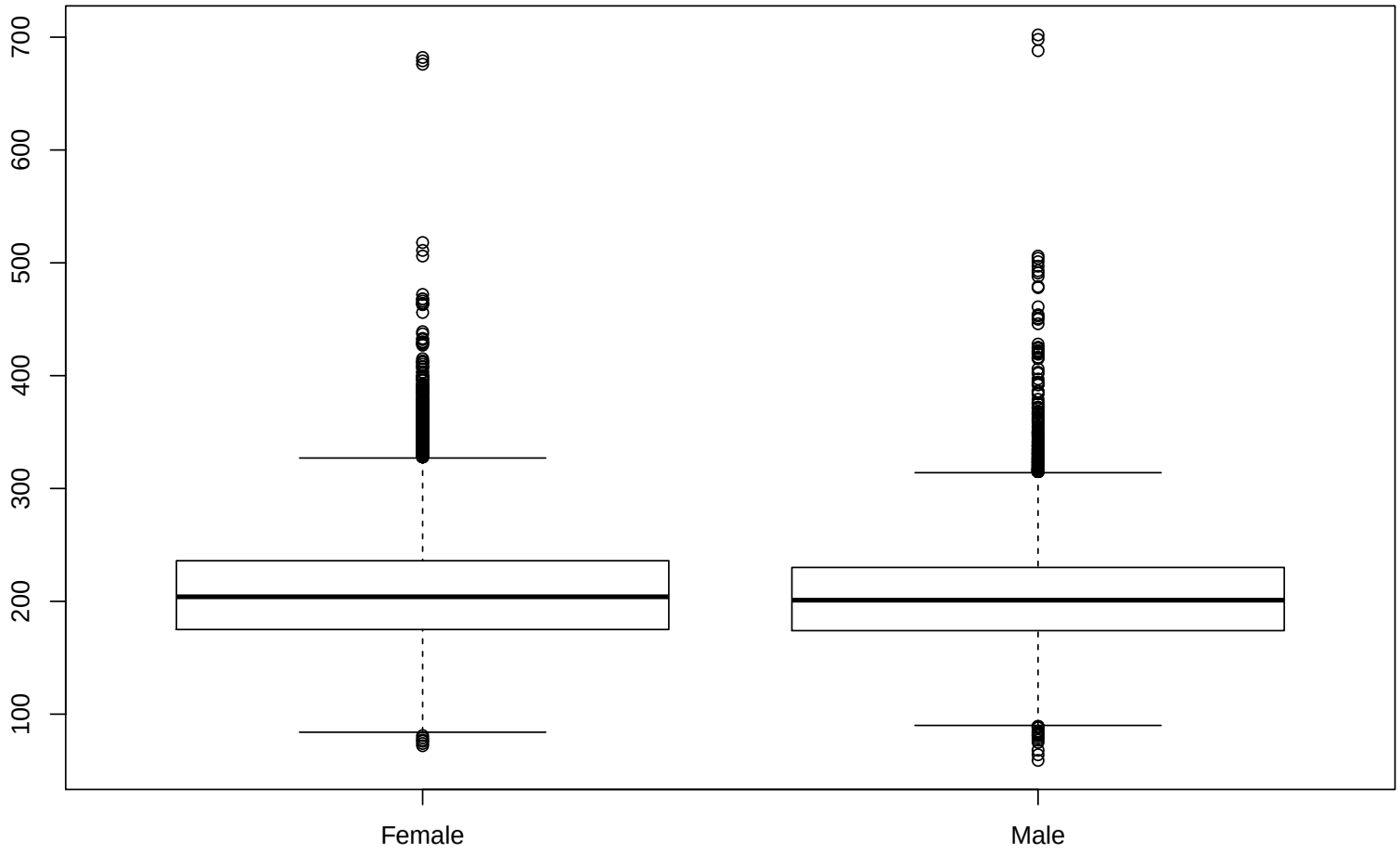
```
with(combined.long.df,
     plot(Sex, Cholesterol, xlab = "Gender",
          ylab = "Cholesterol level (mg/100ml)",
          cex.lab = 1.5,
          main = "Cholesterol level for male and female"))
```

main


Cholesterol level for male and female
Cholesterol level (mg/100ml)
100
200
300
400
500
600
700
Female
Male
Gender

boxplot()

`boxplot()` is probably more fexible than `plot()`, if you are drawing boxplots. Check the help document for more details.



Smoking group by Age group

Examine the data via twp=way freqency table.

```
with(combined.df, table(Smoke.group, Age.group))
```

```
##              Age.group
## Smoke.group Under 35 36 to 60 Over 61
##          No       643     1548    2064
##          Yes     1732     1840     799
```

Smoking group by Age group

Then, convert the counts to percentages.

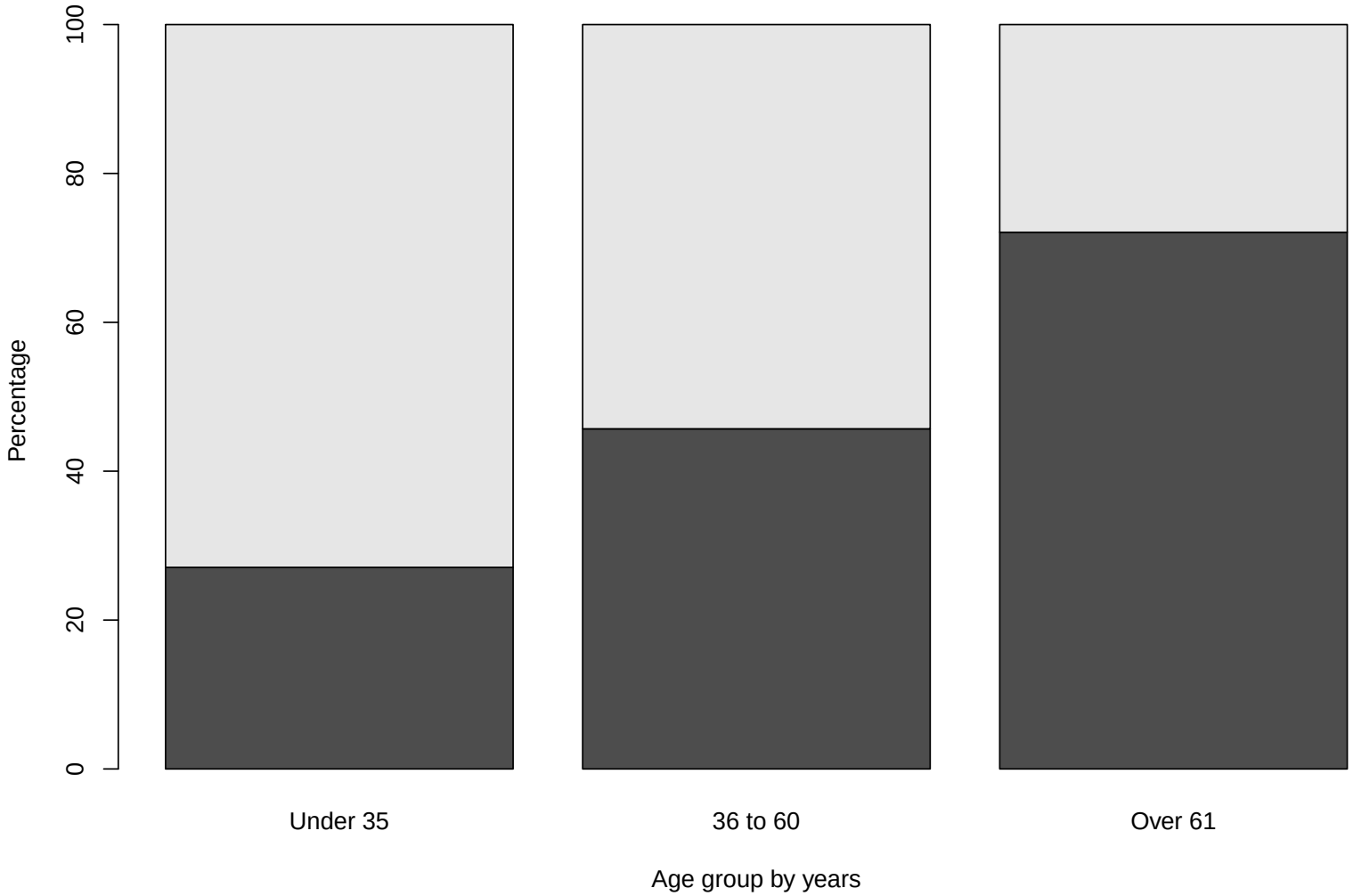
```
smoke.age <- with(combined.df, 100*
                      prop.table(table(Smoke.group,
                                       Age.group), 2))

round(smoke.age, 2)
```

```
##              Age.group
## Smoke.group Under 35 36 to 60 Over 61
##          No     27.07    45.69   72.09
##          Yes    72.93    54.31   27.91
```

barplot

```
barplot(smoke.age, xlab = "Age group by years",
        ylab = "Percentage")
```

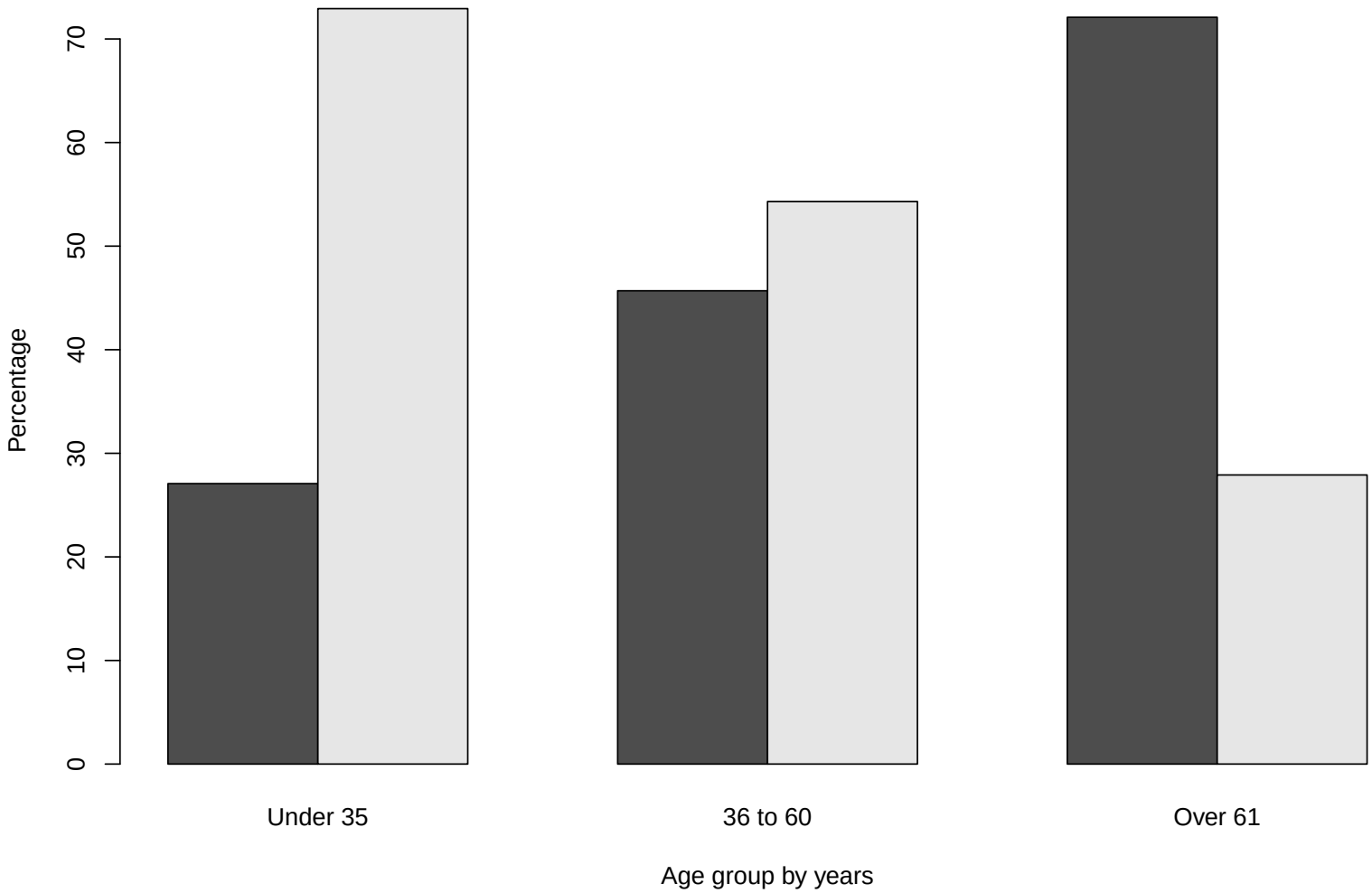
barplot


Percentage
0
20
40
60
80
100
Under 35
36 to 60
Over 61
Age group by years

Side-by-side barplot

```
barplot(smoke.age, beside = TRUE,
        xlab = "Age group by years",
        ylab = "Percentage")
```

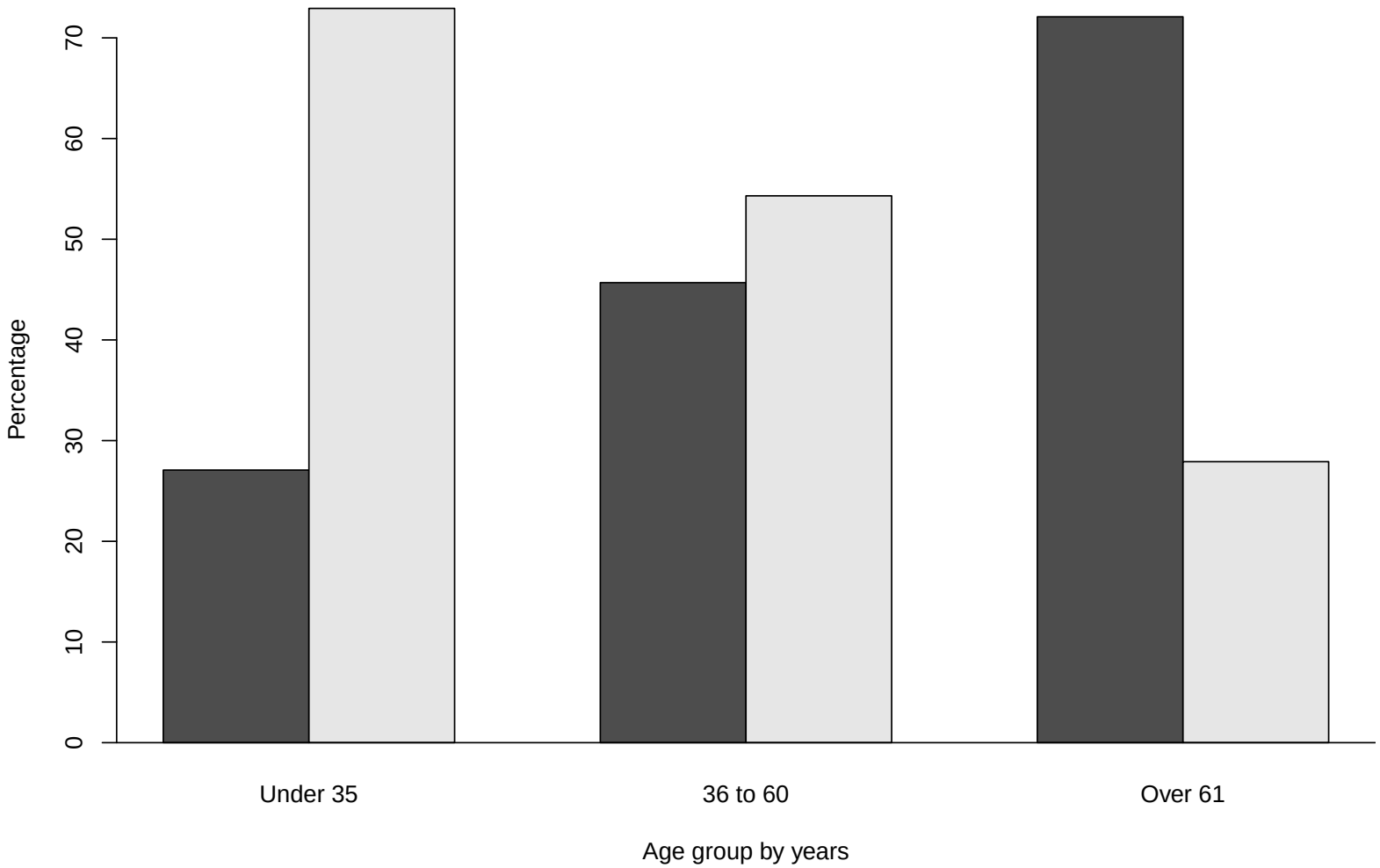
beside = TRUE


Percentage
0
10
20
30
40
50
60
70
Under 35
36 to 60
Over 61
Age group by years

Line along horizontal axis?

```
barplot(smoke.age, beside = TRUE,
        xlab = "Age group by years",
        ylab = "Percentage")
abline(h = 0)
```

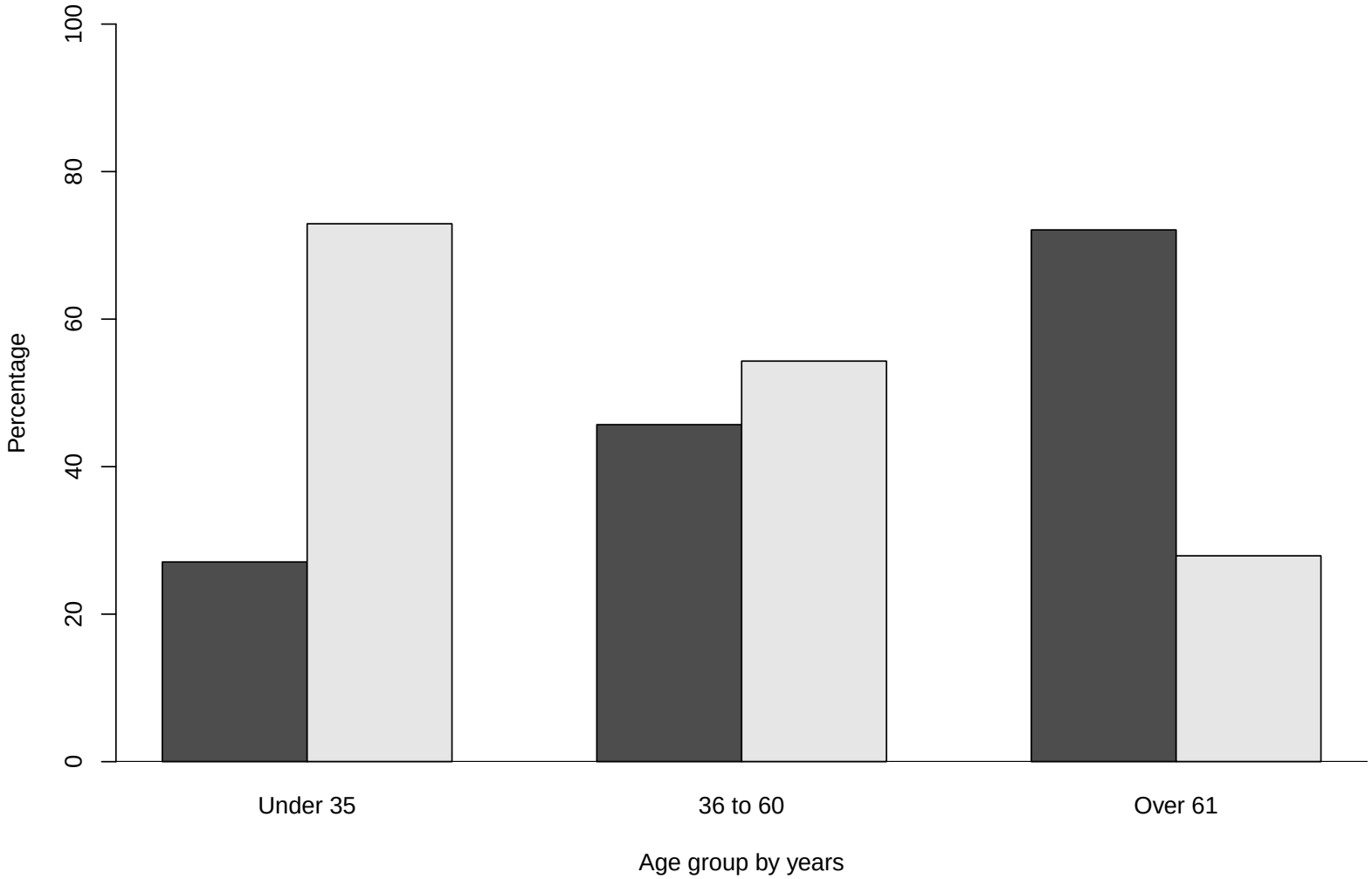
abline(h = 0)


Percentage
0
10
20
30
40
50
60
70
Under 35
36 to 60
Over 61
Age group by years

Vertical axis higher than bars?

```
barplot(smoke.age, beside = TRUE,
        xlab = "Age group by years",
        ylab = "Percentage",
        ylim = c(0, 100))
abline(h = 0)
```

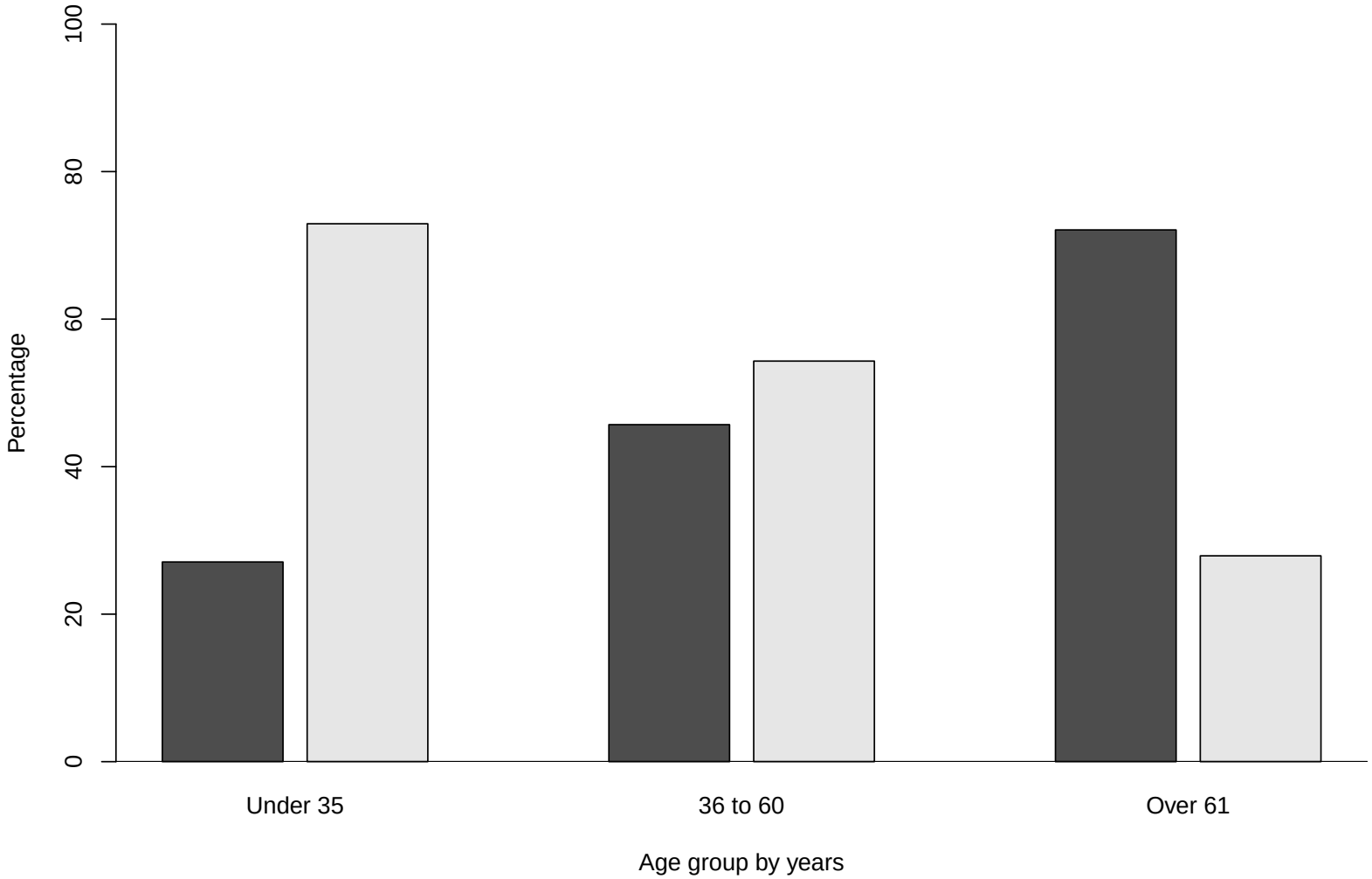
ylim = c(0, 100)


Percentage
0
20
40
60
80
100
Under 35
36 to 60
Over 61
Age group by years

Adding space between bars

```
barplot(smoke.age, beside = TRUE,
        xlab = "Age group by years",
        ylab = "Percentage",
        ylim = c(0, 100),
        space = c(0.2, 1.5))
abline(h = 0)
```

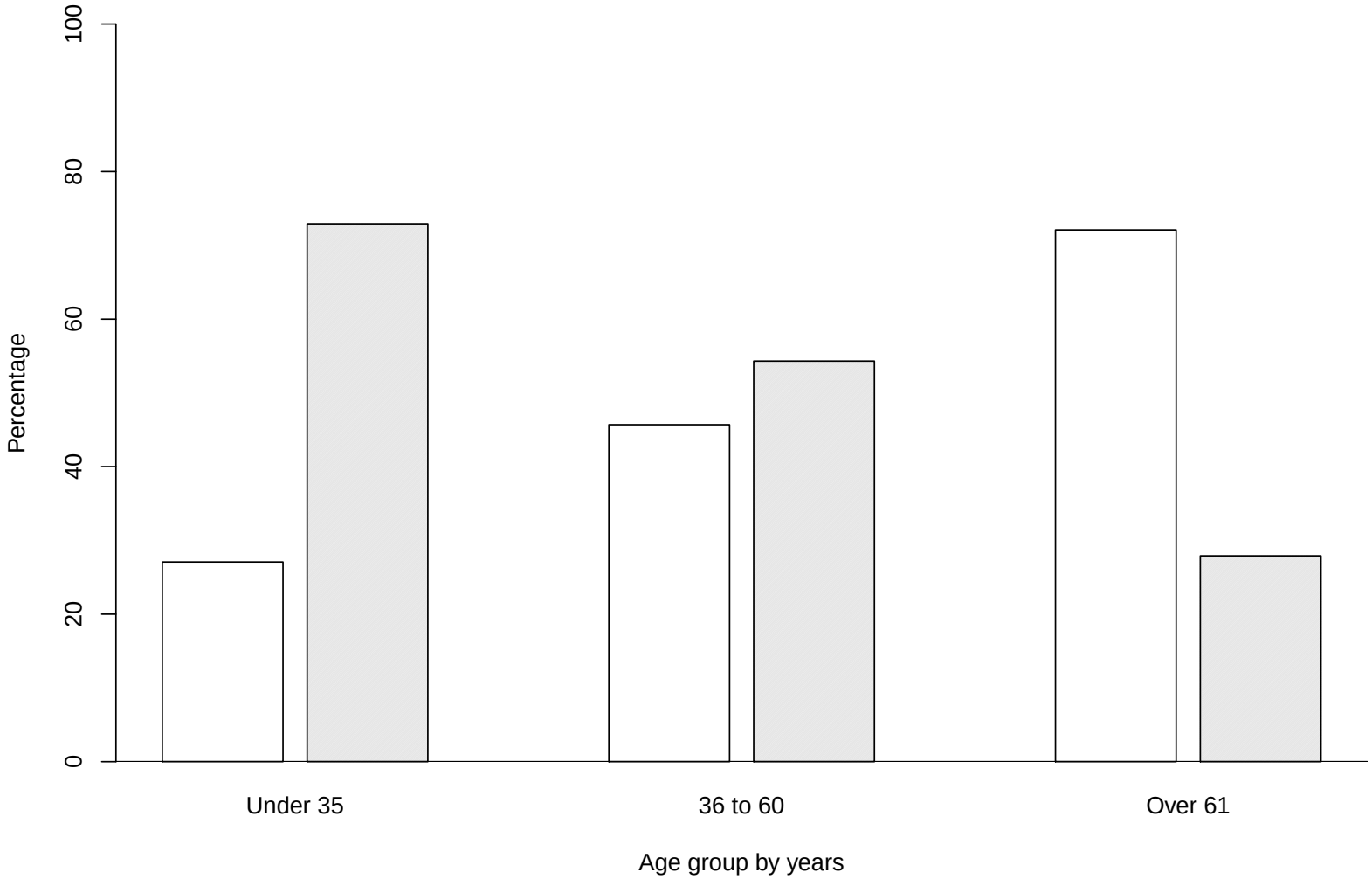
space = c(0.2, 1.5)


Percentage
0
20
40
60
80
100
Under 35
36 to 60
Over 61
Age group by years

Shading with lines

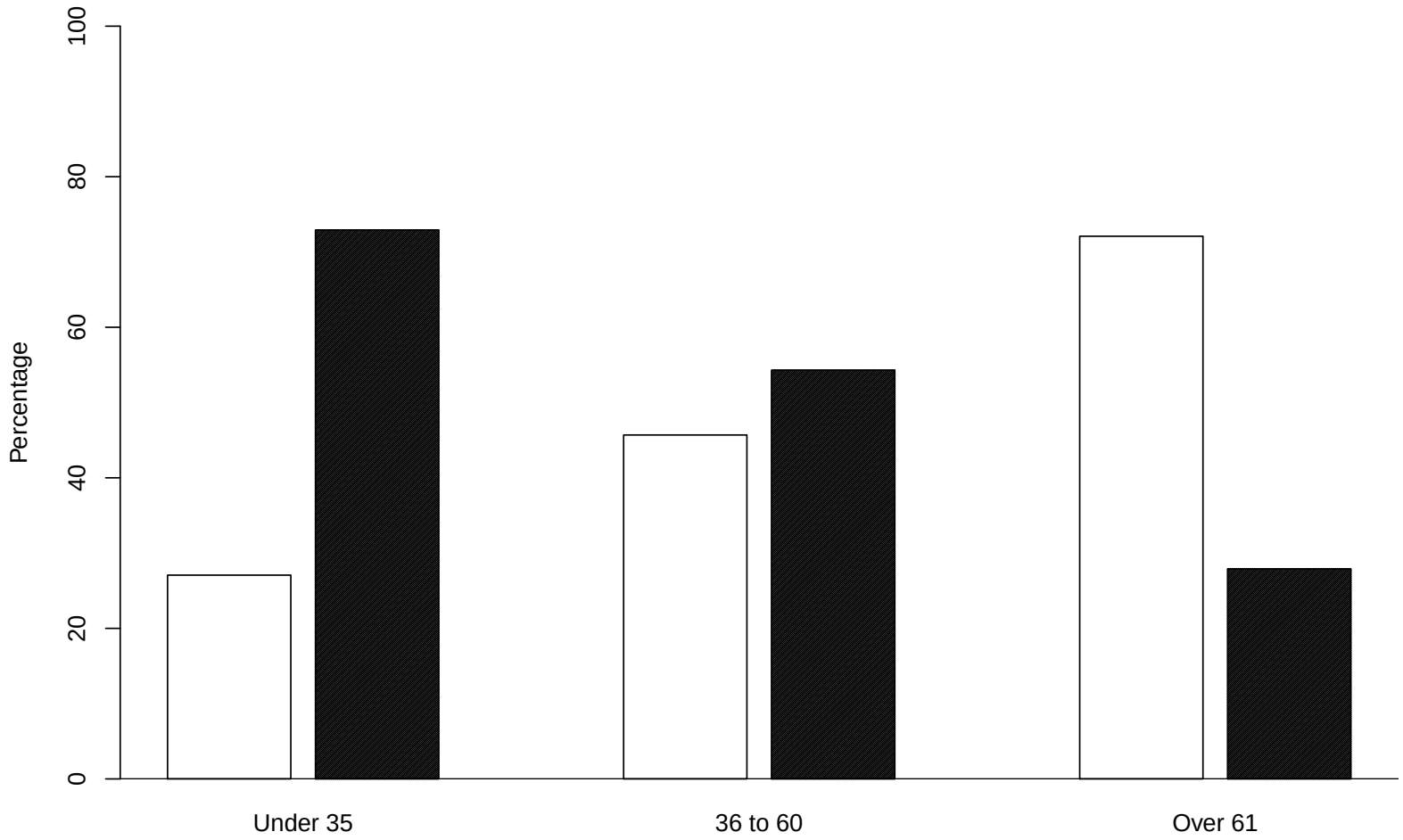
```
barplot(smoke.age, beside = TRUE,
        xlab = "Age group by years",
        ylab = "Percentage",
        ylim = c(0, 100),
        space = c(0.2, 1.5),
        density = c(0, 100))
abline(h = 0)
```

density = c(0, 100)

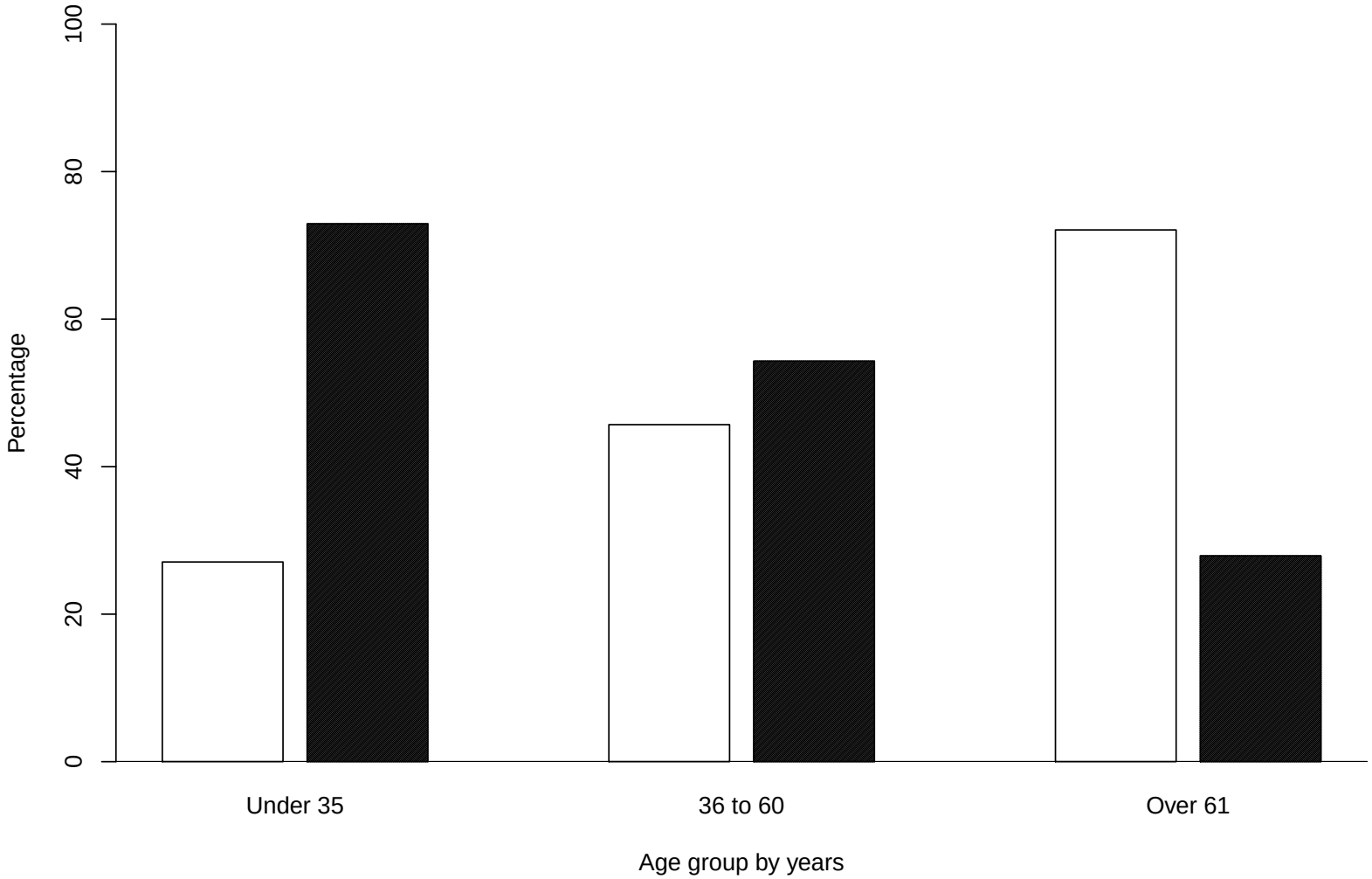


Shading with lines

The defualt colour for shading lines is pale while! Change to black using the `col` argument.



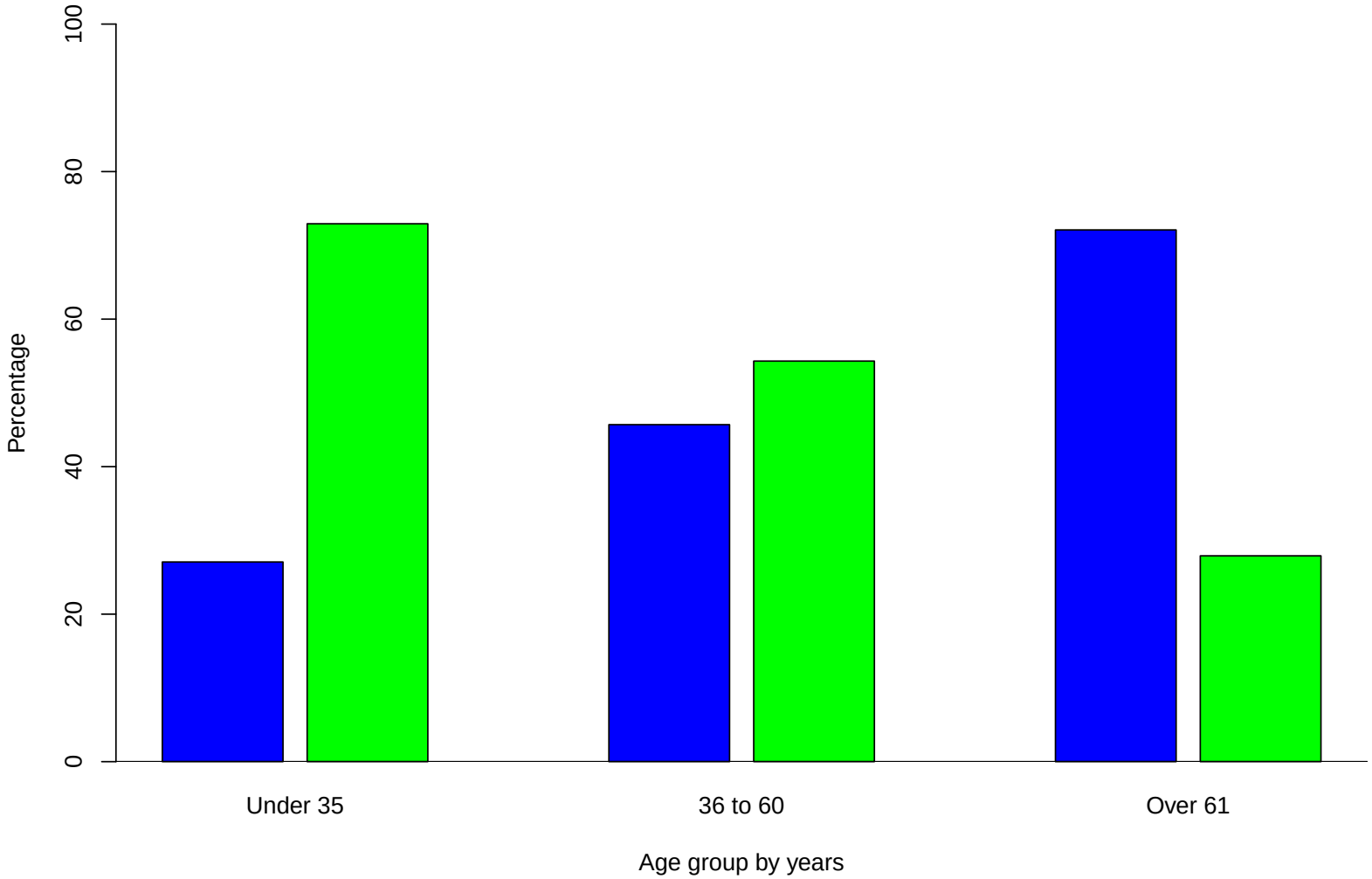
col = "black"


Percentage
0
20
40
60
80
100
Under 35
36 to 60
Over 61
Age group by years

Color-filled bars?

```
barplot(smoke.age, beside = TRUE,
        xlab = "Age group by years",
        ylab = "Percentage",
        ylim = c(0, 100),
        space = c(0.2, 1.5),
        col = c("blue", "green"))
abline(h = 0)
```

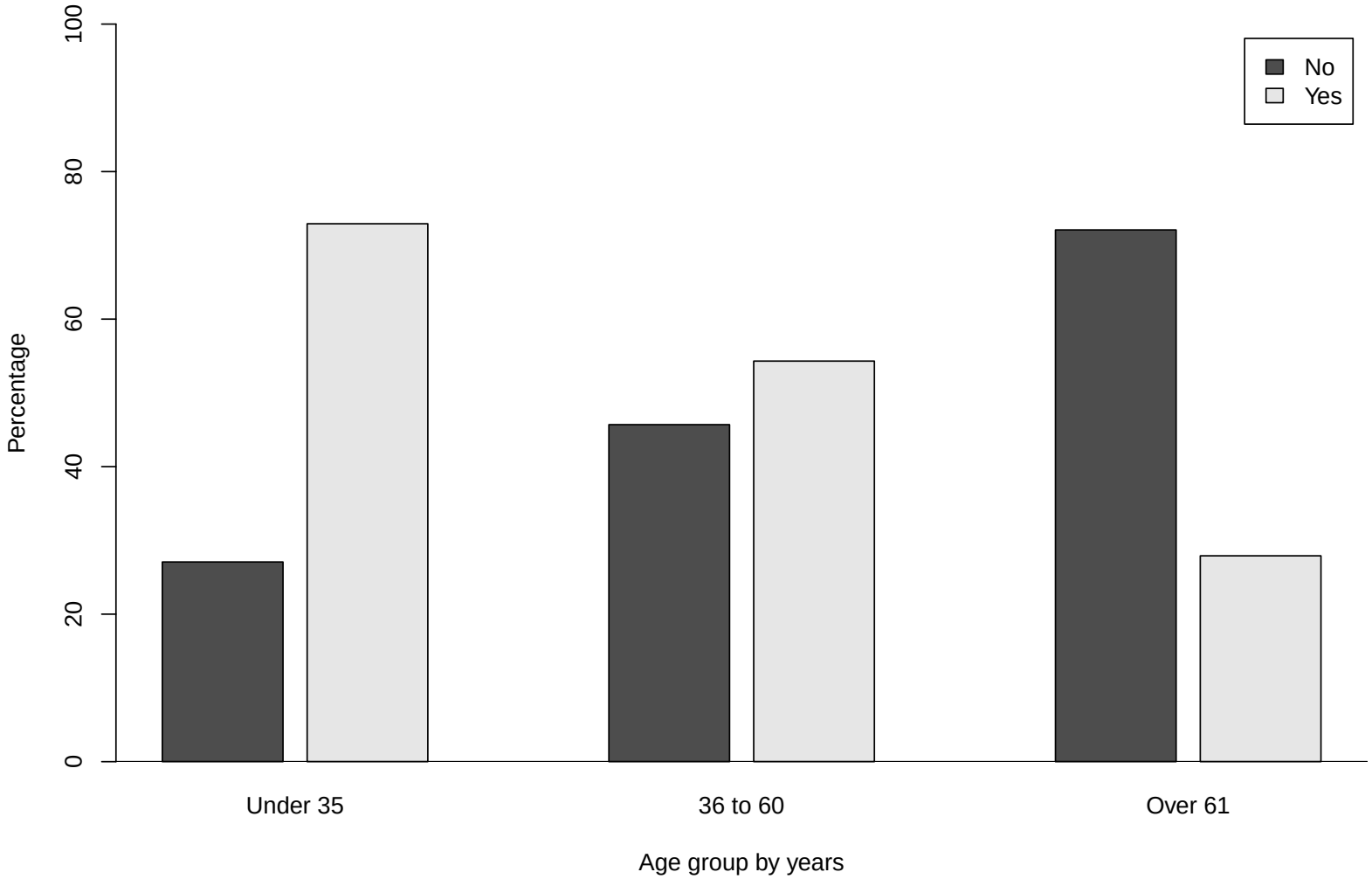
col = c("blue", "green")



Legend?

```
barplot(smoke.age, beside = TRUE,
        xlab = "Age group by years",
        ylab = "Percentage",
        ylim = c(0, 100),
        space = c(0.2, 1.5),
        legend.text = TRUE)
abline(h = 0)
```

legend.text = TRUE

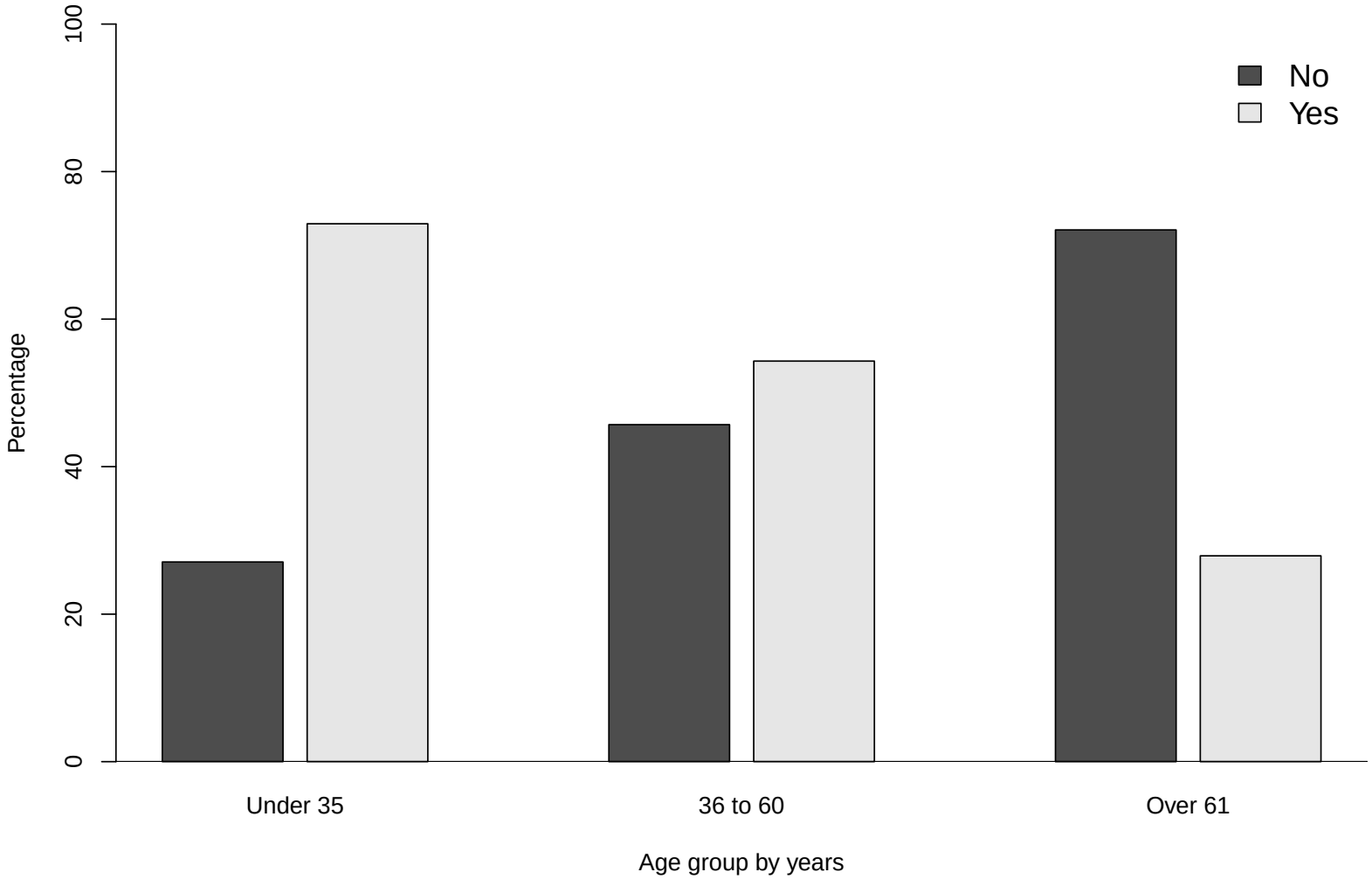


Further improvements

- Larger title for axes
- Get rid of the box drawn outide the legend
- Larger legend

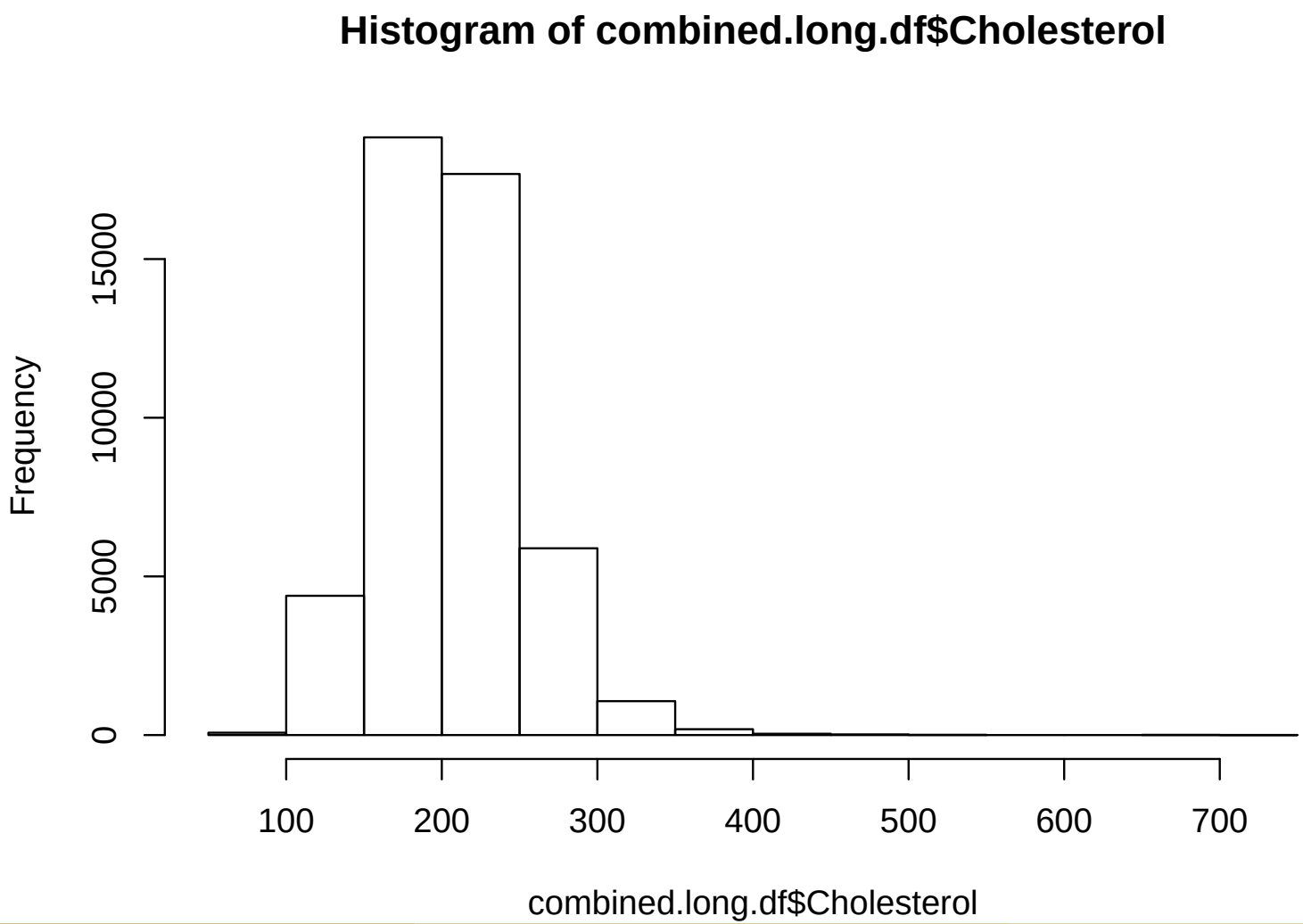
```
barplot(smoke.age, beside = TRUE,
        xlab = "Age group by years",
        ylab = "Percentage",
        ylim = c(0, 100),
        space = c(0.2, 1.5),
        legend.text = TRUE,
        args.legend = list(bty = "n", cex = 1.3))
abline(h = 0)
```

args.legend = list(bty = "n", cex = 1.3)



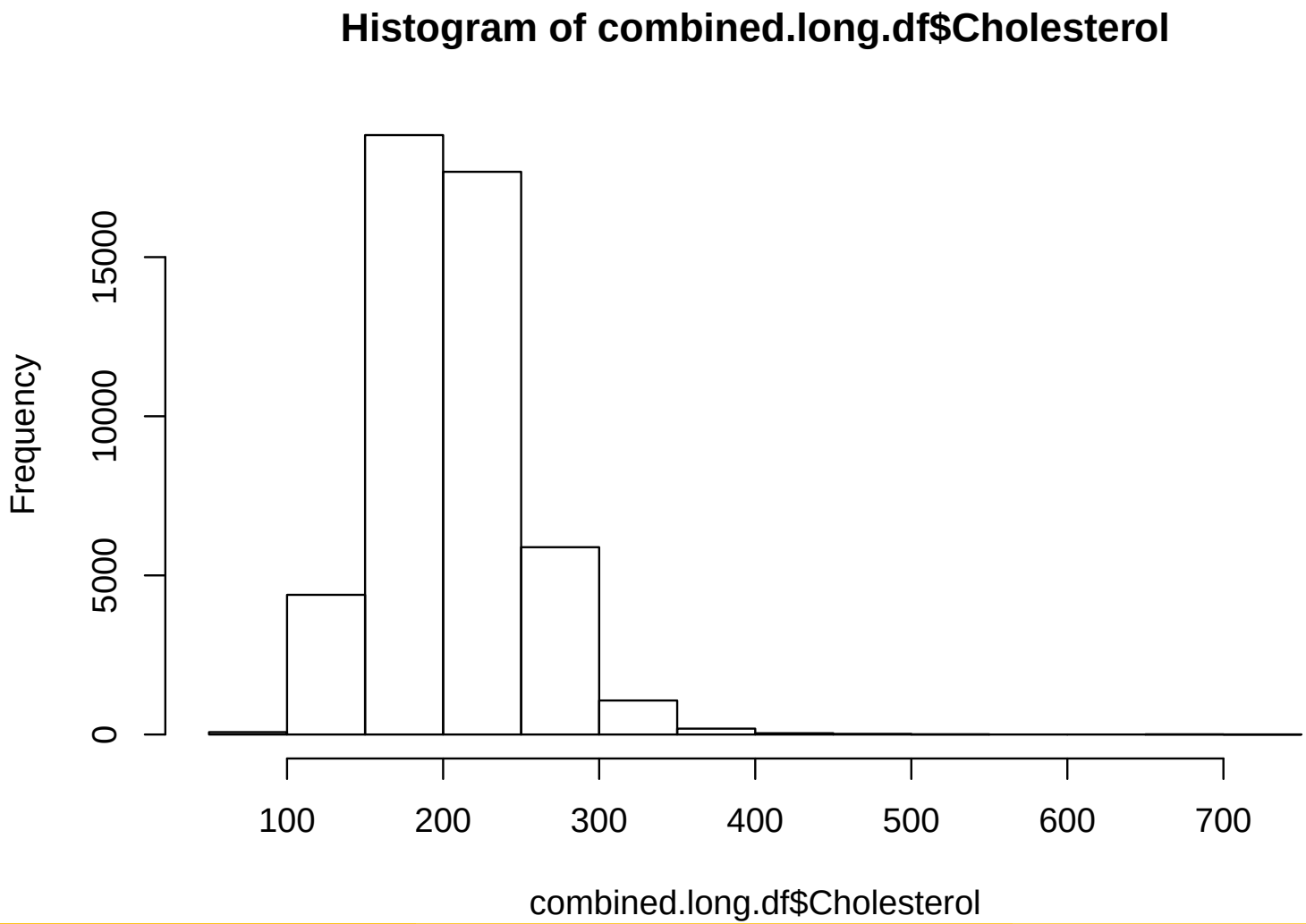
Histgrams

```
hist(combined.long.df$Cholesterol)
```



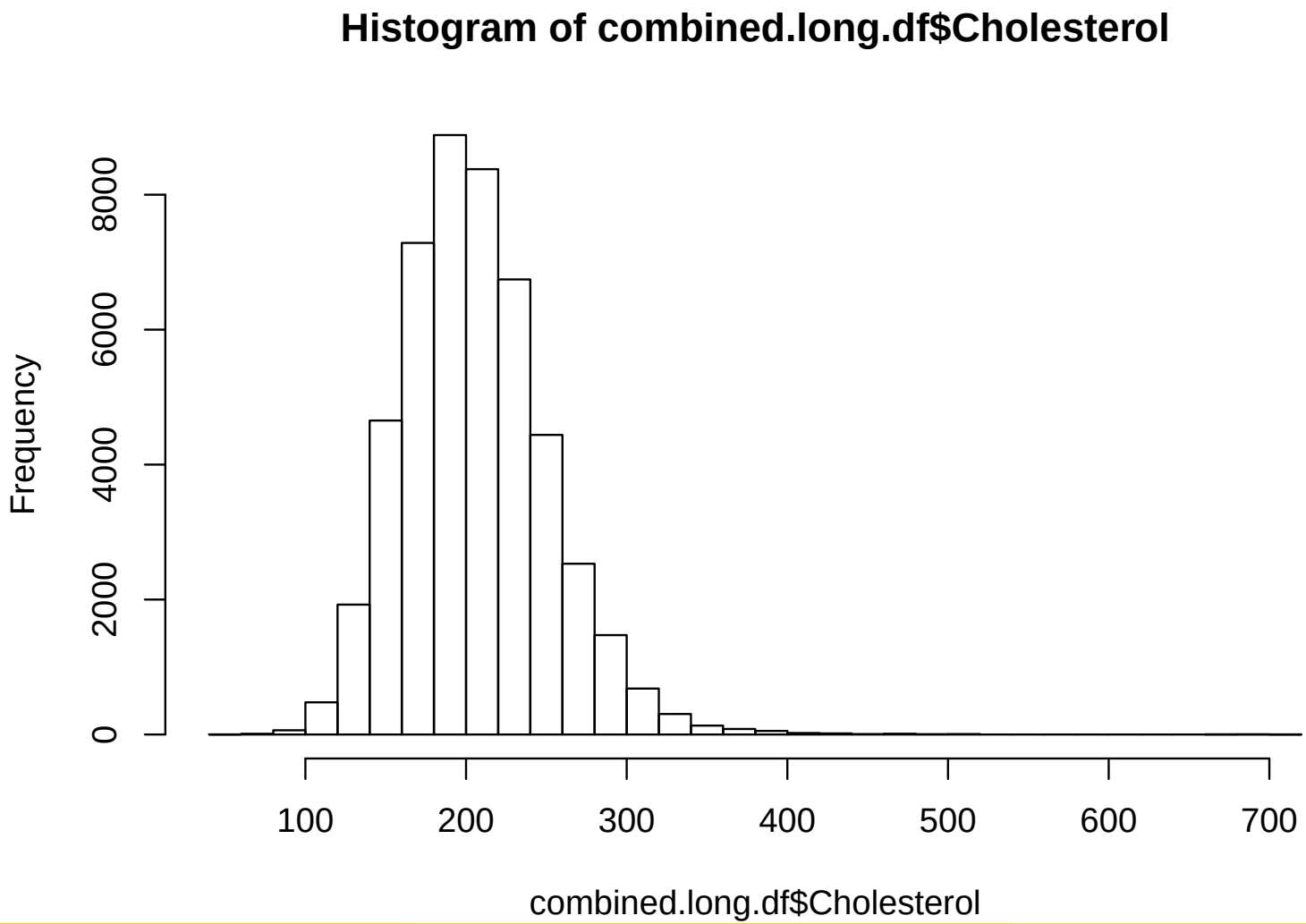
Fewer bins?

```
hist(combined.long.df$Cholesterol, breaks = 10)
```



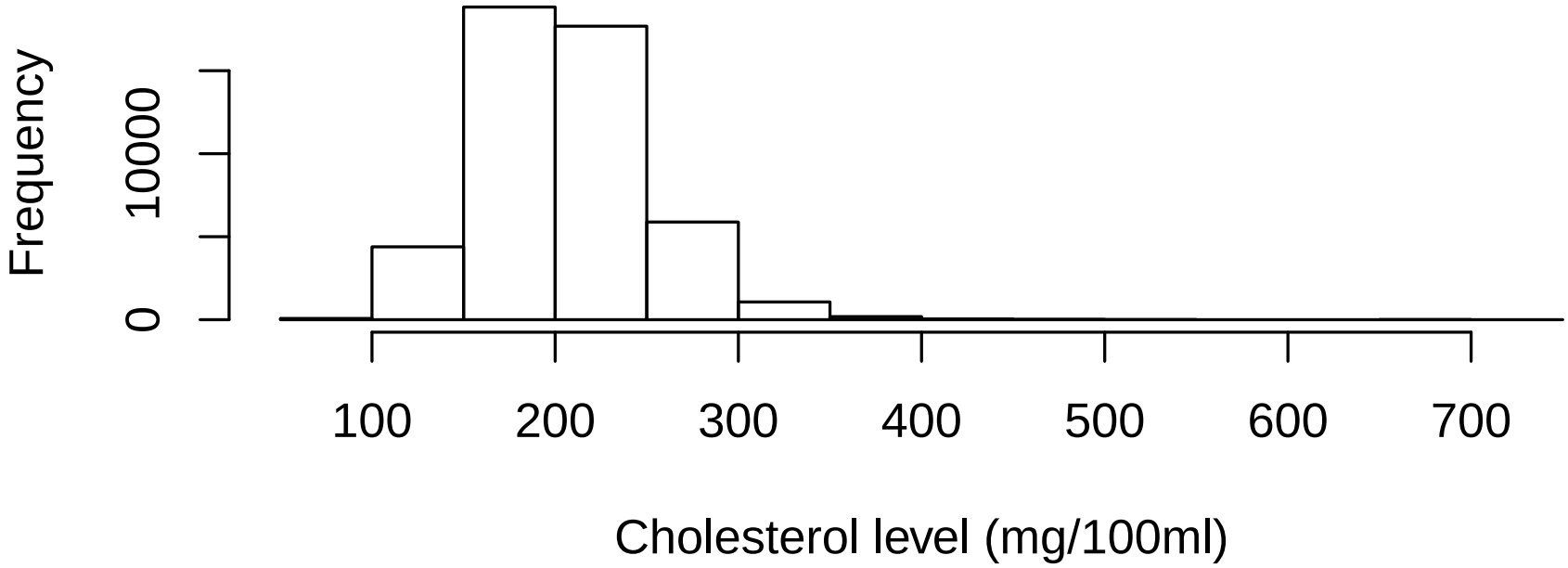
More bins?

```
hist(combined.long.df$Cholesterol, breaks = 40)
```



Always use meaningful axis labels

```
hist(combined.long.df$Cholesterol, main = NULL,
     xlab = "Cholesterol level (mg/100ml)")
```

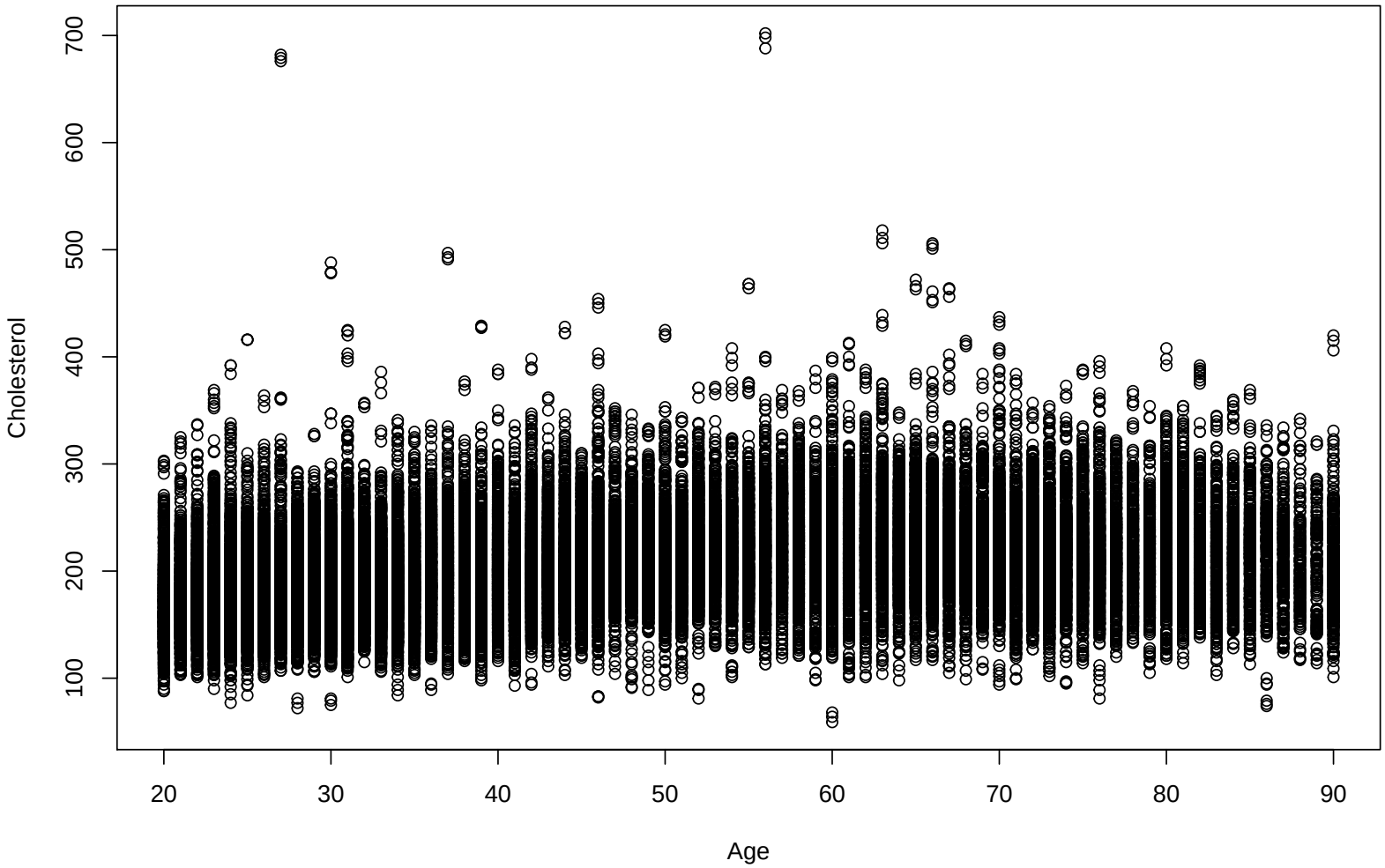


Scatter plots

- Used to dsplay paired quantiative (numeric) data, e.g. age and Cholesterol level.
- Make sure they are numeirc.
- Use `plot()` function.

```
with(combined.long.df, plot(Age, Cholesterol))
```

Scatter plots



type =

type = controls how data are plotted.

- `type = "p"`: points, this is the default,
- `type = "l"`: lines,
- `type = "b"`: both lines and points,
- `type = "n"`: no plotting, which can set up a plot for latter additions.

Plotting character

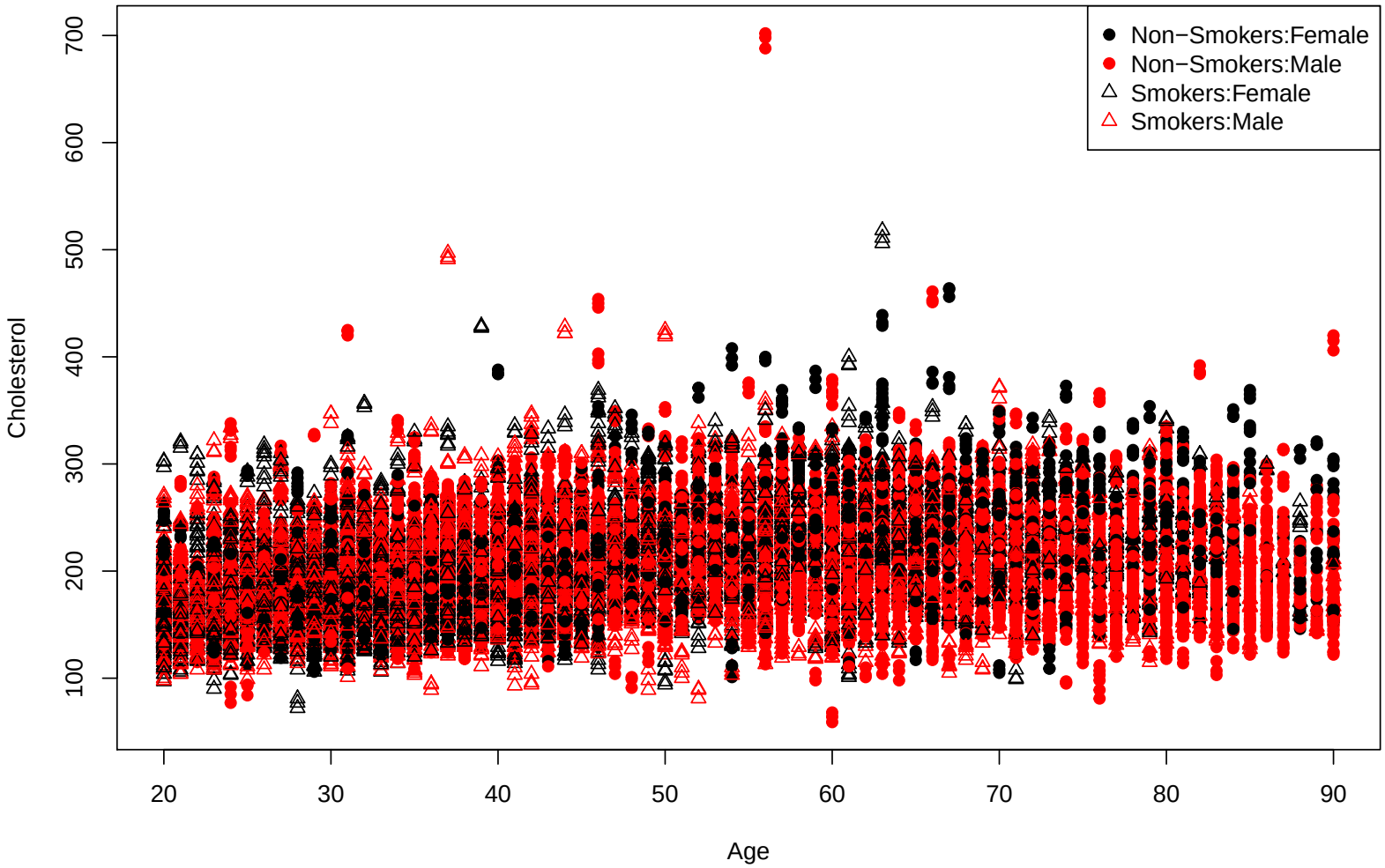
We will usually use different plotting character for different groups. Argument pch = in plot() or points() is used to change the plotting character:

- `pch = 19`: solid circle,
- `pch = 20`: smaller solid circle,
- `pch = 21`: filled circle,
- `pch = 22`: filled square,
- `pch = 23`: filled diamond,
- `pch = 24`: filled triangle point up,
- `pch = 25`: filled triangle point down.

Scatter plots

```
with(combined.long.df, plot(Age, Cholesterol,
     col = c("black", "red")[Sex],
     pch = c(19, 24)[Smoke.group]))
legend("topright", pch = c(19, 19, 24, 24),
       col = c("black", "red", "black", "red"),
       pt.bg = c("black", "red", NA, NA),
       legend = c("Non-Smokers:Female", "Non-Smokers:Male",
                  "Smokers:Female", "Smokers:Male"))
```

Scatter plots



Graphical Parameters

```
?par
op <- par(mfrow = c(2, 2))
  with(combined.long.df, boxplot(Cholesterol~Age.group,
      xlab = "Age Group", ylab = "Total score"))
  barplot(smoke.age, beside = TRUE, ylab = "Percentage",
      xlab = "Age group in years", ylim = c(0, 100),
      space = c(0.2, 1.5), legend.text = TRUE)
      abline(h = 0)
  hist(combined.long.df$Age)
  with(combined.long.df, plot(Age, Cholesterol))
## At end of plotting, reset to previous settings:
par(op)
```

Graphical Parameters