Base Graphics (part 2)

Graphics

Gaston Sanchez

R Coding Compendium

Donate

About

In this slides we cover the base graphics system: `"graphics"` packages

Base Graphics

Base Graphics in R

Traditional Graphics

- R `"graphics"` follows a static, "painting on canvas" model.
- Graphics elements are drawn, and remain visible until painted over.
- For dynamic and/or interactive graphics, R is limited.

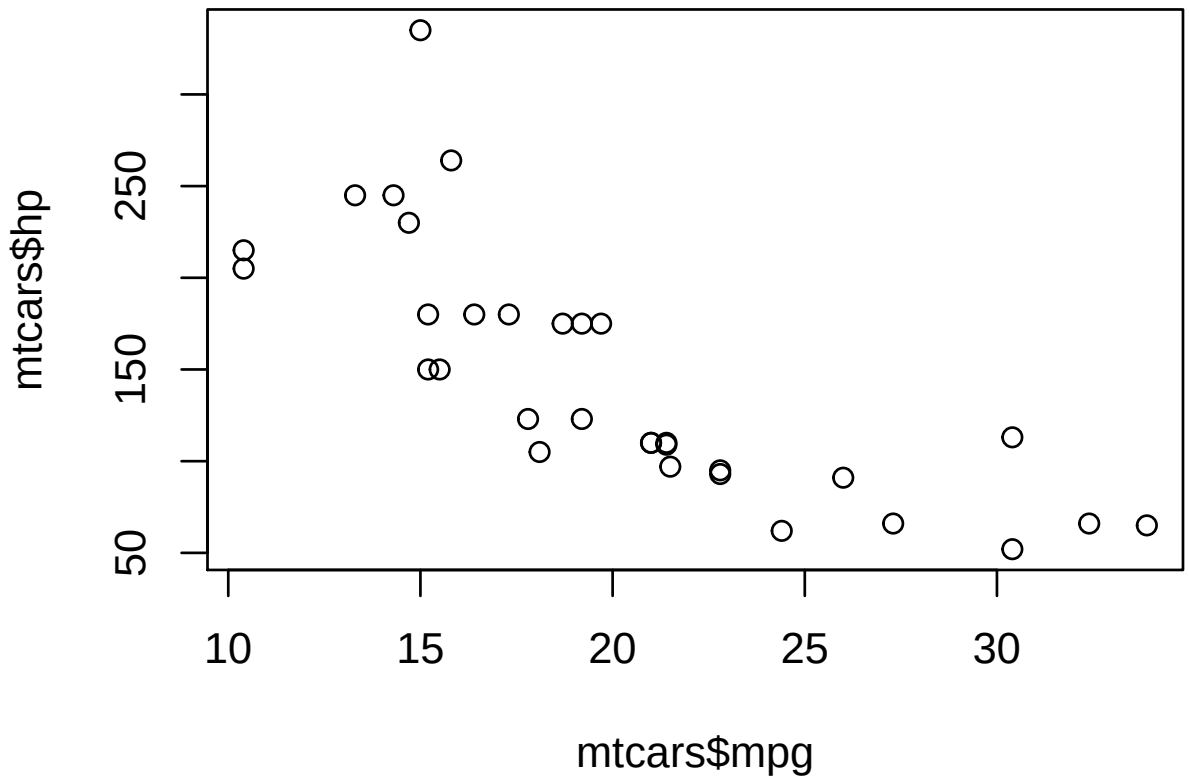
Traditional Graphics in R

In the traditional model, we create a plot by first calling a high-level function that creates a complete plot, and then we call low-level functions to add more output if necessary

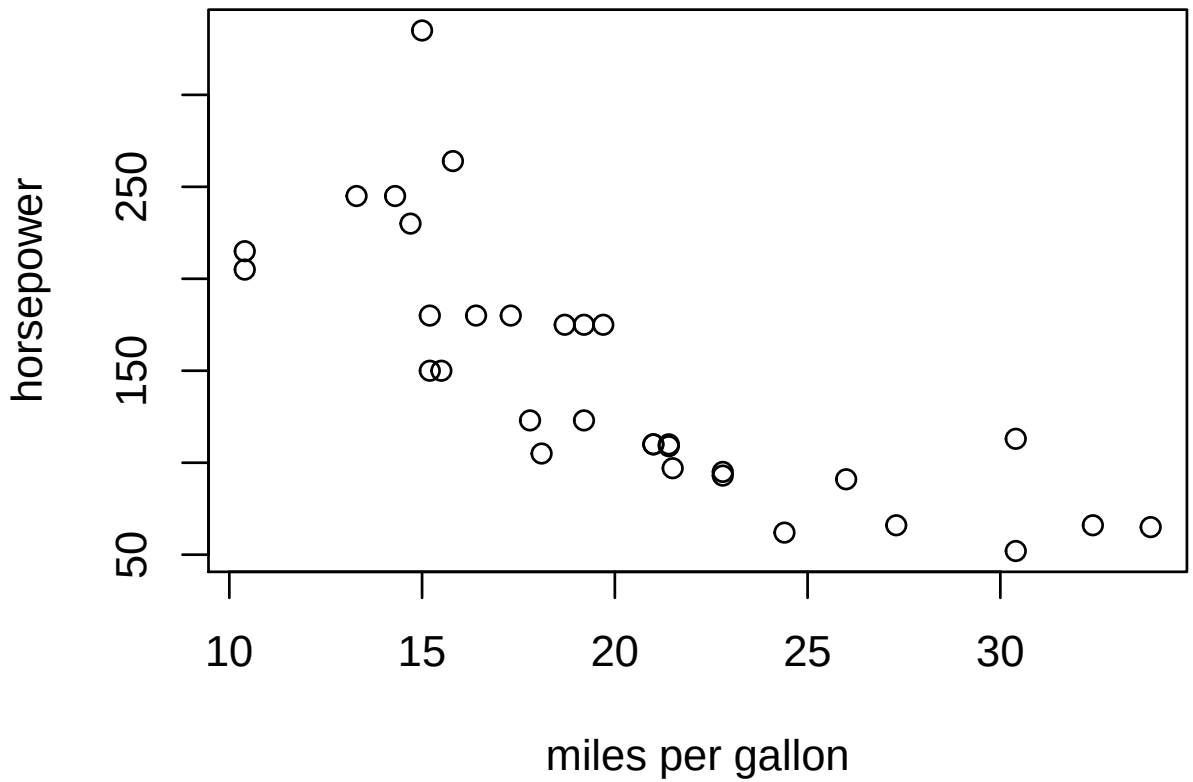
```r
head(mtcars)
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

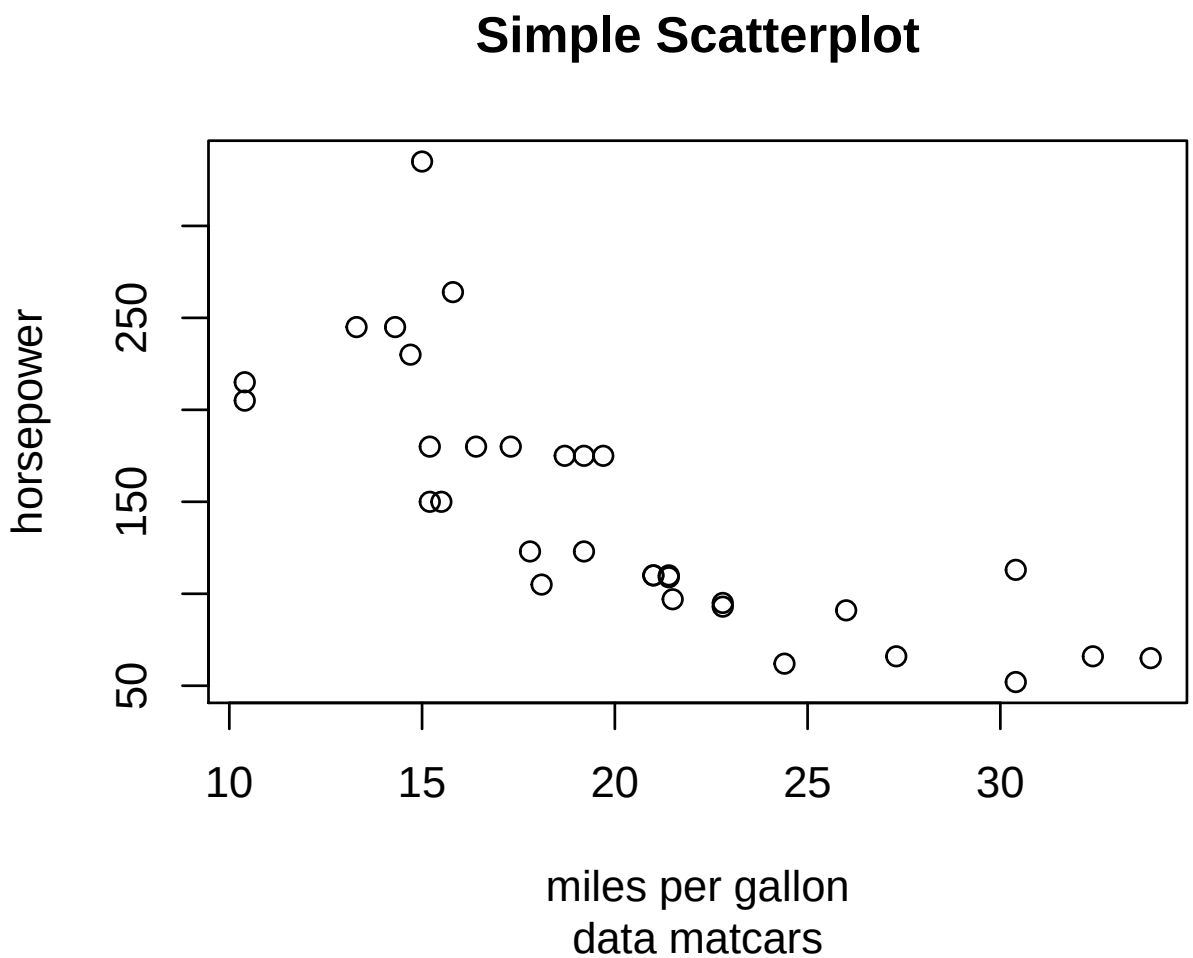
```r
# simple scatter-plot
plot(mtcars$mpg, mtcars$hp)
```



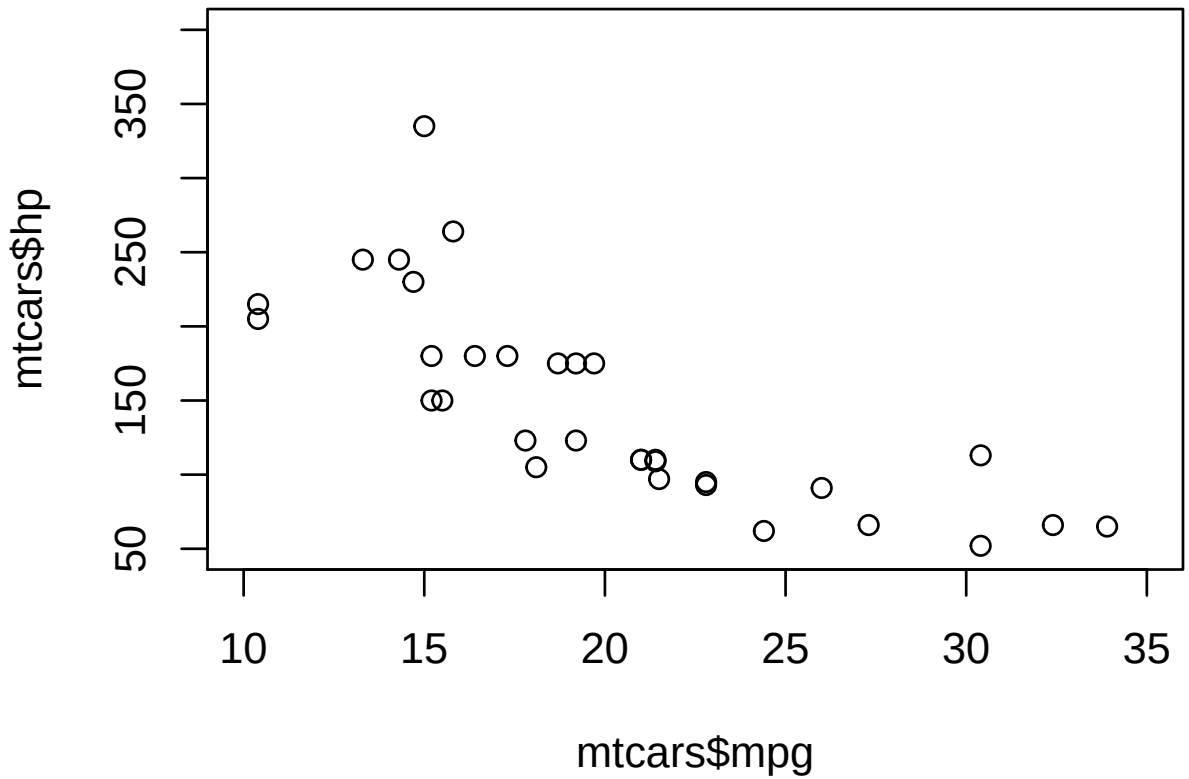
```r
# x-axis and y-axis labels
plot(mtcars$mpg, mtcars$hp,
     xlab = "miles per gallon",
     ylab = "horsepower")
```



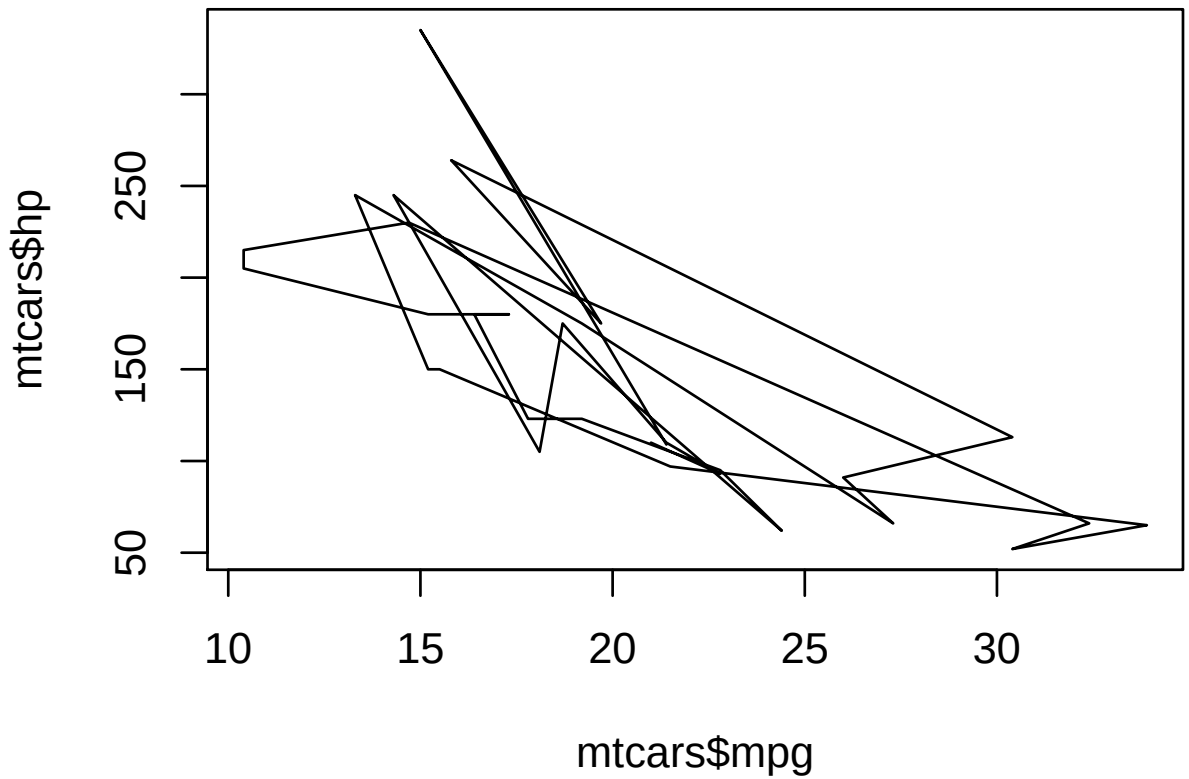
```r
# title and subtitle
plot(mtcars$mpg, mtcars$hp,
     xlab = "miles per gallon", ylab = "horsepower",
     main = "Simple Scatterplot", sub = 'data matcars')
```



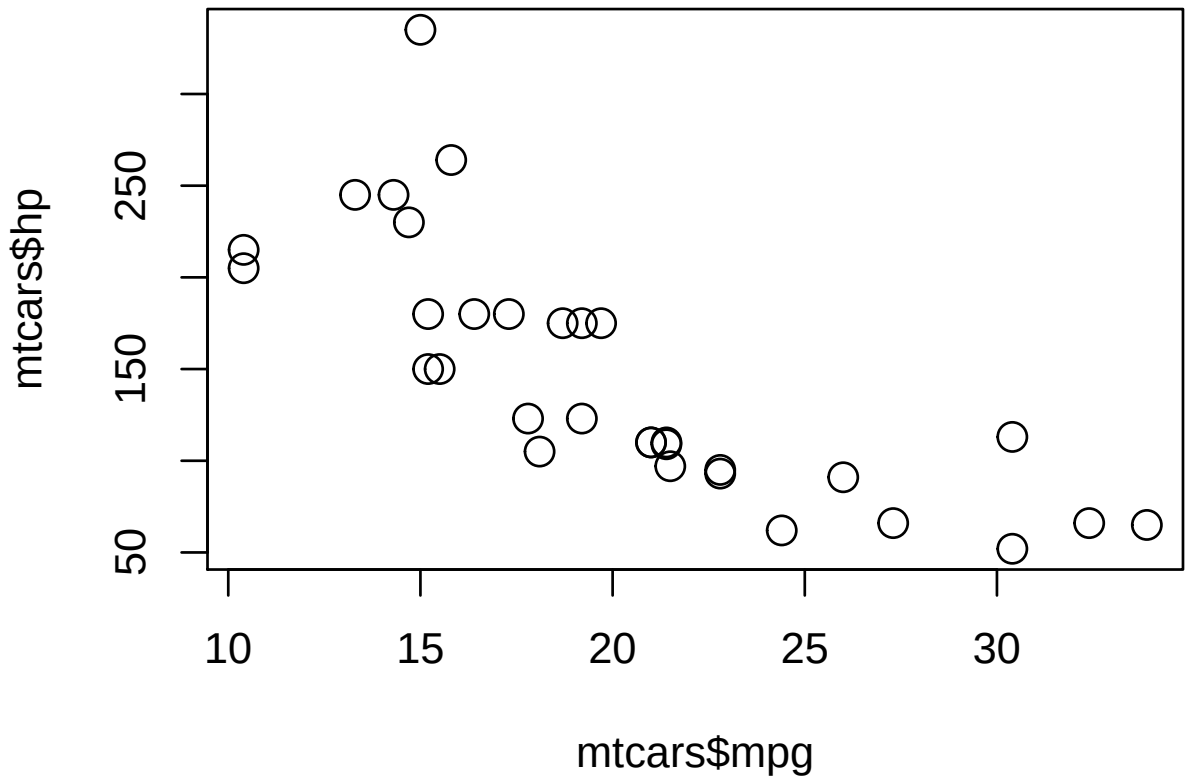
```r
# 'xlim' and 'ylim'
plot(mtcars$mpg, mtcars$hp, xlim = c(10, 35), ylim = c(50, 400))
```



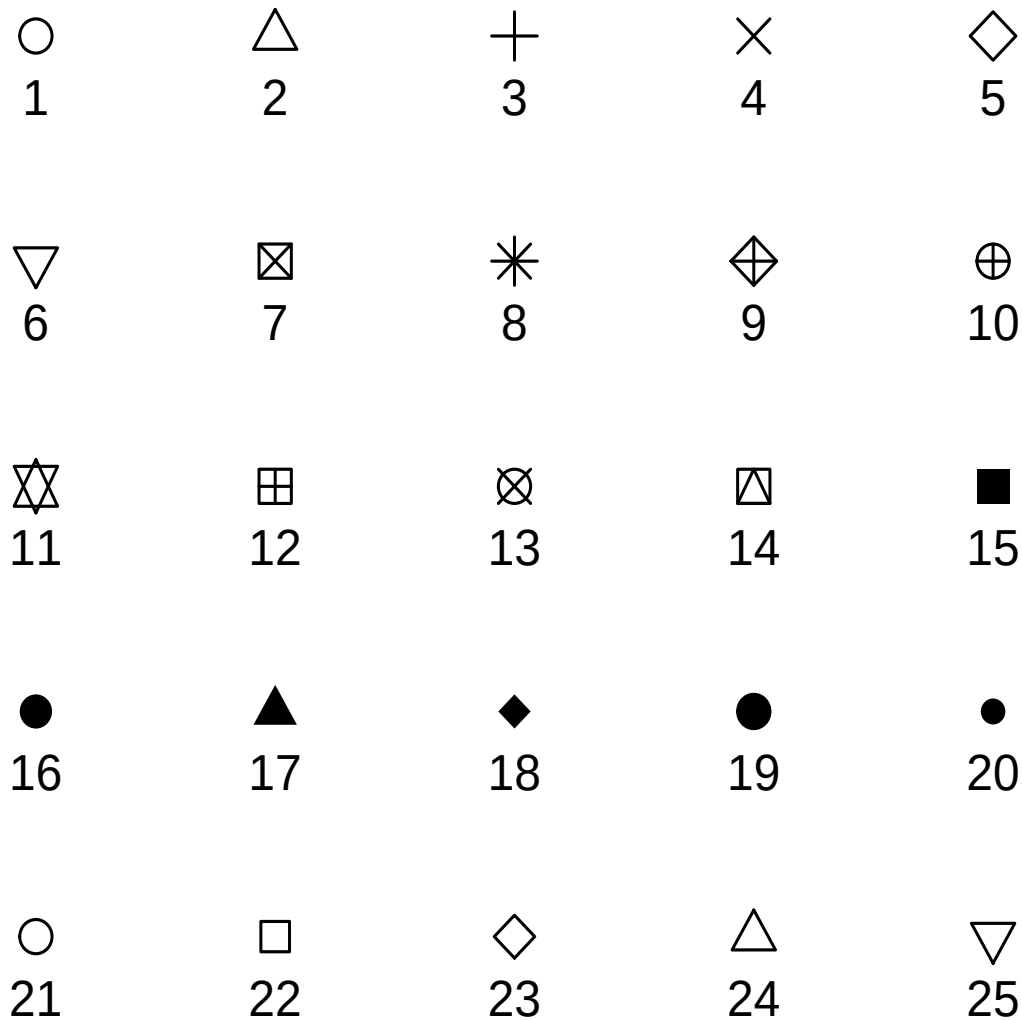
```r
# using 'type' (e.g. lines)
plot(mtcars$mpg, mtcars$hp, type = "l")
```



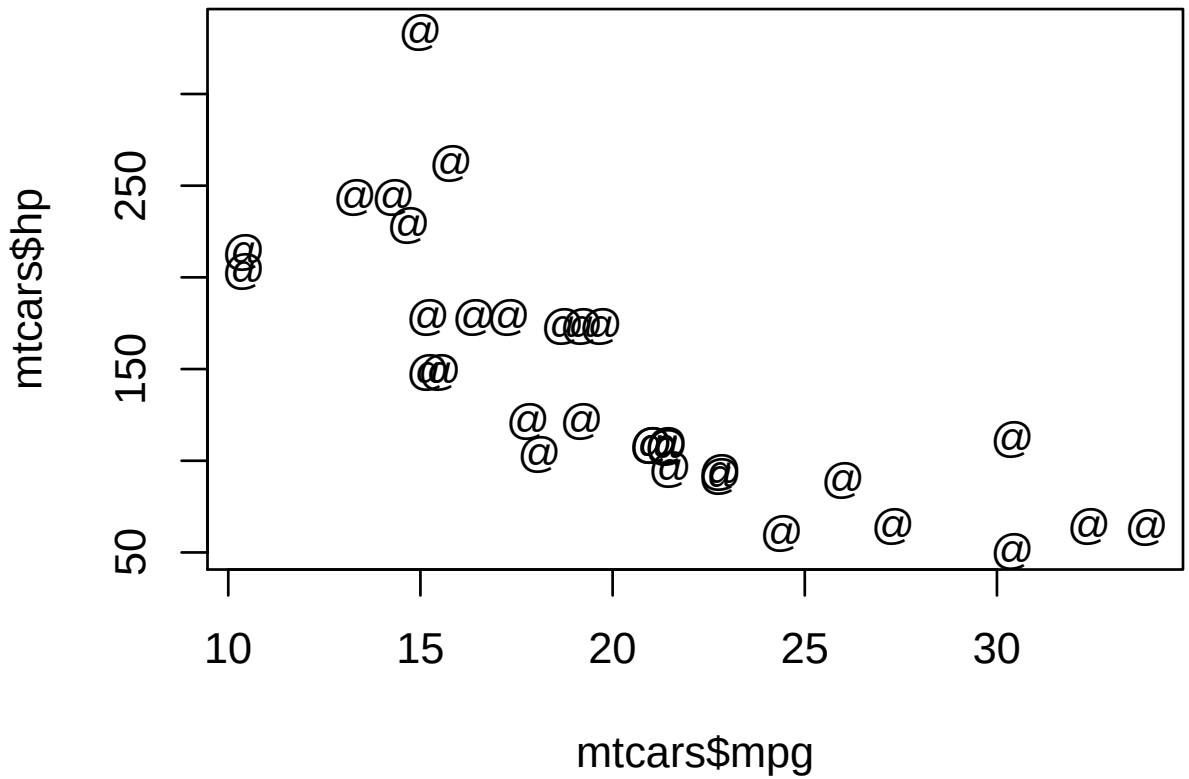
```r
# character expansion 'cex', and point character 'pch'
plot(mtcars$mpg, mtcars$hp, cex = 1.5, pch = 1)
```



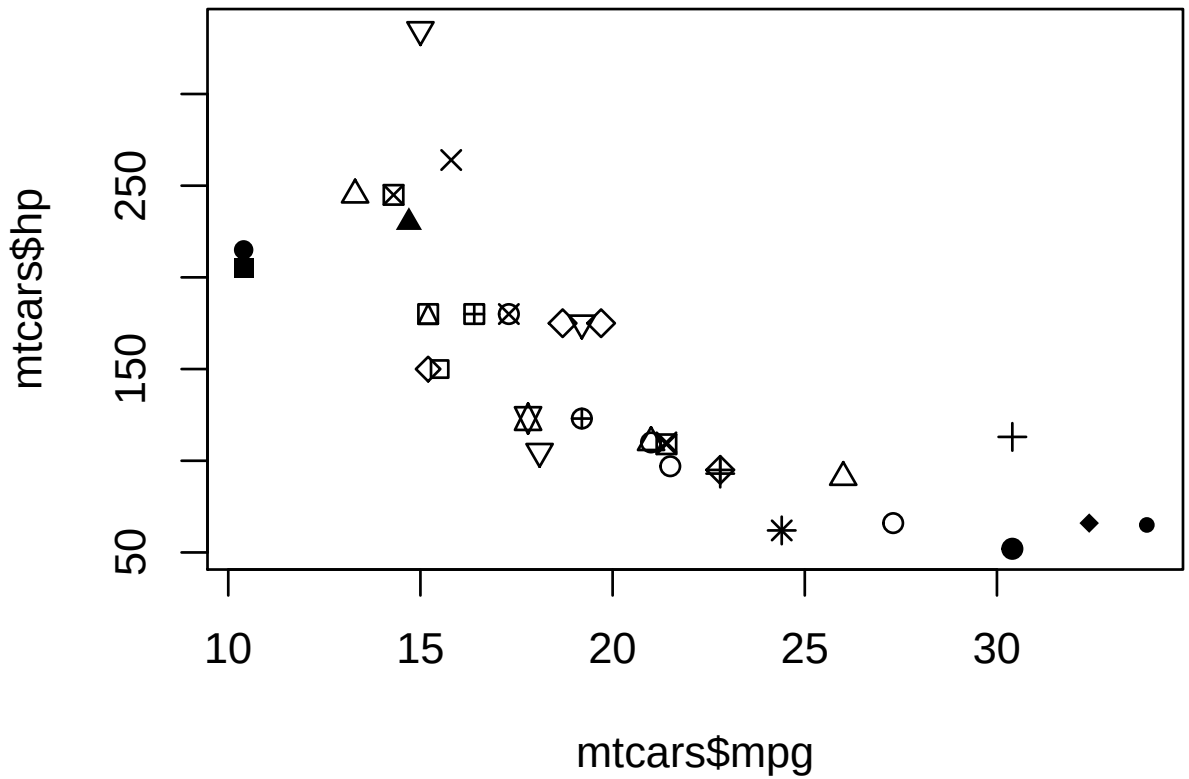
Point symbols (`pch`) available in R



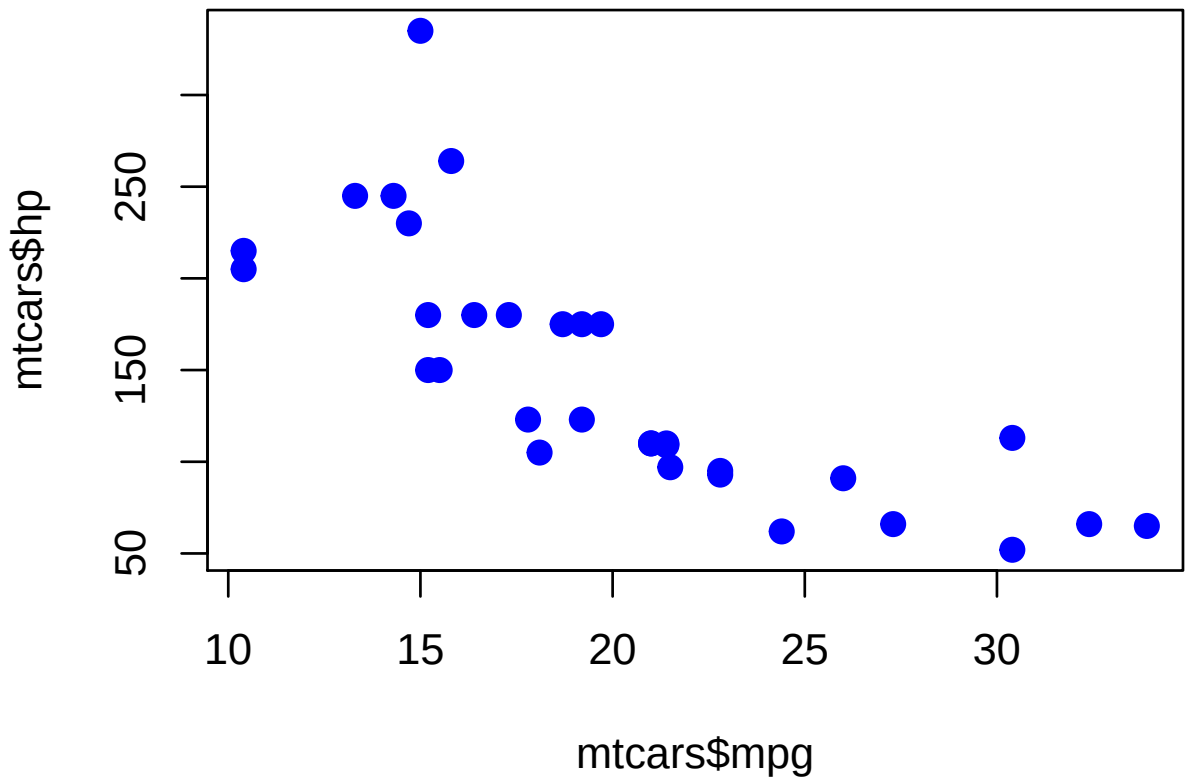
```r
# 'pch' can be any character
plot(mtcars$mpg, mtcars$hp, pch = "@")
```



```r
# 'pch' symbols will be recycled
plot(mtcars$mpg, mtcars$hp, pch = 1:25)
```



```r
# color argument 'col'
plot(mtcars$mpg, mtcars$hp, pch = 19, col = "blue", cex = 1.2)
```

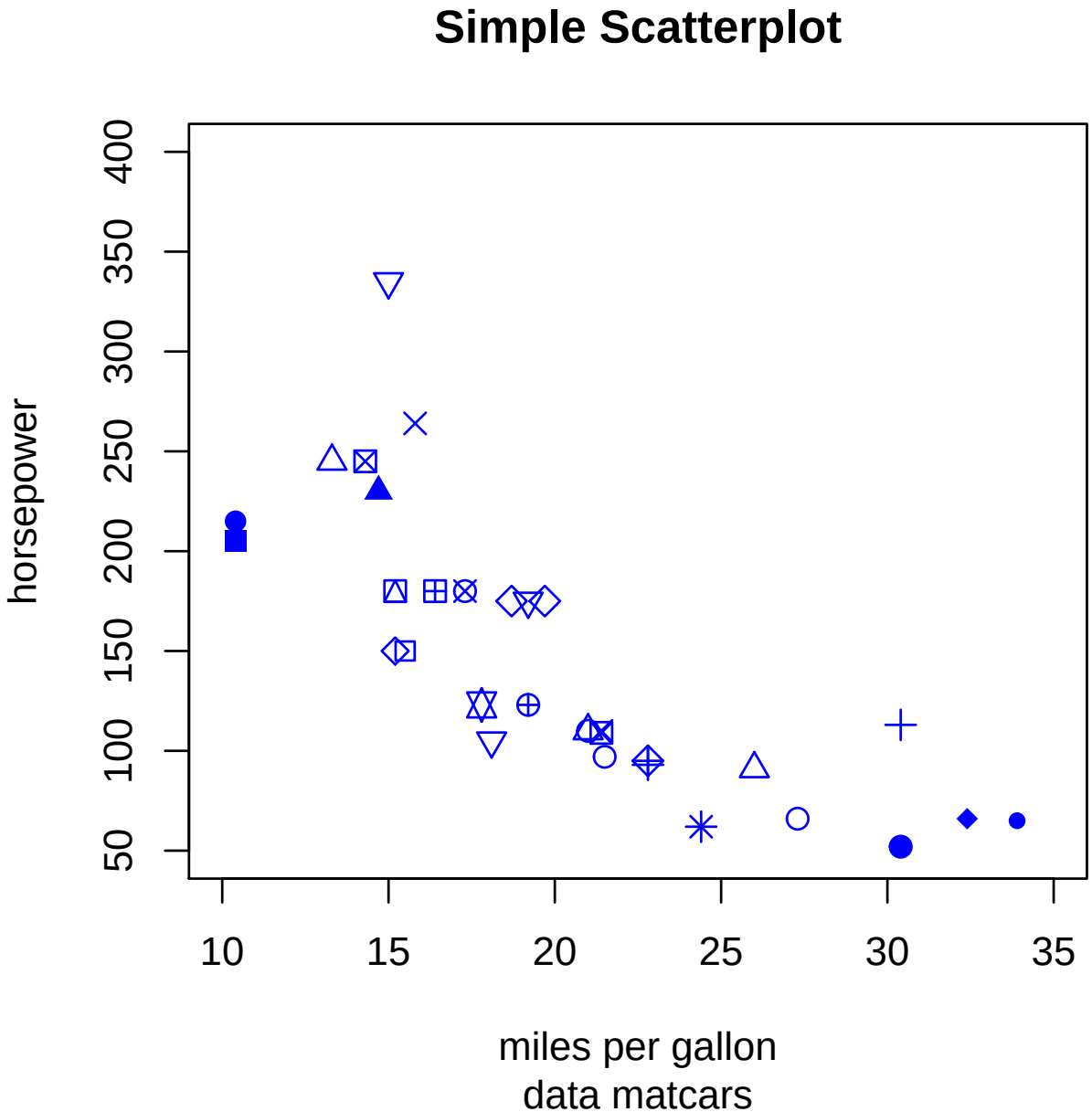


Coloring Point Symbols

- the `col` argument can be used to color symbols
- symbols 21 through 25 can additionally have their interiors filled by using the `bg` (background) argument

Coloring Point Symbols

```r
# using plot()
plot(mtcars$mpg,
     mtcars$hp,
     xlim = c(10, 35),
     ylim = c(50, 400),
     xlab = "miles per gallon",
     ylab = "horsepower",
     main = "Simple Scatterplot",
     sub = "data matcars",
     pch = 1:25,
     cex = 1.2,
     col = "blue")
```

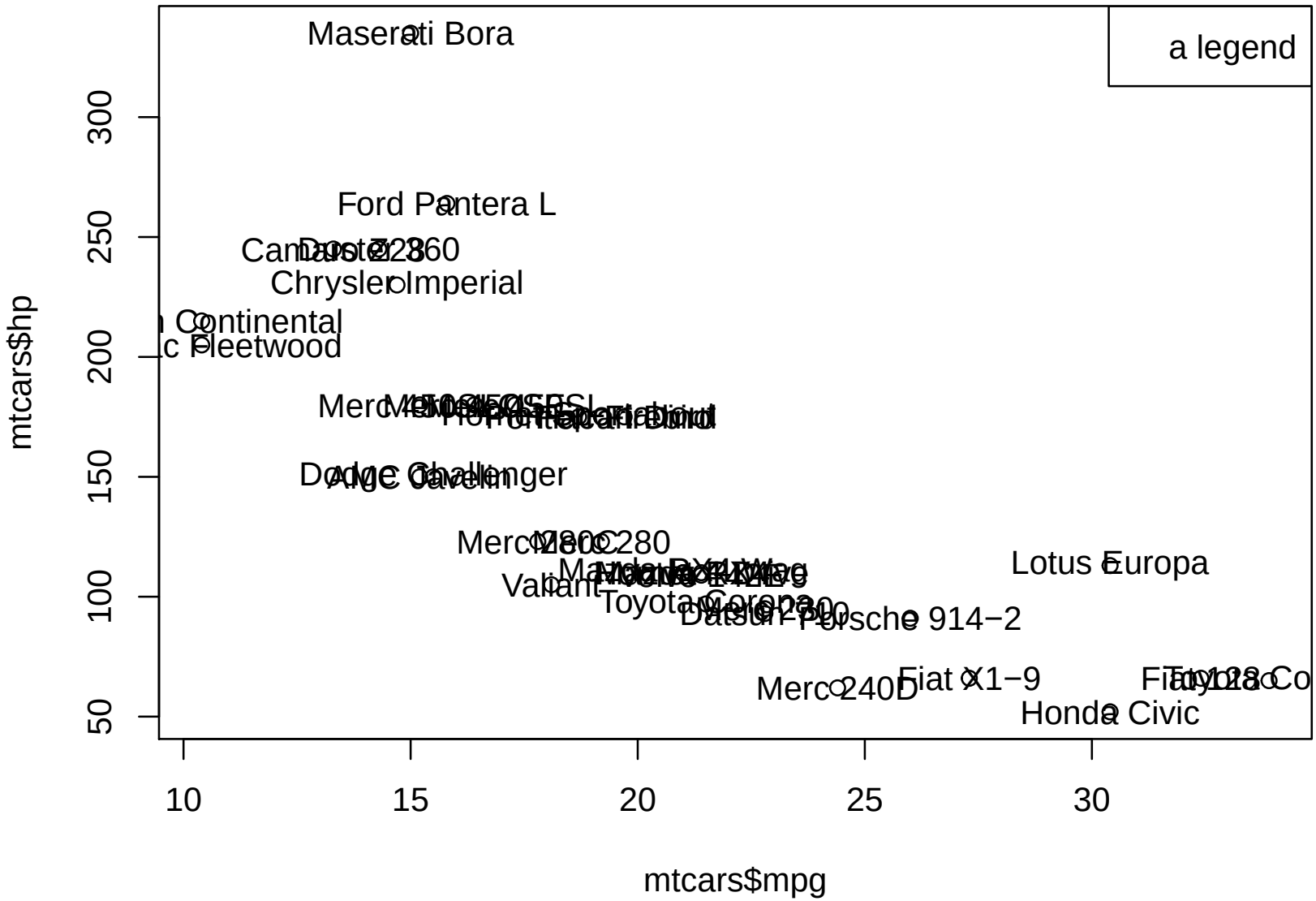

Simple Scatterplot
horsepower
50
100
150
200
250
300
350
400
10
15
20
25
30
35
miles per gallon
data matcars

Low-Level Functions

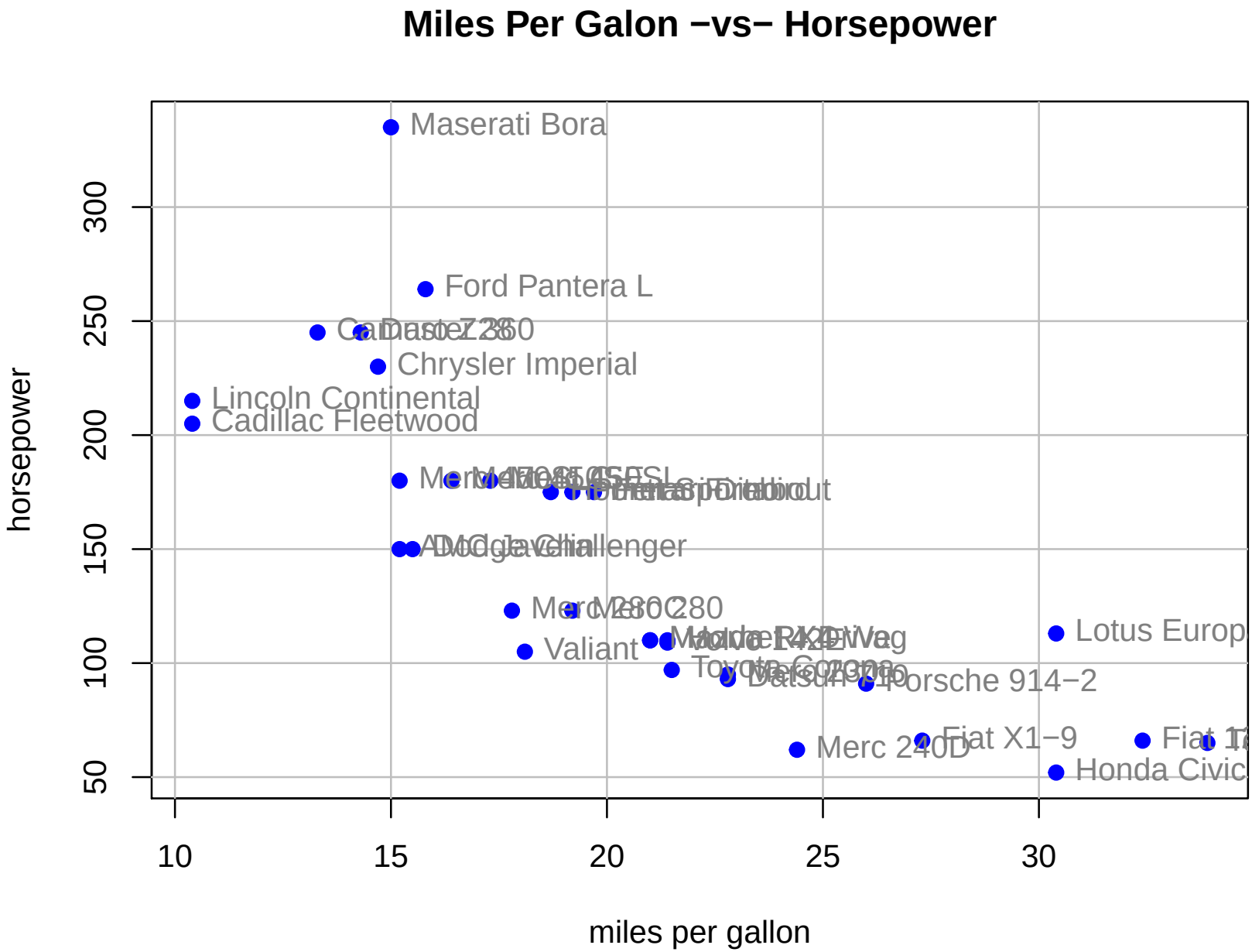
High and Low level functions

- Usually we call a high-level function
- Most times we change the default arguments
- Then we call low-level functions

```r
# simple scatter-plot
plot(mtcars$mpg, mtcars$hp)

# adding text
text(mtcars$mpg, mtcars$hp, labels = rownames(mtcars))

# dummy legend
legend("topright", legend = "a legend")

# graphic title
title("Miles Per Galon -vs- Horsepower")
```


Miles Per Galon −vs− Horsepower
a legend
Maserati Bora
Ford Pantera L
Chrysler Imperial
Lotus Europa
Porsche 914−2
Fiat X1−9
Merc 240D
Honda Civic
mtcars$hp
mtcars$mpg
50
100
150
200
250
300
10
15
20
25
30

```r
# simple scatter-plot
plot(mtcars$mpg, mtcars$hp, type = "n",
     xlab = "miles per gallon", ylab = "horsepower")
# grid lines
abline(v = seq(from = 10, to = 30, by = 5), col = 'gray')
abline(h = seq(from = 50, to = 300, by = 50), col = ' gray')
# plot points
points(mtcars$mpg, mtcars$hp, pch = 19, col = "blue")
# plot text
text(mtcars$mpg, mtcars$hp, labels = rownames(mtcars),
     pos = 4, col = "gray50")
# graphic title
title("Miles Per Galon -vs- Horsepower")
```

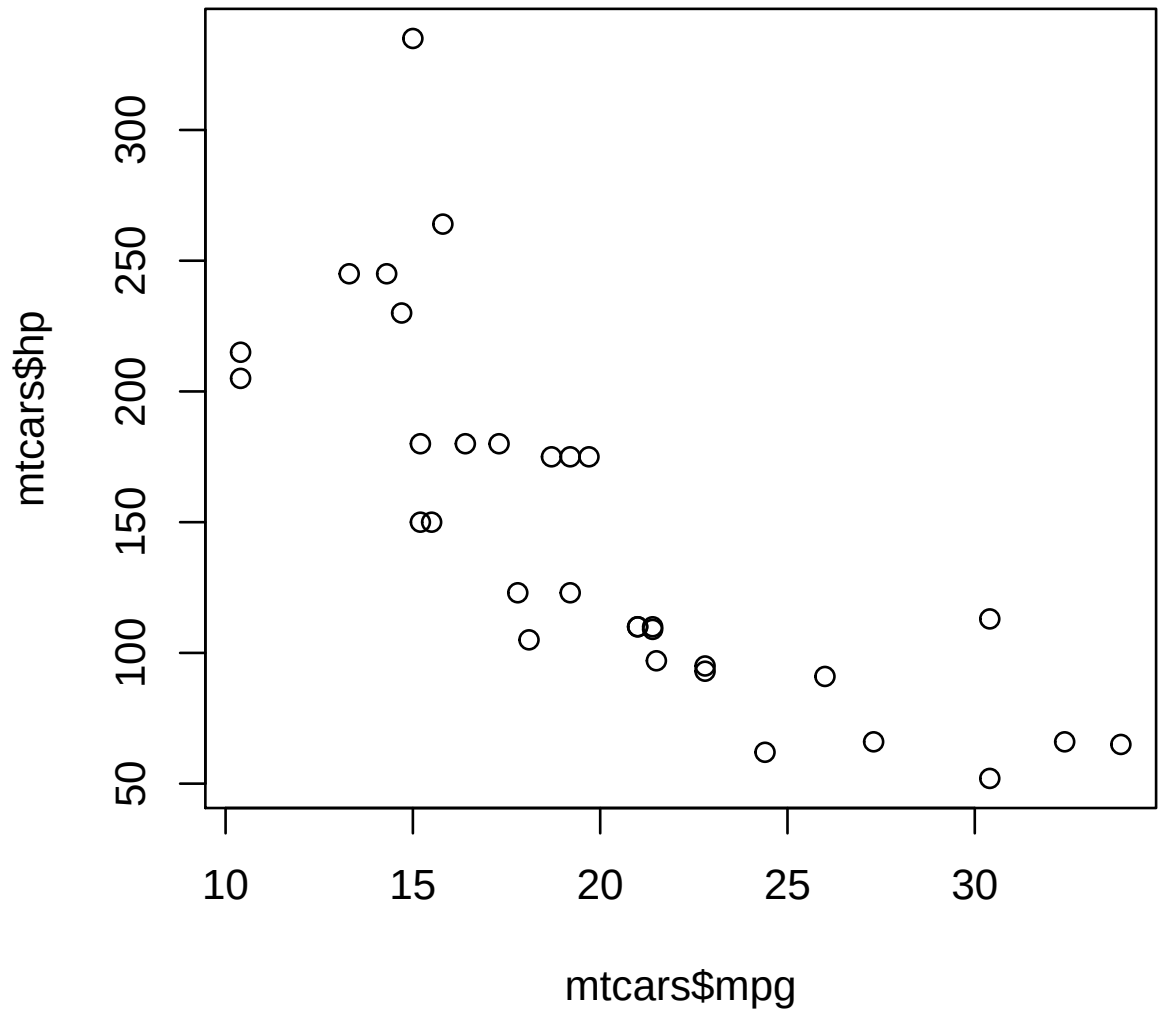

Miles Per Galon −vs− Horsepower
horsepower
miles per gallon
50
100
150
200
250
300
10
15
20
25
30
Maserati Bora
Ford Pantera L
Chrysler Imperial
Lincoln Continental
Cadillac Fleetwood
Merc 280C
Valiant
Lotus Europ
Porsche 914−2
Fiat X1−9
Merc 240D
Honda Civic

Low-level functions

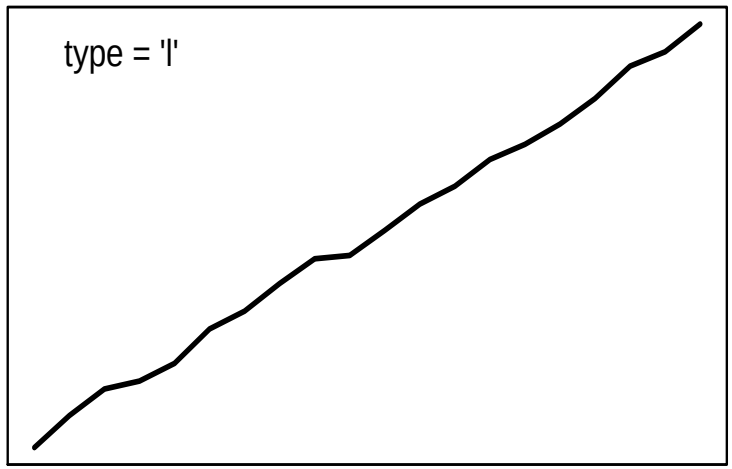
High-level graphics of a single variable

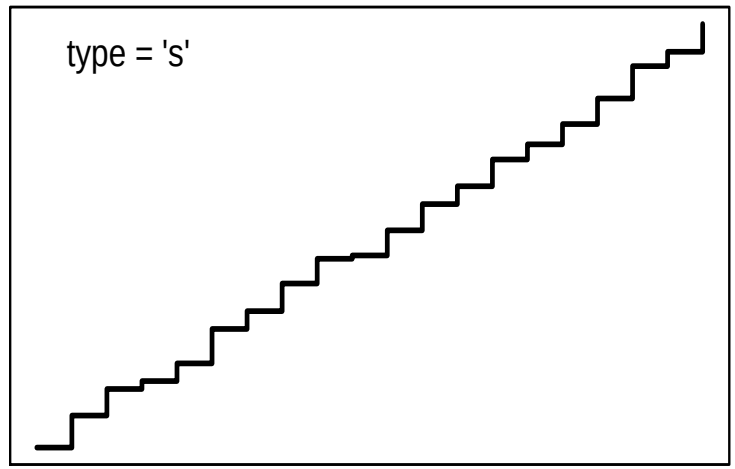
Function	Description
`points()`	points
`lines()`	connected line segments
`abline()`	straight lines across a plot
`segments()`	disconnected line segments
`arrows()`	arrows
`rect()`	rectangles
`polygon()`	polygons
`text()`	text
`symbols()`	various symbols
`legend()`	legends

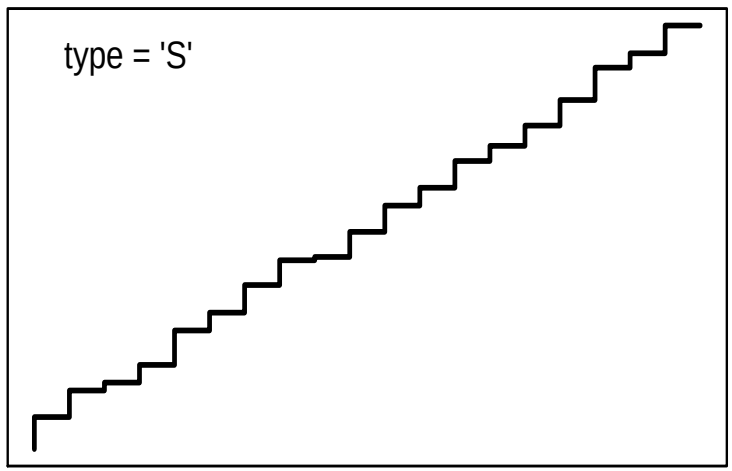
```r
plot(mtcars$mpg, mtcars$hp, type = "n")
points(mtcars$mpg, mtcars$hp)
```

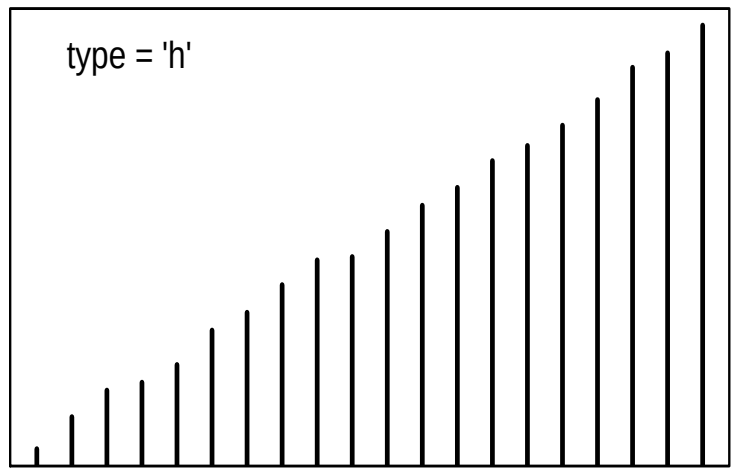


Line Graph Options

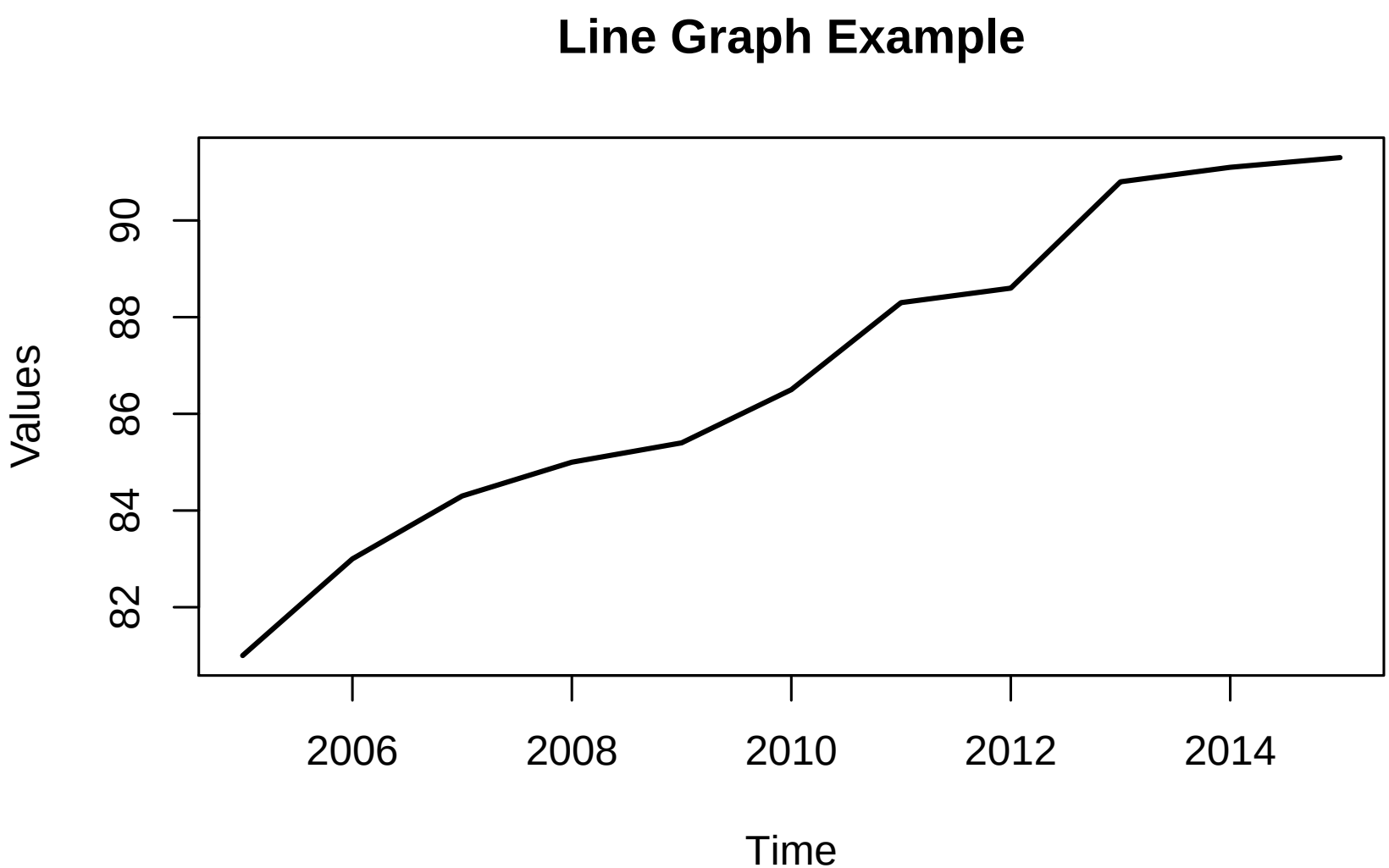






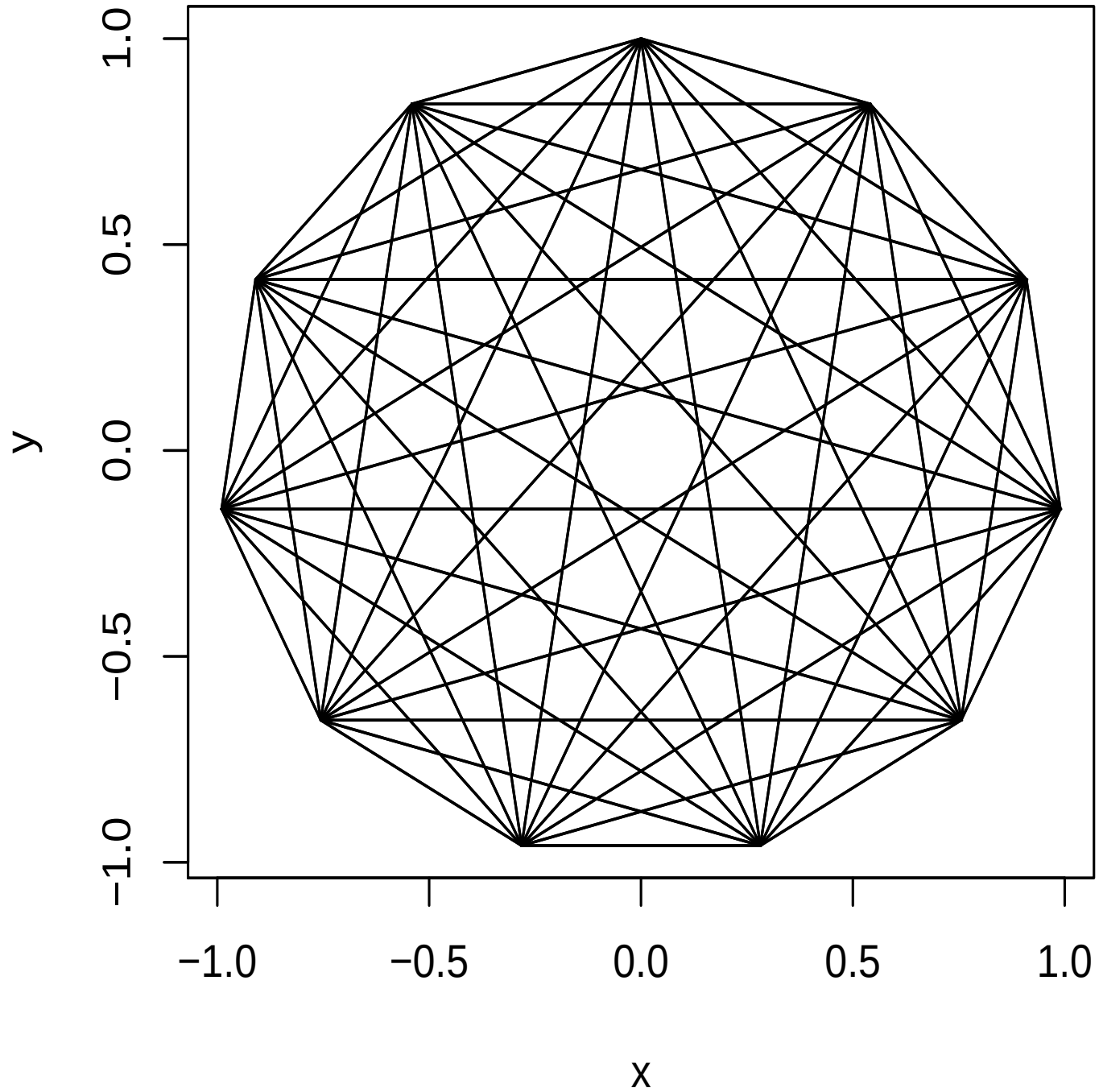


```r
x <- 2005:2015

y <- c(81, 83, 84.3, 85, 85.4, 86.5, 88.3,
       88.6, 90.8, 91.1, 91.3)

plot(x, y, type = 'n', xlab = "Time", ylab = "Values")
lines(x, y, lwd = 2)
title(main = "Line Graph Example")
```


Line Graph Example
Values
82
84
86
88
90
2006
2008
2010
2012
2014
Time

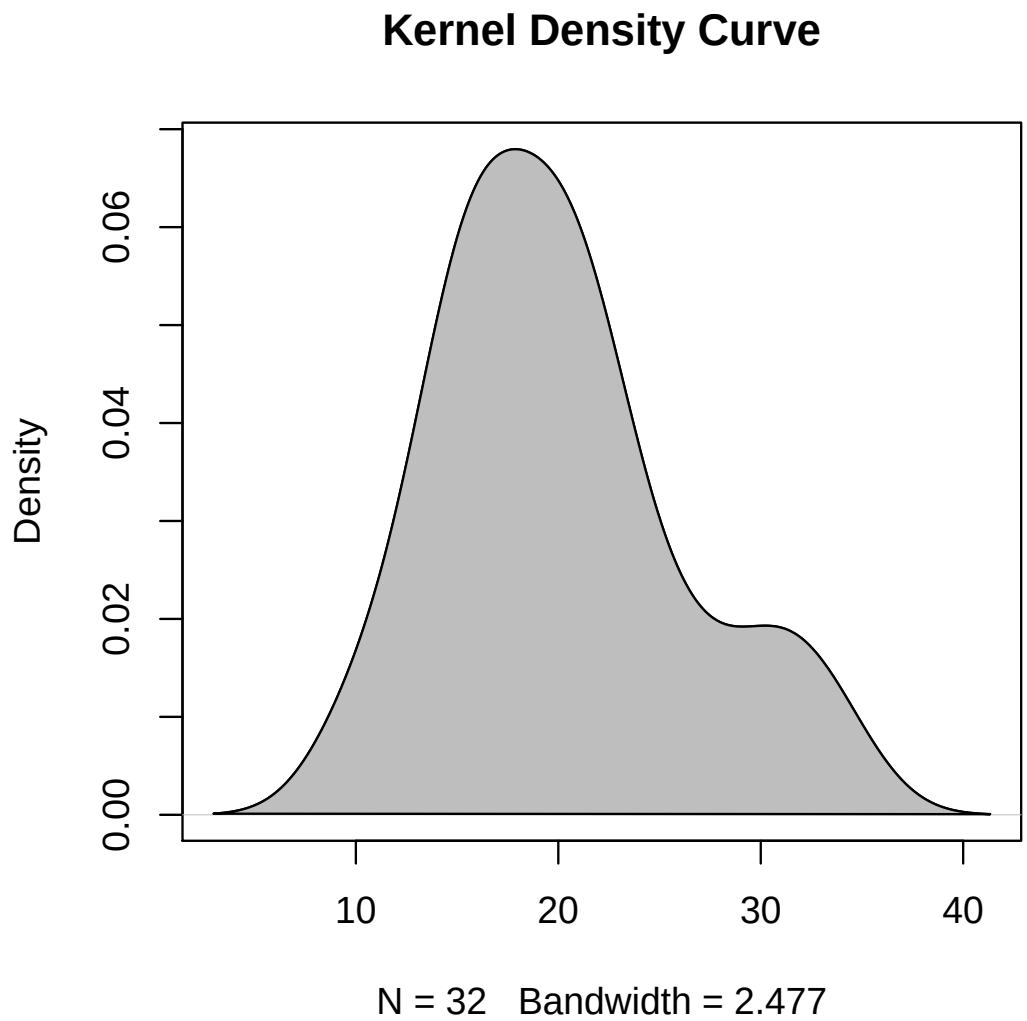
Drawing Line Segments

```
n <- 11
theta <- seq(0, 2 * pi, length = n + 1)[1:n]
x <- sin(theta)
y <- cos(theta)
v1 <- rep(1:n, n)
v2 <- rep(1:n, rep(n, n))

plot(x, y, type = 'n')
segments(x[v1], y[v1], x[v2], y[v2])
```


1.0
0.5
0.0
−0.5
−1.0
y
−1.0
−0.5
0.0
0.5
1.0
x

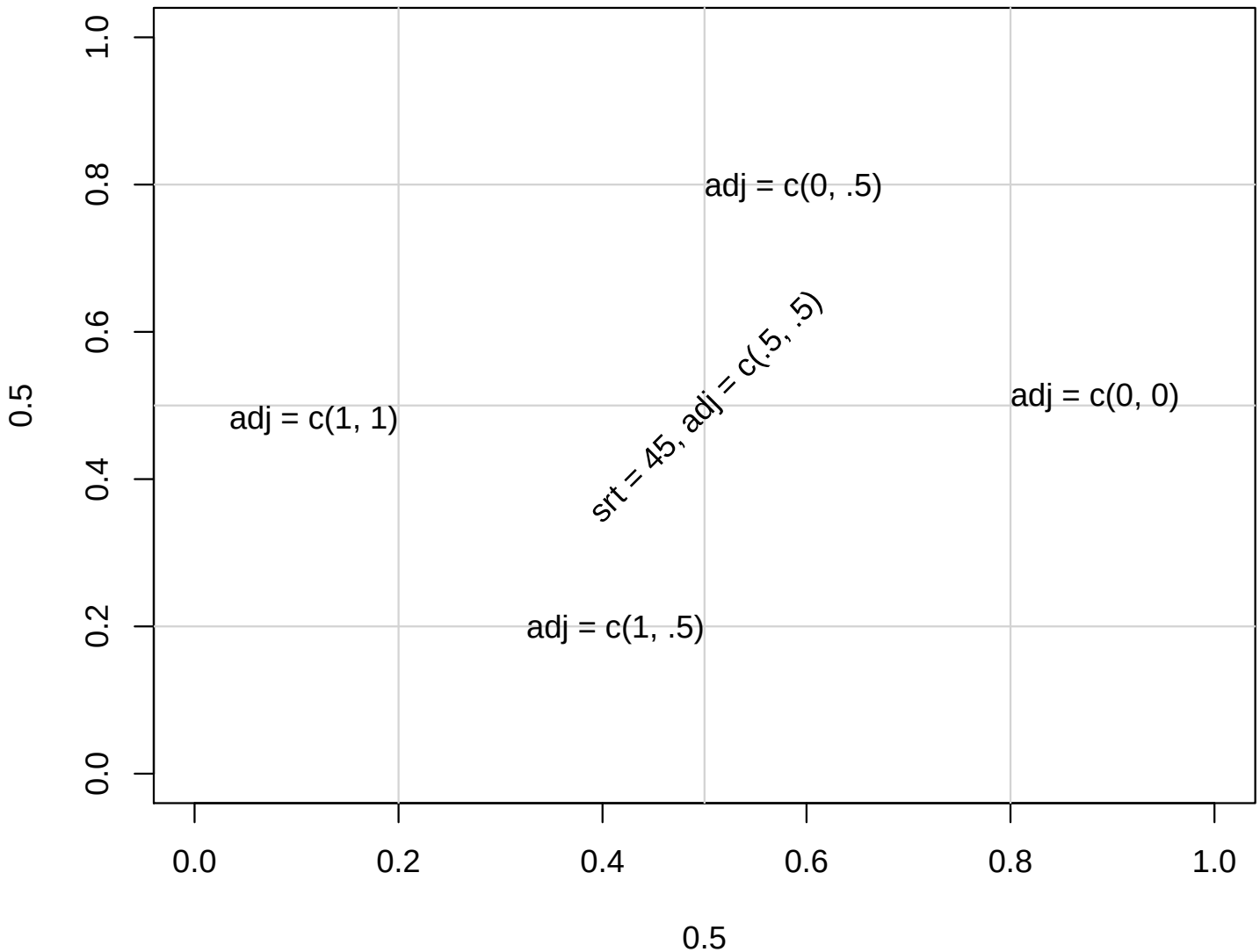
Drawing Polygons

```
mpg_dens <- density(mtcars$mpg)

plot(mpg_dens, main = "Kernel Density Curve")
polygon(mpg_dens, col = 'gray')
```

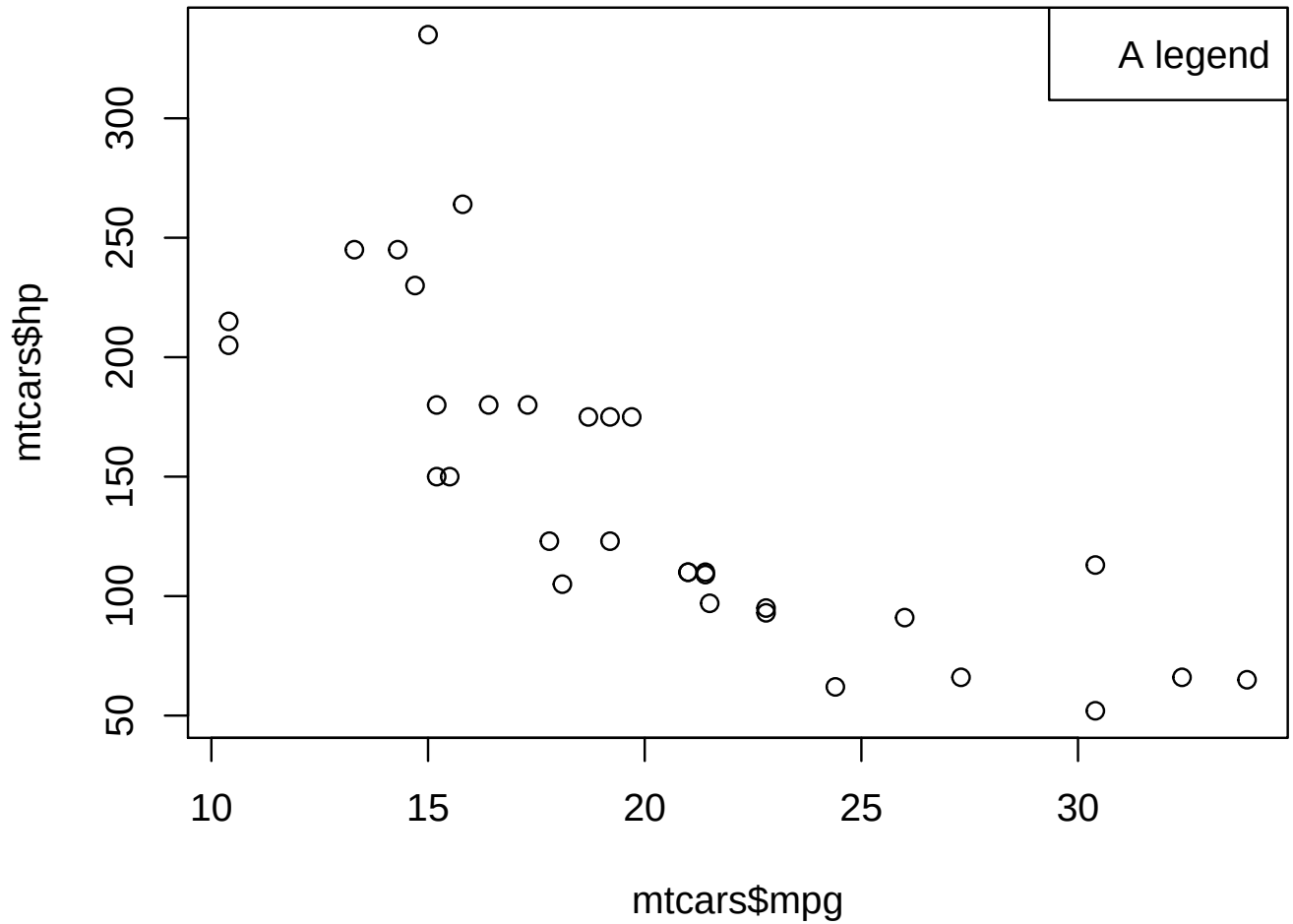

Kernel Density Curve
Density
0.00
0.02
0.04
0.06
10
20
30
40
N = 32 Bandwidth = 2.477

Drawing Text

```
plot(0.5, 0.5, xlim = c(0, 1), ylim = c(0, 1), type = 'n')
abline(h = c(.2, .5, .8),
       v = c(.5, .2, .8), col = "lightgrey")
text(0.5, 0.5, "srt = 45, adj = c(.5, .5)",
     srt = 45, adj = c(.5, .5))
text(0.5, 0.8, "adj = c(0, .5)", adj = c(0, .5))
text(0.5, 0.2, "adj = c(1, .5)", adj = c(1, .5))
text(0.2, 0.5, "adj = c(1, 1)", adj = c(1, 1))
text(0.8, 0.5, "adj = c(0, 0)", adj = c(0, 0))
```


adj = c(0, .5)
srt = 45, adj = c(.5, .5)
adj = c(1, 1)
adj = c(0, 0)
adj = c(1, .5)
0.0
0.2
0.4
0.6
0.8
1.0
0.5
0.5

```r
# adding a legend
plot(mtcars$mpg, mtcars$hp)
legend("topright", legend = "A legend")
```



Donation

If you find any value and usefulness in this set of slides, please consider making a one-time donation in any amount (via paypal). Your support really matters.

Donate

https://www.paypal.com/donate?business=ZF6U7K5MW25W2&currency_code=USD