

Graphic visualization of data

just scratch the surface

The graphic page

- Graphic functions add drawing components to the graphic page
- main functions (non exhaustive)
 - **plot**
 - **hist**
 - **barplot**
 - **boxplot**
 - **pie**
- additional components (non exhaustive)
 - **lines**
 - **abline**
 - **text**
 - **axis**

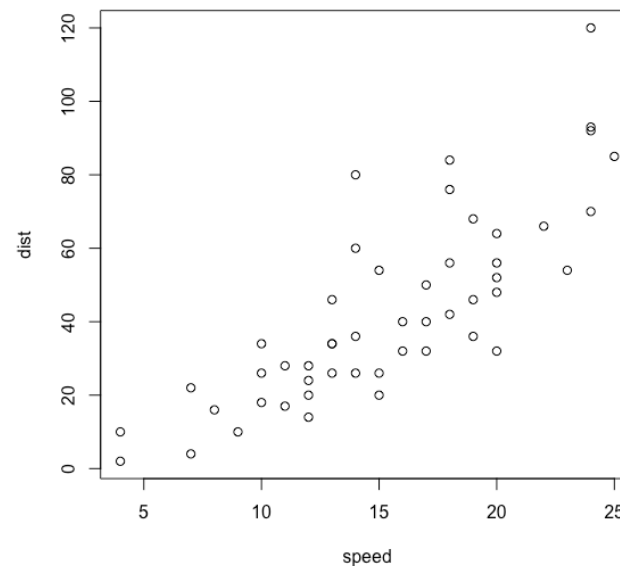
In the following, with “plot” in standard character we will mean a generic graphic function, while with **plot** we will mean the specific function with that name

Standard components and default values

- plot functions have a set of standard components and default values
 - in many cases default values are drawn from data, e.g. column names in data frames
- graphic appearance can be highly personalised
- by default, if the parameter is a two-columns dataframe we obtain a scatter plot of the columns

```
# plot data from the predefined dataframe cars  
# a dataframe with two columns (dist,speed)  
# Speed and Stopping Distances of Cars \(ethz.ch\)
```

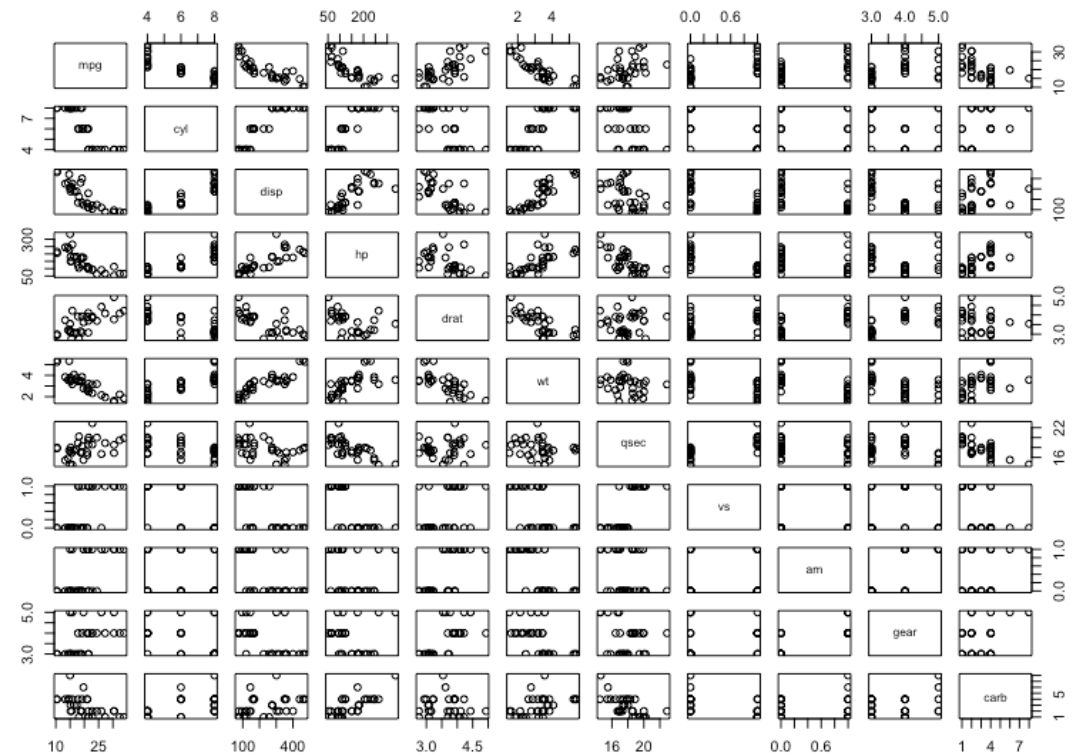
```
plot(cars)
```



Standard components and default values

```
# plot data from the  
# predefined  
# dataframe mtcars  
plot(mtcars)
```

- by default, if the parameter is a many-columns dataframe we obtain the *pair-plot*
 - scatter plot of all the pairs
- [mtcars](#)

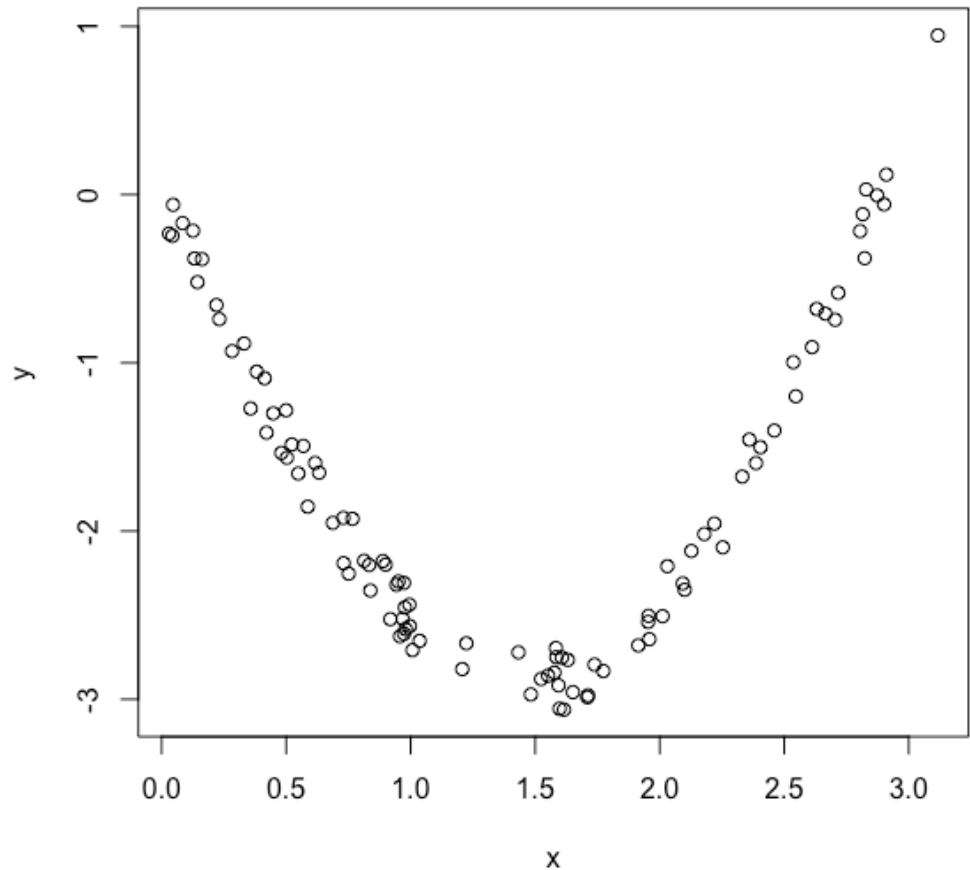


Standard components (non-exhaustive)

- x label, ticks, values, ... (same for y)
- types
 - “p”: Points (default)
 - “l”: Lines
 - “b”: Both
 - “c”: The lines part alone of “b”
 - “o”: Both “overplotted”
 - “h”: **Histogram** like (or high-density) vertical lines.
 - “n”: No plotting.

Plot output type

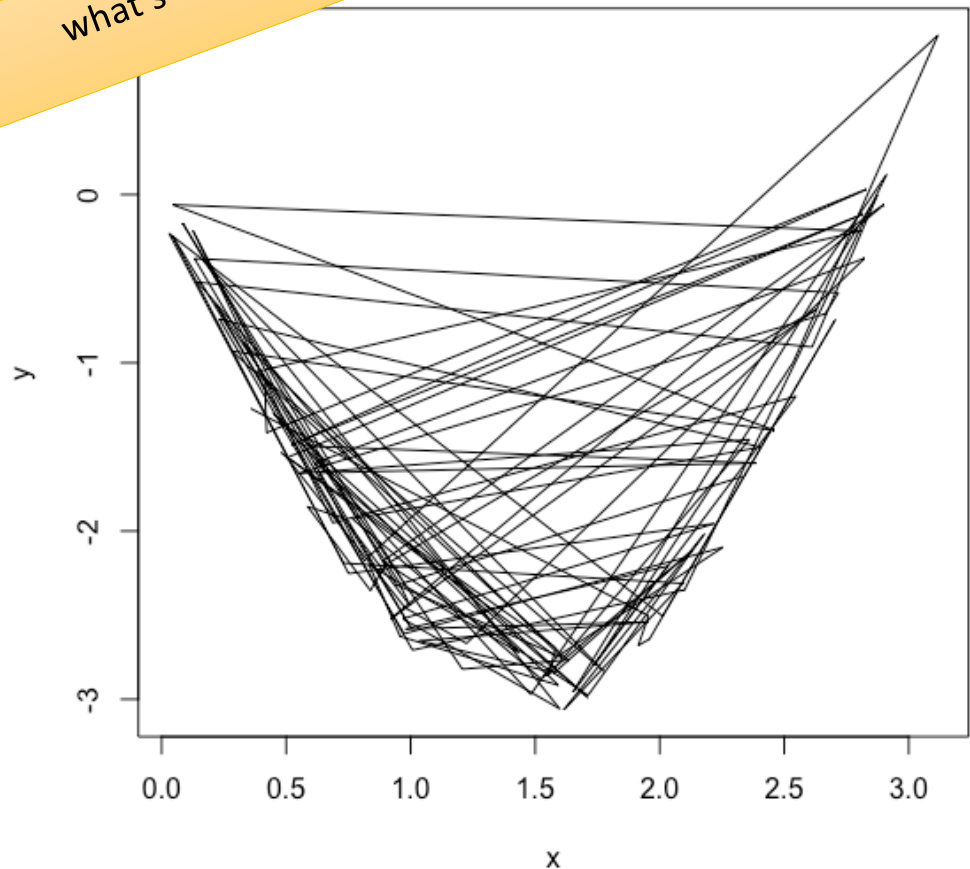
```
rm(list = ls())  
fun <- function(x){  
  noise <- 0.2  
  y <- 0.1*x^2-pi*sin(x)  
    + noise * runif(n,-1,1)  
}  
n <- 100  
seed <- 1234  
set.seed(seed)  
x <- runif(n,0,pi)  
y <- fun(x)  
plot(x,y)
```



Plot output type

```
rm(list = ls())  
fun <- function(x){  
  noise <- 0.2  
  y <- 0.1*x^2-pi*sin(x)  
    + noise * runif(n,-1,1)  
}  
n <- 100  
seed <- 1234  
set.seed(seed)  
x <- runif(n,0,pi)  
y <- fun(x)  
plot(x,y,type = "l")
```

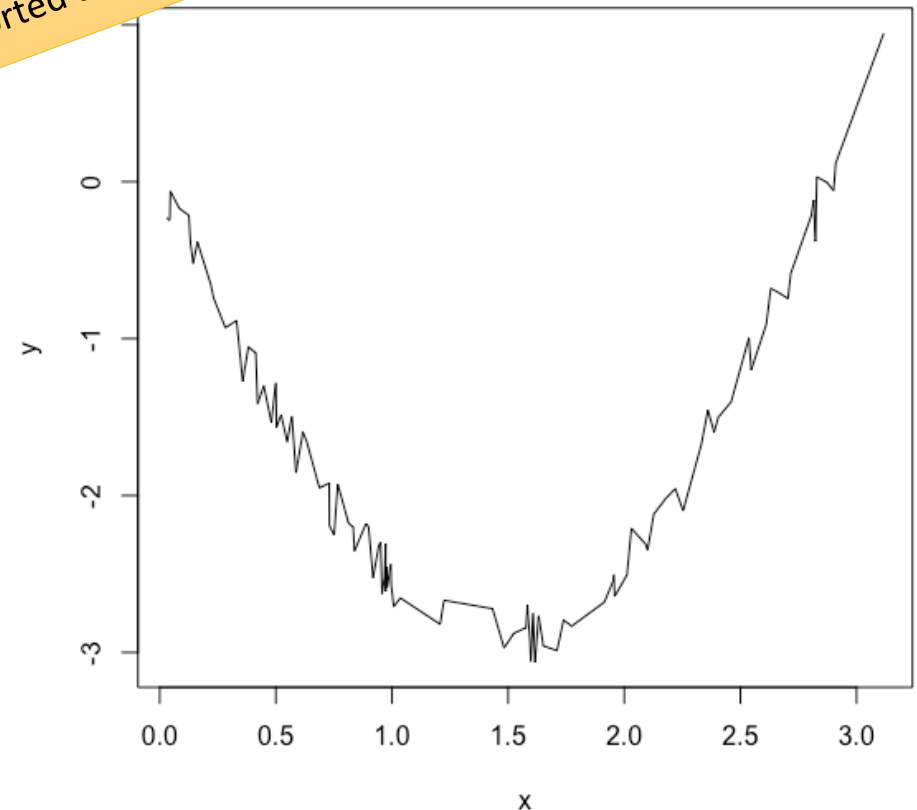
what's wrong?



Plot output type

plotting lines requires the
x data to be **sorted**
y data must be sorted accordingly

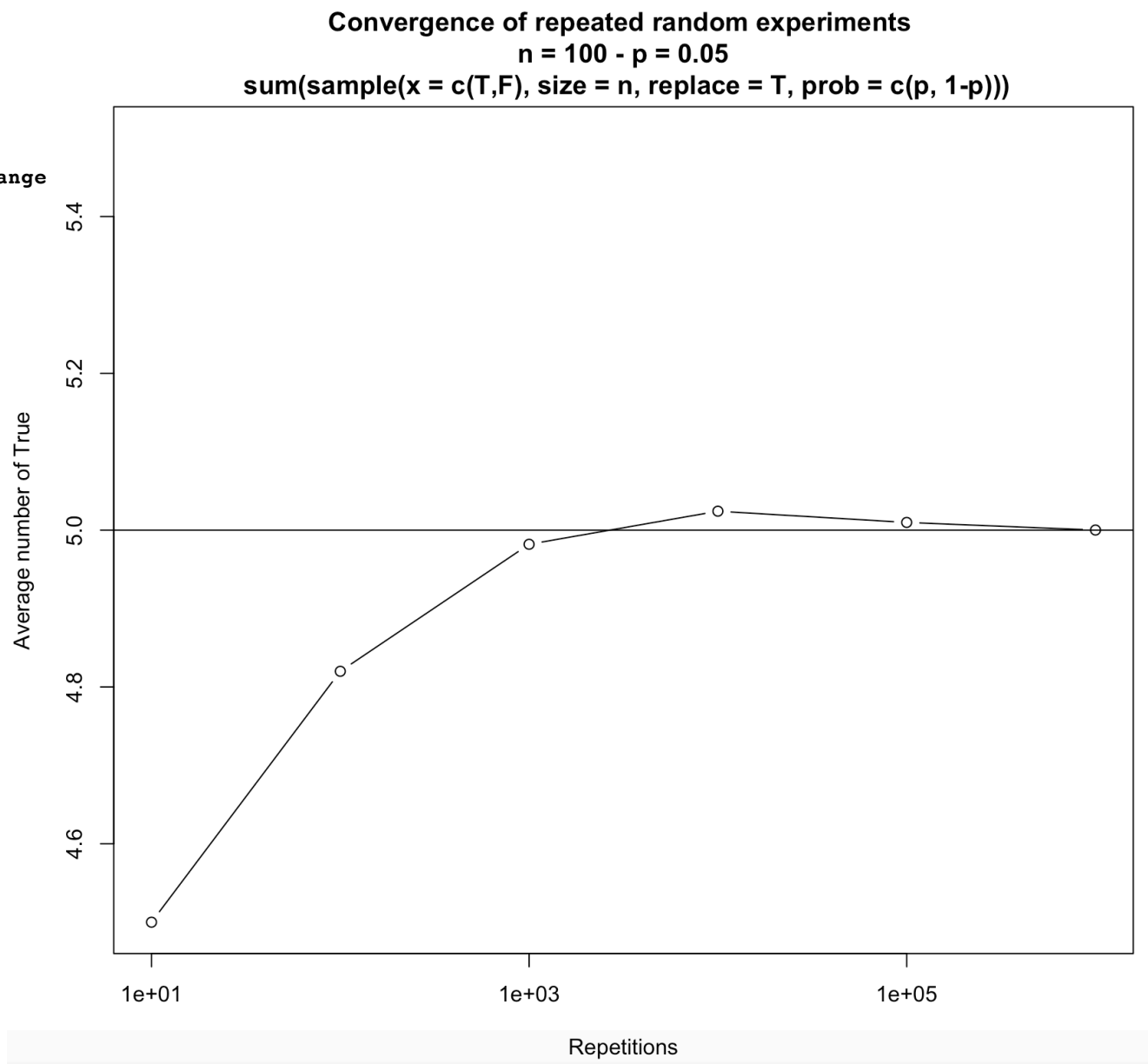
```
...  
x <- runif(n,0,pi)  
y <- fun(x)  
o_x<- order(x)  
    # the index sequence that makes x  
sorted  
x <- x[o_x]  
    # reuse the same variables  
y <- y[o_x]  
plot(x,y, type = "l")
```




```

plot(x = 10^plotRange, y = avgT, type = "b",
     main = mainTitle,
     xlab = "Repetitions",
     log = "x",
     ylab = "Average number of True",
     ylim = c(n*p*0.9,n*p*1.1) # adjust the y label range
)
abline(h = as.integer((n*p))) # put a horizontal line

```



```
hist(outcome
      , main = histTitle
      , xlab = "Outcome"
    )
```

n = 100 - p = 0.05 - repetitions = 1e+06
sum(sample(x = c(T,F), size = n, replace = T, prob = c(p, 1-p)))

