Simulation and parallelization in R

Author: Nicholas G Reich, Andrea Foulkes, Gregory Matthews

*This material is part of the **statsTeachR** project*

Module learning goals

At the end of this module you should be able to...

- Simulate data from a parametric distribution.
- Formulate a statistical model and simulate data from it.
- Design and implement a simulation experiment to test a hypothesis.
- Run simulations in parallel, when appropriate.

What is simulation?

Definitions

- Broadly: "The technique of imitating the behaviour of some situation or process (whether economic, military, mechanical, etc.) by means of a suitably analogous situation or apparatus, esp. for the purpose of study or personnel training." (from the *OED*)
- In science: Creating a model that imitates a physical or biological process.
- In statistics: The generation of data from a model using rules of probability.

What simulations have you run?

- Drawing pseudo-random numbers from a probability distribution (e.g. proposal distributions, ...).
- Generating data from a specified model (e.g. building a template dataset, calculating statistical power).
- Resampling existing data (e.g. permutation, bootstrap).

In the right setting, any of the above methods can be used in a data analysis.

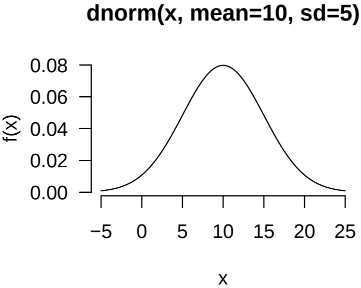
Random number generation in R

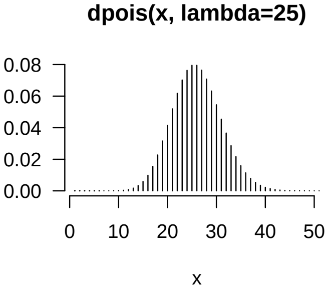
`rnorm()`, `rpois()`, etc...

Built-in functions for simulating from parametric distributions.

```r
y <- rnorm(100, mean = 10, sd = 5)
(p <- rpois(5, lambda = 25))

## [1] 24 16 19 32 26
```





Resampling data in R

`sample()`
Base R function for sampling data (with or without replacement).

```
p

## [1] 24 16 19 32 26

sample(p, replace = FALSE)

## [1] 26 19 16 32 24

sample(p, replace = TRUE)

## [1] 16 16 32 26 26
```

Generating data from a model

A Simple Linear Regression model

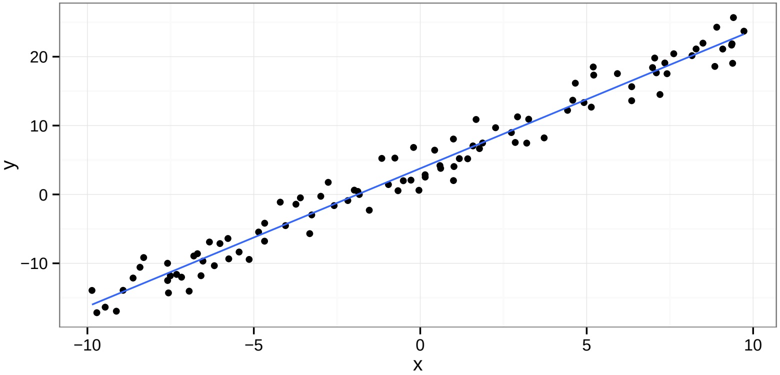
$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

What is needed to simulate data (i.e. Y_i) from this model?

- The X_i: fixed quantities.
- Error distribution: e.g. $\epsilon_i \overset{iid}{\sim} N(0, \sigma^2)$.
- Values for parameters: β_0, β_1, σ^2.

Generating data from $Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$

```
require(ggplot2)
n <- 100; b0=4; b1=2; s=2      ## define parameters
x <- runif(n, -10, 10)         ## fix the X's
eps <- rnorm(n, sd=s)          ## simulate the e_i's
y <- b0 + b1*x + eps           ## compute the y_i's
qplot(x, y, geom=c("point", "smooth"), method="lm", se=FALSE)
```



Example data: heights of mothers and daughters

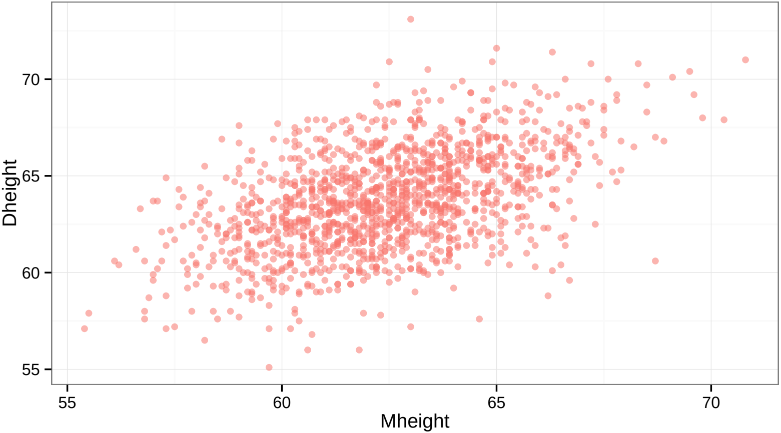
Heights of $n = 1375$ mothers in the UK under the age of 65 and one of their adult daughters over the age of 18 (collected and organized during the period 1893–1898 by the famous statistician Karl Pearson)

```
require(alr3)
data(heights)
head(heights)

##   Mheight Dheight
## 1    59.7    55.1
## 2    58.2    56.5
## 3    60.6    56.0
## 4    60.7    56.8
## 5    61.8    56.0
## 6    55.5    57.9
```

Example data: heights of mothers and daughters

```
qplot(Mheight, Dheight, data=heights, col="red", alpha=.5) +
     theme(legend.position="none")
```



One way to draw inference about height association

Using normal approximations and simple linear regression

$$Dheight_i = \beta_0 + \beta_1 \cdot Mheight_i + \epsilon_i$$

```
mod1 <- lm(Dheight ~ Mheight, data = heights)
summary(mod1)$coefficients

##             Estimate Std. Error t value  Pr(>|t|)
## (Intercept)  29.9174    1.62247   18.44 5.212e-68
## Mheight       0.5417    0.02596   20.87 3.217e-84
```

Another way to draw inference about height association

Using a simulation-based permutation test

- This can evaluate evidence for/against a null hypothesis.
- We are interested in $H_0 : \beta_1 = 0$ i.e. there is no relationship between heights of mother and daughter.
- The trick: we can easily simulate multiple sets of data that we know have no association!
- All we need is `sample()`.

```r
resampDheight <- sample(heights$Dheight, replace = FALSE)
```

Single permutation results

We can then fit this model

$$Dheight_i = \beta_0 + \beta_1 \cdot Mheight_i + \epsilon_i$$

to data from this model

$$Dheight_i = \beta_0 + \epsilon_i$$

```
mod2 <- lm(resampDheight ~ Mheight, data = heights)
summary(mod2)$coefficients

##             Estimate Std. Error t value   Pr(>|t|)
## (Intercept) 64.69803    1.86190  34.748 2.395e-190
## Mheight     -0.01516    0.02979  -0.509  6.109e-01
```

Permutation tests require repeated samples!

A permutation test algorithm

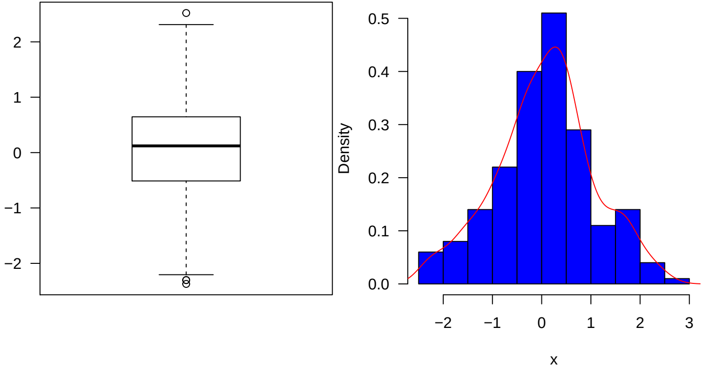
- Run original analysis (i.e. fit our linear model), store $\hat{\beta}_1$.
- For i in $1, 2, \ldots, N$:
    - Permute the Ys.
    - Re-run original analysis, store $\hat{\beta}_1^{(i)}$.
- Calculate fraction of the $\hat{\beta}_1^{(i)}$ as or more extreme than $\hat{\beta}_1$

- generate new Y_i from model without β_1. Either de novo or or from a bootstrap/permutation?
- fit model with β_1
- answer questions: how often do CIs for β_1 not cover zero (or p-val$<$.05)? fit model to real data and compare $\hat{\beta}_1$ to simulated $\hat{\beta}_1$s?

Second Test

Text is nice but let's see what happens if we make a couple of plots in our chunk:

```r
x <- rnorm(200)
par(las = 1, mar = c(4, 4, 0.1, 0.1))  # tick labels direction
boxplot(x)
hist(x, main = "", col = "blue", probability = TRUE)
lines(density(x), col = "red")
```


Density
x