Simulations

In this notebook we are going to discuss the methods for simulation that can be conducted in R. Simulations are scenario’s, by which to test the sensitivity of a system. More simply put, understanding a system, sometimes requires to run several scenario’s. Doing this is in methodological sound way, is referred to as a set of simulations.

Firstly, we need to delve into several basic concepts. The exercises that are proposed here, always relate to understanding of the total system. This means that you wil get exercises that provide key metrics and describe the entire system. Metrics will involve attributes like a mean, sample-size and/or the type of system (Normal-distribution, Poission-distribution and many more). These atributes will be used to initiate a random-number generator (random to these attributes that is) to simulate circumstances. A list of a few of these generators are shown below:

(d,p,q,r)beta(shape1, shape2, ncp)

(d,p,q,r)binom(size, prob)

(d,p,q,r)chisq(df, ncp)

(d,p,q,r)exp(rate)

(d,p,q,r)gamma(shape, scale)

(d,p,q,r)logis(location, scale)

(d,p,q,r)norm(mean, sd)

(d,p,q,r)pois(lambda)

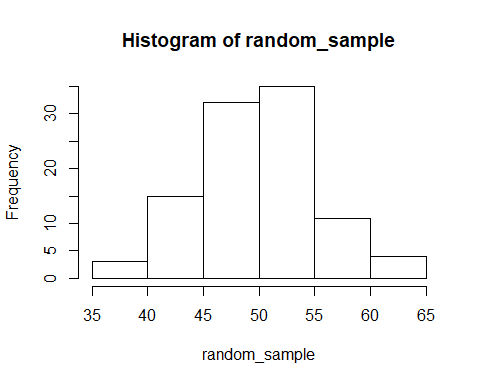
(d,p,q,r)t(df, ncp)

(d,p,q,r)unif(min, max)

Kinda Random, but thats ok

In field of research that you are investigating especially the norm() and pois() are going to be improtant. They represent the normal-distribution and the Poisson-distribution. Please read up in the manual which flavour (d,p,q,r) goes into what by typing and typing e.g. “?rnorm” or “pnorm”. The different types actually enable you to directly impute sample-size and the previously mentioned attributes. An example of a few random samples are shown below:

# Binomial distribution  
  
  
sample\_size = 100  
  
random\_sample=rbinom(100, 100, 0.5)  
  
hist(random\_sample)



# Please re-run the code to see how it differs throughout.   
  
quantile(random\_sample,c(0.25,0.75)) # 50% interval

## 25% 75%   
## 46.75 53.00

quantile(random\_sample, c(0.025, 0.975)) # 95% interval

## 2.5% 97.5%   
## 39.95 62.05

Note that the quartile() function provides confidence intervals at certain levels.

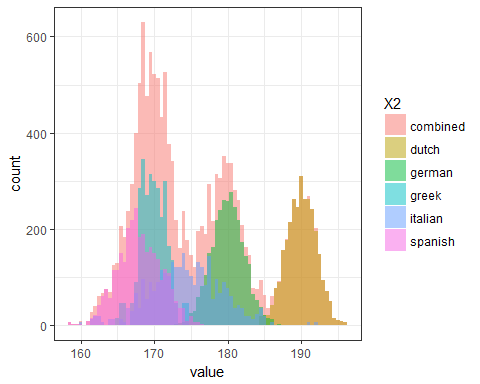
Cool stuff

To end this basic introduction into simulations, we are going to stack 5 simulations into one graph. When doing this we should expect different distributions throughout. We assume that we have several distributions of population heights. We have 2000 Dutch, 3000 German, 1000 Spanish, 500 Italian and 200 Greek students for a certain course at the SBE. What is the most likely average length? How is this distributed? What range will most certainly the average length be in (confidence interval 95%)?

library(ggplot2)  
library(reshape)  
  
# Creating the samples  
dutch <- rnorm(2000, 190, 2)  
german <- rnorm(3000, 180, 2.2)  
spanish <- rnorm (1000, 168, 3)  
italian <- rnorm(500, 174, 5)  
greek <- rnorm(200, 170, 2)  
  
# Data manipulation  
combined <- cbind(dutch, german, spanish, italian, greek)

## Warning in cbind(dutch, german, spanish, italian, greek): number of rows of  
## result is not a multiple of vector length (arg 1)

df <- melt(combined)  
df2 <- rbind(as.matrix(df), (cbind(rep(0, nrow(df)), rep("combined", nrow(df)), df[,3])))  
df3 <- as.data.frame(df2, stringsAsFactors=FALSE)  
df3$value <- as.numeric(df3$value)  
  
# plotting the plot  
ggplot(df3,aes(x=value,group=X2,fill=X2)) + geom\_histogram(position="identity",binwidth=0.5, alpha=0.5) + theme\_bw()



Remeber that the quaritle() function can also be run of the aggregated dataset, in this case to get the confidence intervals that you need. The code of how to run this is provided below. Please note that the “combined” object is a matrix and therefore deals with the entire object as if it where a singular vector, ignoring any column like structures.

quantile(combined,c(0.25,0.75)) # 50% interval

## 25% 75%   
## 169.4063 181.9097

quantile(combined, c(0.025, 0.975)) # 95% interval

## 2.5% 97.5%   
## 164.2899 192.3239