# Computer lab 1: Random Variable Generation and Monte Carlo Integration

## Learning objectives

The main objective of this computer lab is to make the student familiar with numerical random variable generation and Monte Carlo integration in R.

After completing the lab the student shall be able to:

1. Implement and evaluate some of the most common algorithms for random variable generation in R.
2. Use exact and approximate Monte Carlo simulation techniques for numerical integration in R.

## Recommended reading

Chapter 2-3 in Robert and Casella (2009)

Chapter 6 in Givens and Hoeting (2013)

## Assignment 1: Random variable generation

a). Implement R-code of the Box-Müller algorithm for random generation of variable X from a normal distribution. Table 6.1 in Givens and Hoeting (2013) provides information on the algorithm. Compare the results with simulations from rnorm() in terms of summary statistics (mean and standard deviation) and a histogram based on *n* = 100 000 draws.

b). Make a for loop (or something equivalent) and produce 100 replicate data sets of the set-up in the earlier part of the exercise. Use the mean to calculate expectations for each of these, and finally calculate the standard deviation of the means. Repeat this procedure with *n* = 100, 1 000 and 10 000. Present a table with *n* and the standard deviation of the means. Discuss your result and present the R-code you together with explaination # comments.

## Assignment 2: Monte Carlo simulation

a). The purpose of this part is to test the variance reduction properties of the importance sampling algorithm. The logic behind importance sampling is to introduce an importance sampling function into the integral

where are importance weights. Consider the function . The first task is to write R code that implements ordinary Monte Carlo sampling to obtain where . Use the MC samples to obtain the expectation, standard deviation and coefficient of variation based on 10 000 samples. Plot against .

#Add: Compare with the integrate() function in R.

You will now introduce an importance sampling function that together with results in . The full integral can be re-written as

The following R-code can be used for importance sampling:

w <- function(x) dunif(x, 0, 10)/dnorm(x, mean=5, sd=1)

f <- function(x) 10\*exp(-2\*abs(x-5))

X <- rnorm(10000,mean=5,sd=1)

Y <- w(X)\*f(X)

Explain what each part of the code performs, run it and compare the results with the earlier results obtained without IS.

b). The next task is to transform the IS algorithm into a Sampling importance resampling (SIR) algorithm following the lecture notes (and course literature) and compare the results with these from a). Use the same *n* and choose *m* according to recommendations. Discuss the results and provide code with explanations.

## To hand in

A written report (Word, pdf, html) where you summarize your main findings in the assignments. Submit your report via Moodle before the deadline.