

TMA4300 Computer Intensive Statistical Methods

- ▶ Home page: <https://wiki.math.ntnu.no/tma4300/2018v/>
- ▶ Lecturer: Håkon Tjelmeland
- ▶ Teaching assistants: Ingeborg Hem and Maxime Conjard

Course outline

- ▶ The course is divided in three parts:

Part 1: Algorithms for stochastic simulation.

Part 2: Markov chains Monte Carlo methods and Bayesian inference

Part 3: Bootstrap and the expectation-maximisation algorithm

- ▶ Each part consists of:
 - ▶ Plenary lectures and interactive lectures
 - ▶ Exercises (obligatory)
 - ▶ Oral presentations (obligatory)

TMA4300: Learning outcome, Knowledge

- ▶ The student knows computational intensive methods for doing statistical inference.
- ▶ This includes direct and iterative Monte Carlo simulations, as well as the expectation-maximisation algorithm and the bootstrap.
- ▶ The student has basic knowledge in how hierarchical Bayesian models can be used to formulate and solve complex statistical problems.

TMA4300: Learning outcome, Skills

- ▶ The student can apply computational methods, such as Monte Carlo simulations, the expectation-maximisation algorithm and the bootstrap, on simple applied problems.
- ▶ General competence. The student is able to give an oral presentation where he or she communicate his or her findings in a project.

Do you have experience with ...

- ▶ Markov chains?
- ▶ The computer language R?
- ▶ Basic Bayesian inference?

Access to the computer lab

- ▶ For those who do not have access to Nullrommet 380A: please send Ingeborg Hem (ingeborg.hem@ntnu.no) a mail as soon as possible with your name, NTNU username/e-mail address and study programme.
- ▶ You will get:
 - ▶ Physical access to the computer lab
 - ▶ User at the (linux) computers
- ▶ Don't wait, do it today!

Partner for the exercises

- ▶ The exercises should be done in groups of two persons. You can
 - ▶ let the teaching assistants find exercise partner for you. If so, send an email with your name and email address to Maxime Conjard (maxime.conjard@ntnu.no) and ask for this.
 - ▶ find an exercise partner yourself. If so, send an email with (the two) names and email addresses to the teaching assistant (maxime.conjard@ntnu.no).
- ▶ Don't wait, do it today. Deadline: Tuesday January 16th.

Reference group

- ▶ Three students should be in the reference group.
 - ▶ Preferably from different study programmes.
- ▶ Duties:
 - ▶ Stay in dialogue with the other students.
 - ▶ Participate in three meetings with the lecturer and the teaching assistants, and provide suggestions for improvement in exercises, lectures, home page, and so on.
 - ▶ Write a joint brief final report providing constructive feedback and evaluation of the course. This report will be published unedited in the course evaluation.
- ▶ Volunteers?

The word simulation ...

... refers to the treatment of a real problem through reproduction in an environment controlled by the experimenter.

Gamerman & Lopes, Markov Chain Monte Carlo, 2nd Edition, Page 9

Motivation: Queueing problem

M/G/1 - queue:

- ▶ Customers arrive to a queueing system according to a Poisson process, i.e. interarrival times are exponentially distributed.
- ▶ One server
- ▶ Independent service times distributed according to $f(t)$.
- ▶ Queue system empty at time $t = 0$

$X(t)$ customers in queueing system at time t .

Motivation: Queueing problem

If service times are exponentially distributed, $X(t)$ is a Markov process and an explicit analytical formula for the limiting distribution is available. For general f , analytical solutions might not be available.

⇒ Idea: Simulate the queueing process on a computer and return the quality of interest, e.g. $\min\{t > 0 : X(t) \geq 7\}$.

Simulation, why do we need it?

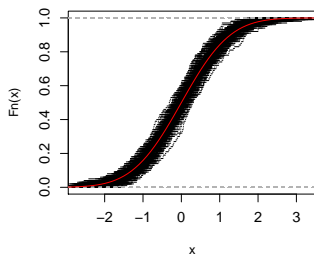
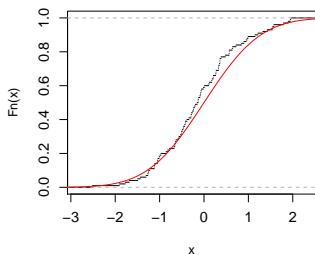
Necessity to produce chance on the computer:

- ▶ Evaluation of the behaviour of a complex system (Epidemics, weather forecast, networks, economic actions, etc)

Simulation, why do we need it?

Necessity to produce chance on the computer:

- ▶ Evaluation of the behaviour of a complex system (Epidemics, weather forecast, networks, economic actions, etc)
- ▶ Determine probabilistic properties of a novel statistical procedure or under an unknown distribution.

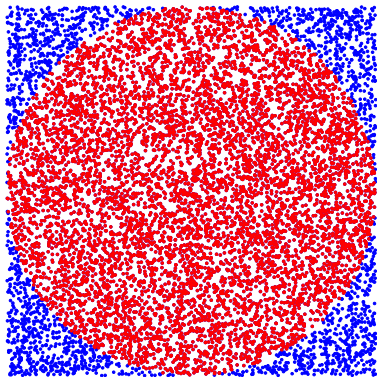


(Left: Estimation of CDF from a normal sample of 100 points,
Right: Variation of the estimation over 200 samples.)

Simulation, why do we need it?

- ▶ Approximation of an integral/area

```
n = 1000    ▷ (# of simulations)
m = 0      ▷ (# points in circle)
i = 1
while i < n do
  x = Rand(1), y = Rand(1)
  if  $x^2 + y^2 < 1$  then
    m ← m + 1
  end if
  i = i + 1
end while
return  $4 \cdot m/n$ 
```



$$\hat{\pi} = 3.1353.$$

Simulation, how do we do it?

Central building block of simulation: Always requires availability of uniform $\mathcal{U}(0, 1)$ random variables.

```
> runif(10)
[1] 0.96112670 0.83981815 0.77269526 0.03363567 0.89374420 0.25631906
[7] 0.82222815 0.77355197 0.17093594 0.12035040
```

Pseudo-random generator

A pseudo-random generator is a **deterministic function** f which takes a uniform random bit string as input and outputs a bit string which cannot be distinguished from a uniform random string.

In more detail, this means that for starting value (seed) u_0 and any n , the sequence

$$\{u_0, f(u_0), f(f(u_0)), f(f(f(u_0))), \dots, f^n(u_0)\}$$

behaves **statistically** like an $\mathcal{U}(0, 1)$ sequence (when appropriately scaled).

A standard uniform generator

The **congruential random generator** on $\{0, 1, \dots, M - 1\}$

$$f(x) = (a \cdot x + b) \pmod{M}$$

has a period equal to M for proper choices (a, b) and becomes a generator on $[0, 1)$ when dividing by M .

Example

Take

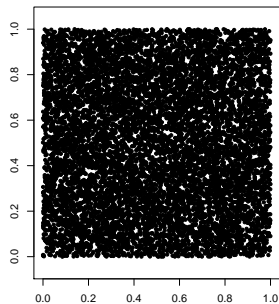
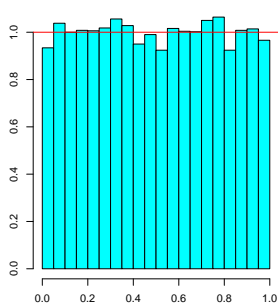
$$f(x) = (69069069 \cdot x + 12345) \bmod (2^{32})$$

and produce

..., 69081414, 2406887111, 1109307232, 2802677792, 3651430880, 806776992, ...

i.e.

..., 0.01608427, 0.56039708, 0.25828072, 0.65254927, 0.85016500, 0.18784241, ...



Random number generator

From now on, we assume to have a random generator on the unit interval available.