Problem set 7

Exercise 1. The SIR model (R and Python)

The SIR model describes the spread of an epidemic. Let S(t), I(t) and R(t) be the number of Susceptible, Infected and Recovered individuals at time t. We consider these functions as continuous quantities, even though they are necessarily discrete in practice. Suppose that each Infected individual becomes Recovered at rate β, and each Infected individual infects each Susceptible at rate α, then we have:

$$\frac{dS}{dt} = -\alpha \cdot I(t) \cdot S(t)$$
$$\frac{dI}{dt} = \alpha \cdot I(t) \cdot S(t) - \beta \cdot I(t)$$
$$\frac{dR}{dt} = \beta \cdot I(t)$$

By looking at I’(0), give a condition for the epidemic to die out quickly, in terms of α, and S(0). Test your condition by solving the system of equations numerically, for a variety of initial conditions. Write your code in R and Python.

Exercise 2. The Lanchester combat (R and Python)

The Lanchester combat model was first used to study air combat during World War I. Let A(t) be the number of planes on the side A and B(t) the number of planes on the other side B. We treat A(t) and B(t) as continuous differentiable functions, even though they are discrete.

a) The basic Lanchester model supposes that the rate of decrease in A at any given time is proportional to B, and that the rate of decrease in B is proportional to A. Express the dynamics of A and B as a system of ordinary differential equations (ODEs).
b) Suppose now that side A has A0 reinforcements, that arrive at rate r1 (until they are used up), and that side B has B0 reinforcements, that arrive at rate r2. In addition, suppose that individual aircraft breakdown at a rate b1 for side A and b2 for side B. Incorporate these additional components into your model.

c) Write a program to simulate your model in both R and Python. Consider appropriate parameter values and plot the solutions.

Exercise 3. Normal distribution

A man travels to work by train and by bus. His train is due to arrive at 08:45 and the bus he hopes to catch is due to leave at 08:48. The time of arrival of the train has a normal distribution with the mean 08:44 and standard deviation of three minutes; the departure time of the bus is independently normally distributed with the mean 08:50 and standard deviation of one minute. Calculate the probabilities (in R and Python) and make your predictions for:

- The train is late;
- The bus departs before the train arrives.

Exercise 4. Pseudo random number generator in Python

Consider the following pseudo random number generator, which generates a new "random" number X_{i+1}, from the previous number X_i, as:

$$X_{i+1} = \mathrm{mod}\left(a \cdot X_i + b, M\right)$$

with 'mod' the modulo operation, and *a* and *M* some constants (which should be very large numbers).

This generator will give numbers within the range (0, *M*-1). To obtain random numbers within the unit interval $I_i \in [0,1]$, we set

$$I_i = \frac{X_i}{M}$$

Which of the following codes correctly implement this random number generator to compute 'ntotal' numbers I_i with an initial seed X_0?

A

```
import numpy as np
def rng(seed, a, b, M, ntotal):
    data = np.zeros(ntotal)
    data[0] = seed
    for i in range(1,ntotal):
        data[i] = np.mod((a*data[i-1]+b), M)
    return data/np.float(M)
```

B

```
import numpy as np
def rng(seed, a, b, M, ntotal):
    data = np.zeros(ntotal)
    data[0] = seed
    for i in range(1,ntotal):
        data[i] = np.mod((a*data[i-1]+b), M)/np.float(M)
    return data
```

C

```
import numpy as np
def rng(seed, a, b, M, ntotal):
    data = np.zeros(ntotal)
    for i in range(1,ntotal):
        data[i] = np.mod((a*data[i-1]+b), M)
    return data
```

D

```
import numpy as np
def rng(seed, a, b, M, ntotal):
    data = np.zeros(ntotal)
    data[0] = seed
    for i in range(0,ntotal):
        data[i+1] = np.mod((a*data[i]+b), M)
    return data/np.float(M)
```

Exercise 5. Distribution for a random variable

You have a distribution for a random variable that has a bell shape like that of a Gaussian. Looking at the distribution near the tails, you notice that 0.01 % of the points are more than 5σ (standard deviation) away from the average. Can you say that the distribution is Gaussian?

Write a program to simulate 10^5 numbers for a normal distribution N(0,1) and calculate the amount of points lying more than 5 away from 0.