Statistical Methods

Yiannis Tsapras

Exercise 4 for **August 8**, 2024, 18:00

Correlation, Monte Carlo integration

4.1 Correlations of performance in physics exams and exercises at Heidelberg University

In the zip-file *data_klausuren.zip* (on the web) you find results of eight different physics exams and exercises of recent semesters. The entries in the two columns in each file give the result for a particular student. From the names of the files you can guess which exam or exercise is meant. For these data set:

a: Read the data in (`read.table()` is your friend) and create scatter plots of one result against the other. Take care of a reasonable scaling of the axes.

b: Calculate the (Pearson's product moment) correlation coefficient between each of the two results.

c: What do you think should be done in cases with zero points? Zero points are often "artificial" in the sense that a student strike everything out in order not to pass an exam. Hint: you may replace zeros with the special value `NA` of `R` using Boolean masks, and than use the `use=` argument of `cor()`.

d: What conclusions do you draw from the results above? Anything that appears remarkable to you?

4.2 Performance of Monte Carlo integration in different dimensions

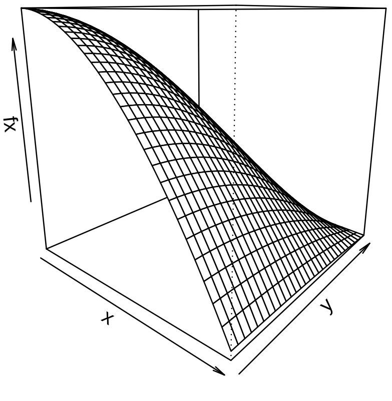
We would like to compare the performance of the Monte Carlo integration technique with the regular midpoint method. To this end, consider the integral

$$I = \int_V f(\mathbf{x})\,\mathrm{d}^d\mathbf{x}\,, \tag{1}$$

where the integration domain V is a d-dimensional hypercube with $0 \le x_i \le 1$ for each component of the vector $\mathbf{x} = (x_1, x_2, \ldots, x_d)$. The function we want to integrate is given by

$$f(\mathbf{x}) = \prod_{i=1}^{d} \frac{3}{2}(1 - x_i^2)\,. \tag{2}$$

In 2D, f looks like



The function f has an analytic solution of course, which is $\langle f \rangle = I = 1$ independent of d and $\left\langle f^2 \right\rangle = (6/5)^d$, but we want to ignore this for the moment and use the problem as a test of the relative performance of Monte Carlo integration and ordinary integration techniques. To this end, calculate the integral in dimensions $d = 1, 2, 3, \ldots, 10$, using

a: the midpoint method, where you divide the volume into a set of much smaller hypercubes obtained by subdividing each axis into n intervals, and where you approximate the integral by evaluating the function at the centers of the small cubes.
Hint: On the web you find the auxiliary function `xmidpoints.R` which may help you to solve the problem. Be aware that one can very easily fill up the whole computer memory with it. So, handle the number of dimensions with care! Alternatively, think about a way of splitting the integration into smaller pieces.

b: standard Monte Carlo integration in d dimensions, using N random vectors. At which dimension starts to outperform Monte Carlo integration the midpoint procedure?

For definiteness, adopt $n = 6$ and $N = 20000$. For both of the methods, report the numerical result for I, estimated uncertainty, and the CPU-time needed for each of the dimensions $d = 1, 2, \ldots, 10$. (If you manage, you can also go to higher dimension. If you run into memory problems stop at dimension smaller than 10.)