Numerical Prerequisites

THE NUMERICAL EXERCISES SERVE NO OTHER PURPOSE THAN GETTING USED TO RANDOMNESS. FOR THIS, WE NECESSARILY DEPEND ON RANDOM NUMBER GENERATORS, AND HENCE SOFTWARE. ALL NUMERICAL EXERCISES WILL BE STRIPPED OFF OTHER NUMERICAL OVERHEAD (FANCY FUNCTIONS, FANCY LIBRARIES, FANCY SYNTAX).

The classes are likely to use python-notebooks, and/or C/C++ with the GSL. Students of this class are advised to familiarize themselves with the following basic programming elements.

1. For-loops.

2. If-statements.

3. Basic maths: addition, multiplication, powers, logarithm, exponential function, sine, cosine.

4. Simple functions, which take input values (parameters, e.g. a, b, x), and return a function value, e.g. $f(a, b, x) = a + bx$.

5. Calling a random number generator.

6. Storing random numbers in a vector.

7. Basic linear algebra: dot-products between vectors, products of matrices and vectors.

8. Plotting: a function; a 1-d histogram; a 2-d histogram.

9. Input and output: Reading and writing text files with columns (ASCII).

The above are recurrent elements in Master's theses, PhD-theses, and academic research in general. Familiarity with these elements is hence of advantage beyond this lecture. If you are fully new to programming, it is advised that you spend a few afternoons studying basic python-tutorials: What is python, what are python-notebooks, how do I use scipy and numpy? Google and Youtube are your friends (in this case). The tutors will support you with more advanced functions during the exercise classes.

IF YOU ARE A C/C++ PROGRAMMER and you wish to change to python notebooks for these exercises, I think you are likely to miss

VECTOR<DOUBLE> X;

To soothe your heartache, try numpy's zeros

NP.ZEROS(100)