Prof. Dr. Volker Roth
volker.roth@unibas.ch
6^{th} floor

Sebastian Keller
sebastianmathias.keller@unibas.ch
1^{st} floor

Departement Mathematik und Informatik
Spiegelgasse 1
4051 Basel

University of Basel

Exercise 3

Solutions for exercises marked with a '$\star$' will be made available online, usually in the following week. Only if needed will these exercises be discussed during the tutorial. All other exercises are to be prepared at home and presented by the participants during the tutorial session.

3.1 LU Decomposition

In this exercise, we will decompose matrix A into the product $L \cdot U$. To do this, we apply the appropriate transformation matrices, i.e. elementary R-matrices, multiple times to the matrix A (see Chapter 1: Linear Systems of Equations). The output will then be the lower and upper triangular matrices L and U. In Python, implement this algorithm and apply it onto the following matrix A:

$$A = \begin{pmatrix} 1 & 0 & -5 \\ -6 & 2 & 9 \\ 2 & -3 & 2 \end{pmatrix}$$

3.2 Back- and Forward Substitution: LU Decomposition for solving $A\boldsymbol{x} = \boldsymbol{b}$

The decomposition $A = LU$ from above is used to solve a linear system $A\boldsymbol{x} = \boldsymbol{b}$. In particular, this is a two-step process which requires forward and backward substitution. In the tutorial we already worked out how to do backward substitution, i.e. solving $U\boldsymbol{x} = \boldsymbol{c}$. From this result it is very easy to find an expression for the forward substitution to solve systems of the form $L\boldsymbol{x} = \boldsymbol{c}$. Write a Python script that uses back- and forward substitution and the LU decomposition of the matrix A calculated in the previous exercise to solve $A\boldsymbol{x} = LU\boldsymbol{x} = \boldsymbol{b}$ with

$$b = \begin{pmatrix} -23 \\ 54 \\ 2 \end{pmatrix}$$

What is the advantage of using an LU decomposition over Gauss-Jordan elimination?

3.3 Cholesky Decomposition*

Finally, for symmetric positive-definite matrices, we can apply the Cholesky decomposition. In our case, matrix

$$A = \begin{pmatrix} 16 & -4 & -12 \\ -4 & 5 & 5 \\ -12 & 5 & 19 \end{pmatrix}$$

shall be written as LL^t, where

$$L = \begin{pmatrix} \ell_{11} & 0 & 0 \\ \ell_{21} & \ell_{22} & 0 \\ \ell_{31} & \ell_{32} & \ell_{33} \end{pmatrix}.$$

It should be clear that the Cholesky decomposition is a special case of the LU decomposition which is computational *less* demanding (why?) than LU decomposition. In Python, implement the Cholesky decomposition either by modifying the code you wrote in exercise 3.2 or by implementing it from scratch and calculate the matrix L.