Compmath: Python Übung 4

Richard Weiss vs. Asst. Prof. Kevin Sturm

May 2019

Problem 4.
Let $L = (\ell_{ij})_{i,j=1,\dots,n} \in \mathbb{R}^n$ be a regular (i.e. L has full rank) and lower triangular matrix (i.e. $\ell_{ij} = 0$ for $i < j$). We write L in the block form

$$L = \begin{pmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{pmatrix}$$

with $L_{11} \in \mathbb{R}^{p\times p}$, $L_{21} \in \mathbb{R}^{q\times p}$ and $L_{22} \in \mathbb{R}^{q\times q}$, where $p + q = n$.

(a) Show that $\det(L) = \prod_{j=1}^{n} \ell_{jj}$.

Solution.
In der Summe $\sum_{\sigma\in S_n} \operatorname{sgn}\sigma \cdot \prod_{j=1}^{n} \ell_{\sigma(j)j}$ liegt höchstens dann ein Summand $\neq 0$ vor, falls $\sigma(1) = 1, \sigma(2) < 2, \dots, \sigma(n) \leq n$, also $\sigma = \mathrm{id}_{1,2,\dots,n}$.

(Havlicek, 2012, Lineare Algebra für Technische Mathematiker)

(b) Show that

$$L^{-1} = \begin{pmatrix} L_{11}^{-1} & 0 \\ -L_{22}^{-1}L_{21}L_{11}^{-1} & L_{22}^{-1} \end{pmatrix}.$$

Solution.
To verify this, we multiply the matrices block-wise:

$$L^{-1}\cdot L = \begin{pmatrix} L_{11}^{-1} & 0 \\ -L_{22}^{-1}L_{21}L_{11}^{-1} & L_{22}^{-1} \end{pmatrix} \cdot \begin{pmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{pmatrix} = \begin{pmatrix} E_p & 0 \\ 0 & E_q \end{pmatrix} = E_n.$$

Problem 7.
Let $U = (u_{ij} \in \mathbb{C}^{n\times n}$ be an upper triangular and regular matrix, i.e., $u_{jk} = 0$ for $j > k$, such that $u_{jj} \neq 0$ for all $j = 1, \dots, n$.

(a) Show that for every $b \in \mathbb{C}^n$, there exists a unique solution $x \in \mathbb{C}^n$ of $Ux = b$.

Solution.
We know that

$$Ux = \begin{pmatrix} u_{11} & \cdots & u_{1n} \\ & \ddots & \vdots \\ 0 & & u_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} \tag{1}$$

Because U is regular, all columns (as vectors) are *linearly independent*. Hence, $u_{11} \neq 0$, because the first column $u_1 \neq \vec{0}$. But then, $u_{22} \neq 0$, for elsewise, u_2 would be a multiple of u_1. But then, $u_{jj} \neq 0$, for elsewise, $u_j \in [u_1, \dots, u_{j-1}]$.

Now, we can multiply $U \cdot x$ from (1), and extract the unique solution to $x_n = \frac{b_n}{u_{nn}}$. This gives us a unique solution for x_{n-1} and even x_j.

Problem 8.
The integral $\int_a^b f dx$ of a continuous function $f : [a, b] \to \mathbb{R}$ can be approximated by so called quadrature formulas

$$\int_a^b f\, dx \approx \sum_{j=1}^{n} \omega_j f(x_j)$$

where one fixes some vector $x = (x_1, \dots, x_n) \in [a, b]^n$ with $x_1 < \dots < x_n$ and approximates the function f by some polynomial $p(x) = \sum_{j=1}^{n} a_j x^{j-1}$ of degree $\leq n - 1$ with $p(x_j) = f(x_j)$ for all $j = 1, \dots, n$. The weights ω_j are defined as the solution of

$$\int_a^b q\, dx = \sum_{j=1}^{n} \omega_j q(x_j) \text{ for all polynomials } q \text{ of degree} \leq n - 1. \tag{2}$$

(a) Show that (2) is equivalent to

$$\int_a^b x^k\, dx = \sum_{j=1}^{n} \omega_j x_j^k \text{ for all } k \in \{0, \dots, n - 1\}. \tag{3}$$

Solution.
Let $n \in \mathbb{N}$, $a, b \in \mathbb{R}$, where $a < b$ and $x_1, \dots, x_n \in [a, b]$, where $x_1 \leq \ldots \leq x_n$.
Note, that $x \neq (x_1, \dots, x_n)$.

"$\Rightarrow$" Obviously, $q(x) := x^k$ is a polynomial of degree $k \leq n - 1$.

"$\Leftarrow$" Let $q(x) = \sum_{i=0}^{k} a_i x^i$ be an arbitrary polynomial of degree $k \leq n - 1$. Applying (3) and basic integration rules, we can write

$$\int_a^b q\, dx = \int_a^b \sum_{i=0}^{k} a_i x^i\, dx = \sum_{i=0}^{k} a_i \int_a^b x^i\, dx \overset{(3)}{=} \sum_{i=0}^{k} a_i \sum_{j=1}^{n} \omega_{ij} x_j^i = \sum_{j=1}^{n} \sum_{i=0}^{k} \omega_{ij} a_i x_j^i \overset{!}{=} \sum_{j=1}^{n} \omega_j q(x_j).$$

The last equation holds true, because (3) actually means:

$$\forall n \in \mathbb{N} : \forall a, b \in \mathbb{R}, a < b : \forall x = (x_1, \dots, x_n) \in \mathbb{R}^n : \exists \omega_1, \dots \omega_n \in \mathbb{R} : \forall k = 0, \dots, n - 1 :$$
$$\int_a^b x^k\, dx = \sum_{j=1}^{n} \omega_j x_j^k$$

Hence, $\forall j = 1, \dots, n : \forall i = 0, \dots, k : \omega_{ij} = \omega_j$.

(b) Write a function `integrate` which takes the vector $x = (x_1, \dots, x_n) \in [a, b]^n$ and the function value vector $(f(x_1), \dots, f(x_n))$ and which returns the approximated value of the integral $\sum_{j=1}^{n} \omega_j f(x_j)$. Avoid loops and use appropriate vector functions and arithmetic instead.
Hint: (3) is a linear system in $(\omega_1, \dots, \omega_n) \in \mathbb{R}^n$, which you can solve with scipy.

Solution.
(Thx for the hint Prof., now we know, what (3) actually means.)

Using (3), we can write $\forall k = 0, \dots, n-1$, that

$$I_k := \frac{b^{k+1} - a^{k+1}}{k+1} = \int_a^b x^k dx = \sum_{j=1}^{n} \omega_j x_j^k = \omega_1 x_1^k + \dots + \omega_n x_n^k$$

Similarly to below, this yields

$$\underbrace{\begin{pmatrix} 1 & \cdots & 1 \\ x_1 & \cdots & x_n \\ \vdots & \ddots & \vdots \\ (x_1)^{n-1} & \cdots & (x_n)^{n-1} \end{pmatrix}}_{X:=} \begin{pmatrix} \omega_1 \\ \vdots \\ \omega_n \end{pmatrix} = \begin{pmatrix} I_0 \\ \vdots \\ I_{n-1} \end{pmatrix}$$

With these values $\omega_1, \dots, \omega_n$, we can calculate the sum $\sum_{j=1}^{n} \omega_j f(x_j)$.

We don't actually need this, but I wrote it anyhow ...
Let $p(x) = \sum_{j=1}^{n} a_j x^{j-1}$ be a polynomial of degree $\leq n-1$ and $\forall j = 1, \dots, n : p(x_j) = f(x_j)$.

$$\begin{matrix} a_1 x_1^0 + \dots + a_n x_1^{n-1} = f(x_1) \\ \vdots \\ a_1 x_n^0 + \dots + a_n x_n^{n-1} = f(x_n) \end{matrix} \quad \Rightarrow \quad \underbrace{\begin{pmatrix} 1 & x_1 & \cdots & (x_1)^{n-1} \\ 1 & x_2 & \cdots & (x_2)^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & (x_n)^{n-1} \end{pmatrix}}_{=X^T} \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} f(x_1) \\ \vdots \\ f(x_n) \end{pmatrix}.$$

This gives us the values $a_1, \dots, a_n$, of the approximation polynomial p of f.