Universität Tübingen
Mathematisches Institut
Prof. Dr. Christian Lubich

Tübingen, 07.06.2023

# 7. Exercise sheet for Algorithmen der Numerischen Mathematik

**Exercise 20:** (Frobenius norm)

Show that $\|A\|_F := (\sum_{i=1}^n \sum_{j=1}^m |a_{ij}|^2)^{1/2}$ defines a norm on the vector space of $n \times n$ matrices, for which $\|A\|_F^2 = \mathrm{trace}(A^T A)$ holds. Show further that there is no norm $\|.\|$ on the $n$-dimensional space with

$$\|A\|_F = \max_{\|v\|=1} \|Av\|.$$

**Exercise 21:** (Properties of the singular value decomposition)

Let $U^T AV = \Sigma = \mathrm{diag}(\sigma_1, \dots, \sigma_n)$ be the singular values decomposition of $A \in \mathbb{R}^{m\times n}$ with singular values $\sigma_1 \geq \dots \sigma_r > \sigma_{r+1} = \dots = \sigma_n = 0$, where $U = (u_1, \dots u_m) \in \mathbb{R}^{m\times m}$ and $V = (v_1, \dots, v_n) \in \mathbb{R}^{n\times n}$. Show:

$$\begin{aligned} A &= \sum_{i=1}^r \sigma_i u_i v_i^T \,, \\ \|A\|_2 &= \sigma_1 \,, \\ \|A\|_F^2 &= \sigma_1^2 + \dots + \sigma_r^2, \end{aligned}$$

where $\|A\|_F$ is the Frobenius norm from exercise 20. Conclude further

$$\begin{aligned} \text{Rank } A &= r \,, \\ \text{Ker } A &= < v_{r+1}, \dots, v_n > \,, \\ \text{Im } A &= < u_1, \dots, u_r > \ . \end{aligned}$$

**Exercise 22:** (Waidmanns Heil)

Perform the "Zickzack-hunt for non-zero elements" described in the lecture for

$$BQ^{(1)} = \begin{pmatrix} 1 & -1 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & -1 \end{pmatrix}.$$

**Programming exercise 6:**

Write a Matlab-Code that uses the singular value decomposition (svd) for image compressing of a black and white image. I.e. you have an image as an input and a compressed, memory reduced version of the same image as an output. Can you estimate the error in a certain norm? Write a Matlab code that computes different rank-$r$ approximations of the black and white image for $r = (50, 100, 200)$. Apply your code to the test picture you can find on the website.
Hints:

(a) You can use $[U, S, V] = \mathrm{svd}(A)$, which computes the svd of $A$ in Matlab.

(b) Use $A$ = double(rgb2gray(imread('tuebingen.jpg'))) to obtain a $m \times n$ matrix of all the black and white pixels and store it in the variable $A$.

(c) With imshow(cast($B$,'uint8')) you can plot the approximation $B$.

**Solutions are discussed on Wednesday 21.06.2023.**
Contact person: Dominik Sulz - when you have questions just come to my office (C3P16) or write me an email.