Numerical Linear Algebra - Sheet 1

to be handed in until October 25, 2023, 11am.

Please hand in your solutions in groups of two to four. The solutions for non-programming problems and the discussion of the programming problems should be handed in on paper (Mathematikon, 1st floor, letterbox 27). The code for programming problems should be handed in via Moodle by one person in your group. To recieve the points for the exercises on Müsli, write everyones first and last name on every hand-in. Details will be discussed in the exercise groups on Friday. If you currently don't have access to Moodle, please contact your tutor Laura in the exercise group (or via Müsli/Mail). If you want to get comments on your code, please also print it out and hand it in with the other solutions.

Problem 1. Review the following items and write down at least one of the definitions and one of the theorems with proof in detail.

- Definition of a projection
- Definition of an orthogonal projection
- Theorem: Orthogonal projection is uniquely determined by subspace

  Consider a finite-dimensional space V with inner product $\langle \cdot, \cdot \rangle$ and a subspace $W \subset V$. Then, there exists a unique orthogonal projection

  $$P_W : V \to W.$$

- Theorem: Best Approximation Theorem

  Let W be a subspace of $\mathbb{R}^n$, let $\mathbf{x}$ be any vector in $\mathbb{R}^n$, and let $\tilde{\mathbf{x}}$ be the orthogonal projection of $\mathbf{x}$ onto W. Then $\tilde{\mathbf{x}}$ is the closest point in W to $\mathbf{x}$, in the sense that

  $$\|\mathbf{x} - \tilde{\mathbf{x}}\| < \|\mathbf{x} - \mathbf{w}\|$$

  for all $\mathbf{w}$ in W distinct from $\mathbf{x}$.

- Theorem: Orthogonal projection in orthonormal basis

  Let $B = \{\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_p\}$ be an orthonormal basis of a subspace W of a finite-dimensional space V with inner product $\langle \cdot, \cdot \rangle$. Then, the orthogonal projection P_i of any vector $\mathbf{v} \in V$ onto $\mathbf{u}_i$, and the orthogonal projection P_W of any vector $\mathbf{v} \in V$ onto W have the following expressions, respectively:

  $$P_i(\mathbf{v}) = \langle \mathbf{v}, \mathbf{u}_i \rangle \mathbf{u}_i, \qquad i = 1, 2, ..., p,$$
  $$P_W(\mathbf{v}) = \sum_{i=1}^{p} \langle \mathbf{v}, \mathbf{u}_i \rangle \mathbf{u}_i,$$

  and

  $$\mathbf{v} = P_W(\mathbf{v}) + \mathbf{z}, \qquad \mathbf{z} \perp W.$$

- Theorem: Parseval identity

  Suppose that W is a finite-dimensional linear space with inner product $\langle \cdot, \cdot \rangle$. Let $\{\mathbf{e}_i\}$, $i = 1, ..., n$ be an orthonormal basis of W. Then, for every $\mathbf{w} \in W$ it holds

$$\sum_{i=1}^{n} |\langle \mathbf{w}, \mathbf{e}_i \rangle|^2 = \|\mathbf{w}\|^2.$$

Problem 2 (Programming)**.** Problem 1.1.4 in the Lecture Notes.

Numerical Linear Algebra - Sheet 1 Solutions

Solution 1.

Solution 2.

Solution 3.

Solution 4 (Programming)**.**

Numerical Linear Algebra - Sheet 2

to be handed in until November 01, 2022, 11am.

To hand in your code for Problem 1.2.5, please form a group on Moodle and then hand in your code as a group.

Problem 1. Problem 1.2.5 in the Lecture Notes.

Problem 2. Problem 1.2.7 in the Lecture Notes.

Problem 3. Compute the eigenvalues and eigenvectors of the matrix

$$\begin{bmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{bmatrix},$$

for $\alpha \in \mathbb{R}$.

Problem 4. Given a normal matrix A and a right eigenvector v of A with eigenvalue λ. Show that then the complex conjugate of v, $\bar{v}$, is a left eigenvector of A with the same eigenvalue λ, i.e.

$$v^* A = \lambda v^*.$$

Numerical Linear Algebra - Sheet 2 Solutions

Solution 1.

Solution 2.

Solution 3.

Solution 4.

1.
2.
3. (Programming)
4.

Numerical Linear Algebra - Sheet 3

to be handed in until November 8, 2023, 11am.

Problem 1. Problem 2.1.23 in the Lecture Notes.

Problem 2. Problem 2.1.24 in the Lecture Notes.

Problem 3. Problem 2.2.14 in the Lecture Notes.

Problem 4. Problem 2.3.1 in the Lecture Notes.

Numerical Linear Algebra - Sheet 3
Solutions

Solution 1.

Solution 2.

Solution 3.

Solution 4 (Programming)**.**

Numerical Linear Algebra - Sheet 4

to be handed in until November 15, 2023,11am.

Problem 1. Write a program `COMPUTE_EIGENVALUES` that

1. computes the dominant eigenvalue of a matrix using the power method.
2. computes all further eigenvalues using suitable matrix polynomials.

Test your program with the following matrix

$$\mathbf{A} = \begin{pmatrix} 1+i & 1 & 0 & 0 & 0 \\ 0 & 1+i & 1 & 0 & 0 \\ 0 & 0 & 1+i & 1 & 0 \\ 0 & 0 & 0 & 1+i & 1 \\ 1 & 0 & 0 & 0 & 1+i \end{pmatrix}.$$

Problem 2. Let $\mathbf{A}$ be a symmetric tridiagonal matrix. Show that the QR-iteration (see Algorithm 2.4.2 in the lecture notes) preserves the tridiagonal structure of the matrix, i.e., all iterates $\mathbf{A}^{(n)}$ generated by the QR-iteration are tridiagonal.

Problem 3. Rewrite the QR factorization of a tridiagonal (complex) symmetric matrix such that its complexity is of order $O(n)$ (this proves the second part of Corollary 2.4.17 in the lecture notes).

Numerical Linear Algebra - Sheet 4 Solutions

Solution 1.

Solution 2.

Solution 3.

Solution 4.

Solution 5 (Programming)**.** The eigenvalues of $\mathbf{A}_n$ are

$$\lambda_{n,j} = 4\sin^2\left(\frac{j\pi}{2n+2}\right)$$

https://math.stackexchange.com/questions/3875168/eigenvalues-of-a-tridiagonal-matrix-with-1-2-1-as-entries

https://math.stackexchange.com/questions/177957/eigenvalues-of-tridiagonal-symmetric-matrix-with-diagonal-entries-2-and-subdiago?rq=1

Numerical Linear Algebra - Sheet 5

to be handed in until November 22, 2023, 11am

Problem 1. Problem 2.4.16 in the Lecture Notes:

1. How many operations do the two versions of the Hessenberg QR step require?
2. Show that if $\mathbf{H}$ is Hermitian, the result of the Hessenberg QR step is Hermitian as well.

Problem 2. Problem 2.4.20 in the Lecture Notes:
Show that every (complex) Hermitian matrix is orthogonally similar to a symmetric tridiagonal matrix with real entries.

Problem 3.

(a) Implement the implicit Hessenberg QR step (Algorithm 2.4.9 in the lecture notes) in real arithmetic.

(b) Test your code with the tridiagonal matrix $\mathbf{A}_n = \text{tridiag}(-1., 2., -1.)$ in dimension $n = 4$.

(c) Use your implementation to run several steps of the QR iteration (Algorithm 2.4.19 in the lecture notes) for the matrix $\mathbf{A}_{10}$. You should obtain the following eigenvalues:

$$\lambda_j = 2 - 2\cos(j\vartheta) = 4\sin^2\left(\frac{j\vartheta}{2}\right)$$

where $\vartheta = \frac{\pi}{n+1}$, $1 \leq j \leq n$.

(d) Discuss the observed convergence of the off-diagonal and diagonal entries, respectively.

Problem 4.

(a) Implement the QR iteration with shift (Algorithm 2.4.25 in the Lecture Notes) using the Wilkinson shift (Definition 2.4.30).

(b) Test your implementation with the matrix $\mathbf{A}_{10}$. Run several steps of the iteration and observe the behaviour of the subdiagonal elements, especially the last one.

(c) For the curious: Implement the QR iteration with deflation (cf. Algorithm 2.4.35).

Numerical Linear Algebra - Sheet 5
Solutions

Solution 1.

Solution 2.

Solution 3.

Solution 4 (Programming)**.**

Numerical Linear Algebra - Sheet 6

to be handed in until November 29, 2023, 11am.

Problem 1. Prove Lemma 2.5.2:
Let $\mathbf{T} \in \mathbb{R}^{n\times n}$ be a real, symmetric, tridiagonal matrix and $\mathbf{QR} = \mathbf{T}$ its QR factorization. Then, $\tilde{\mathbf{T}} = \mathbf{RQ}$ is also symmetric and tridiagonal. Furthermore, $\mathbf{R}$ is zero except for its main and the first two upper diagonals.

The same holds for the shifted version with $\sigma \in \mathbb{R}$,

$$\mathbf{QR} = \mathbf{T} - \sigma\mathbb{I}, \qquad \tilde{\mathbf{T}} = \mathbf{RQ} + \sigma\mathbb{I}.$$

Problem 2. Prove that in case of a normal real matrix, for each complex eigenvalue pair there is a 2×2 matrix with according invariant subspace.

(a) Show that complex eigenvalues and their associated eigenvectors of a real matrix come in complex conjugate pairs.

(b) Choose real linear combinations of these vectors to obtain the 2×2 block.

Problem 3. Let $\mathbf{H}$ be a real Hessenberg matrix. Let $\mathbf{Q}_1, \mathbf{R}_1, \mathbf{Q}_2, \mathbf{R}_2$ be the matrices obtained in the explicit double-shift QR step (Algorithm 2.5.14). Let σ_1 and σ_2 are the shifts in Lemma 2.5.16. Show that the following equation in the proof of Lemma 2.5.16 holds:

$$\mathbf{Q}_1\mathbf{Q}_2\mathbf{R}_2\mathbf{R}_1 = \mathbf{M} = (\mathbf{H} - \sigma_1\mathbb{I})(\mathbf{H} - \sigma_2\mathbb{I})$$

Problem 4. Consider the $n \times n$ tridiagonal matrix

$$\mathbf{T}_\alpha = \begin{pmatrix} \alpha & -1 & & & \\ -1 & \alpha & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & \alpha & -1 \\ & & & -1 & \alpha \end{pmatrix}$$

where α is a real parameter. Verify that the eigenvalues of $\mathbf{T}_\alpha$ are given by

$$\lambda_j = \alpha - 2\cos(j\vartheta),$$

where

$$\vartheta = \frac{\pi}{n+1},$$

and the eigenvectors associated with each λ_j are given by

$$\mathbf{v}_j = \left(\sin(j\vartheta),\ \sin(2j\vartheta),\ \dots\ ,\ \sin(nj\vartheta)\right)^T.$$

What are the conditions on α, such that the matrix $\mathbf{T}_\alpha$ becomes positive-definite?

Problem 5. Implement the inverse power method (Algorithm 2.6.1 in the lecture notes). To solve the system of linear equations use the functions `numpy.linalg.solve` and `numpy.linalg.lstsq`. Test your implementations with the matrix $\mathbf{A}_{20}$ and $\alpha = 2$ from Problem 4 and shift parameters $\sigma = \lambda_1, \lambda_2, \lambda_{10}$. Compare your results for the two different solvers.

Numerical Linear Algebra - Sheet 6
Solutions

Solution 1.

Solution 2.

Solution 3.

Solution 4.

Numerical Linear Algebra - Sheet 7

to be handed in until December 6, 2023, 11am.

Problem 1. Prove Theorem 3.2.15 in the Lecture Notes (Convergence of Richardson Iteration).

Problem 2 (Programming)**.** Let $\mathbf{L}_d$ be the discretization of the d-dimensional Laplace operator on the unit square by the five-point stencel with a uniform Dirichlet boundary condition to the mesh size $h = \frac{1}{n+1}$. In 1D it is given in terms of the matrix $\mathbf{T}_2$ defined in Problem 4 of Sheet 6 by

$$\mathbf{L}_1 = \frac{1}{h^2}\mathbf{T}_2 \in \mathbb{R}^{n\times n}.$$

In 2D it is given by

$$\mathbf{L}_2 = \frac{1}{h^2}\begin{pmatrix} \mathbf{D} & -\mathbb{I} & & & \\ -\mathbb{I} & \mathbf{D} & -\mathbb{I} & & \\ & \ddots & \ddots & \ddots & \\ & & -\mathbb{I} & \mathbf{D} & -\mathbb{I} \\ & & & -\mathbb{I} & \mathbf{D} \end{pmatrix} \in \mathbb{R}^{n^2\times n^2},$$

where $\mathbb{I} \in \mathbb{R}^{n\times n}$ and $\mathbf{D} \in \mathbb{R}^{n\times n}$ are defined as

$$\mathbb{I} = \begin{pmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{pmatrix} \quad \text{and} \quad \mathbf{D} = \begin{pmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 4 & -1 \\ & & & -1 & 4 \end{pmatrix}.$$

Details may be found in appendix B.3 on finite difference methods in the lecture notes.

Let $\mathbf{A} = \mathbf{L}_2 \in \mathbb{R}^{n^2\times n^2}$ as defined above and $\mathbf{b} \in \mathbb{R}^{n^2}$. Write a program that solves the 2D Laplace problem

$$\mathbf{A}\mathbf{x} = \mathbf{b},$$

with the Richardson iteration by observing the following steps:

(a) Implement a function vmult that performs the matrix-vector product $\mathbf{A}\mathbf{v}$ of $\mathbf{A}$ with a given vector $\mathbf{v}$, and returns a vector $\mathbf{w}$. Do not store the system matrix in this function. The parameters should only be the vector $\mathbf{v}$ and, optionally, n.

(b) Write a function richardson_step that implements a single step of the Richardson iteration. Similar to (a) this function should take vectors $\mathbf{v}$ and $\mathbf{b}$, optionally n, and return the resulting vector $\mathbf{w}$. *Hint: The eigenvalues of* $\mathbf{L}_2$ *are given by the pairwise sums of the eigenvalues of* $\mathbf{T}_2$*, see B.3.13.*

(c) Use the null-vector $\mathbf{x}^{(0)} = (0, ..., 0)^T$ as initial guess and test your program for the constant vector $\mathbf{b} = (1, ..., 1)^T$ with $n = 20$ and $n = 100$. Observe the evolution of the residual in the 2-norm

$$r^{(k)} = \|\mathbf{A} \cdot \mathbf{x}^{(k)} - \mathbf{b}\|_2$$

for 50 steps of the Richardson iteration.

(d) Plot the obtained result vector after 1, 5, 15, 30, and 50 iterations as a two-dimensional function. *Hint: For plotting you have to invert the numbering of the vector (use, for example, np.reshape) and then plot the two dimensional grid using a numpy color map (greyscale).*

(e) Compute and discuss the observed convergence rate (see Definition 3.2.22 in the Lecture Notes).

Numerical Linear Algebra - Sheet 7 Solutions

Solution 1.

Solution 2.

Solution 3.

Solution 4 (Programming)**.** C++ sample code ./programs/gauss−seidel.cc without assembling the system matrix. Convergence

Numerical Linear Algebra - Sheet 8

to be handed in until December 13, 2023, 11am.

Problem 1. Let $\mathbf{Ax} = \mathbf{b}$ be a linear system with a symmetric, positive definite matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ that has extremal eigenvalues λ_{min} and λ_{max} and spectral condition number $\text{cond}(\mathbf{A})$. Consider the sequence $\left\{\mathbf{x}^{(k)}\right\}$ of one-dimensional projection processes with $K = L = \text{span}\{\mathbf{e}_i\}$, where $\mathbf{e}_i$ denotes the i-th unit vector in $\mathbb{R}^n$. The index i is selected at each step k to be the index of a component of largest absolute value in the current residual vector $\mathbf{r}^{(k)} = \mathbf{b} - \mathbf{Ax}^{(k)}$.

(a) For $\mathbf{d}^{(k)} = \mathbf{A}^{-1}\mathbf{b} - \mathbf{x}^{(k)}$ show that

$$\|\mathbf{d}^{(k+1)}\|_{\mathbf{A}} \le \left(1 - \frac{1}{n\,\text{cond}(\mathbf{A})}\right)^{\frac{1}{2}} \|\mathbf{d}^{(k)}\|_{\mathbf{A}}.$$

Hint: You can use the expression

$$\left\langle \mathbf{Ad}^{(k+1)}, \mathbf{d}^{(k+1)} \right\rangle = \left\langle \mathbf{Ad}^{(k)}, \mathbf{d}^{(k)} \right\rangle - \frac{\left\langle \mathbf{r}^{(k)}, \mathbf{e}_i \right\rangle^2}{a_{ii}},$$

as well as the inequality $|\mathbf{e}_i^T \mathbf{r}^{(k)}| \ge n^{-1/2}\|\mathbf{r}^{(k)}\|_2$. a_{ii} *denotes the* i*-th diagonal element of* A.

(b) Does (a) prove that the algorithm converges?

Problem 2. Consider the linear system $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A}$ is a symmetric positive definite matrix. We define a projection method which uses a two-dimensional space at each step. At a given step, take $L = K = \text{span}\{\mathbf{r}, \mathbf{Ar}\}$, where $\mathbf{r} = \mathbf{b} - \mathbf{Ax}$ is the current residual.

(a) For a basis of K use the vector $\mathbf{r}$ and the vector $\mathbf{p}$ obtained by orthogonalizing $\mathbf{Ar}$ against $\mathbf{r}$ with respect to the $\mathbf{A}$-inner product. Give the formula for computing $\mathbf{p}$ (no need to normalize the resulting vector).

(b) Write the algorithm for performing the projection method described above.

(c) Can you explain, not compute, why the algorithm converges for any initial guess $\mathbf{x}_0$? *Hint: Exploit the convergence results for one-dimensional projection techniques.*

Problem 3 (Programming)**.**

(a) Implement the steepest decent method (Algorithm 3.3.13 in the lecture notes).

(b) Use your implementation of the steepest decent method to solve the 2D Laplace problem

$$\mathbf{L}_2\mathbf{x} = \mathbf{b}$$

as in Problem 2 on Sheet 7 with right hand side vector $\mathbf{b} = (1, ..., 1)^T$ and initial guess $\mathbf{x}^{(0)} = (0, ..., 0)^T$ with $n = 50$ and $n = 100$.

(c) Compute and discuss the observed convergence rate (see 3.2.21/Definition 3.2.22 in the Lecture Notes).

Problem 4 (Programming)**.**

(a) Implement the minimal residual method (Algorithm 3.3.20 in the lecture notes).

(b) Use your implementation of the steepest decent method to solve the 2D Laplace problem

$$\mathbf{L}_2\mathbf{x} = \mathbf{b}$$

as in Problem 2 on Sheet 7 with right hand side vector $\mathbf{b} = (1, ..., 1)^T$ and initial guess $\mathbf{x}^{(0)} = (0, ..., 0)^T$ with $n = 50$ and $n = 100$.

(c) Compute and discuss the observed convergence rate (see 3.2.21/Definition 3.2.22 in the Lecture Notes). Compare it with the convergence rate observed in Problem 3.

Numerical Linear Algebra - Sheet 8 Solutions

Solution 1 ([?, P-5.1 b]).
Part (a):

Hint 1:

$$\left\langle \mathbf{A}\mathbf{d}^{(k+1)}, \mathbf{d}^{(k+1)} \right\rangle = \left\langle \mathbf{A}\mathbf{d}^{(k)}, \mathbf{d}^{(k)} \right\rangle - \frac{\left\langle \mathbf{r}^{(k)}, \mathbf{e}_i \right\rangle^2}{a_{ii}}$$

Hint 2:

$$|\mathbf{e}_i^T \mathbf{r}^{(k)}| \geq n^{-1/2} \|\mathbf{r}^{(k)}\|_2$$

With the help of hint 1 and 2 we get

$$\begin{aligned}
\|\mathbf{d}^{(k+1)}\|_{\mathbf{A}}^2 = \left\langle \mathbf{A}\mathbf{d}^{(k+1)}, \mathbf{d}^{(k+1)} \right\rangle &\overset{\text{Hint 1}}{=} \left\langle \mathbf{A}\mathbf{d}^{(k)}, \mathbf{d}^{(k)} \right\rangle - \frac{\left\langle \mathbf{r}^{(k)}, \mathbf{e}_i \right\rangle^2}{a_{ii}} \\
&= \left(1 - \frac{\left\langle \mathbf{r}^{(k)}, \mathbf{e}_i \right\rangle^2}{a_{ii}} \frac{1}{\left\langle \mathbf{A}\mathbf{d}^{(k)}, \mathbf{d}^{(k)} \right\rangle} \right) \|\mathbf{d}^{(k)}\|_{\mathbf{A}}^2 \\
&\overset{\text{Hint 2}}{\leq} \left(1 - \frac{1}{n a_{ii}} \frac{\|\mathbf{r}^{(k)}\|_2^2}{\left\langle \mathbf{A}\mathbf{d}^{(k)}, \mathbf{d}^{(k)} \right\rangle} \right) \|\mathbf{d}^{(k)}\|_{\mathbf{A}}^2
\end{aligned}$$

The result follows by the following two estimates of Rayleigh quotients:

$$a_{ii} = \langle \mathbf{A}\mathbf{e}_i, \mathbf{e}_i \rangle = \frac{\langle \mathbf{A}\mathbf{e}_i, \mathbf{e}_i \rangle}{\langle \mathbf{e}_i, \mathbf{e}_i \rangle} \leq \lambda_{\max},$$

and

$$\frac{\left\langle \mathbf{A}\mathbf{d}^{(k)}, \mathbf{d}^{(k)} \right\rangle}{\|\mathbf{r}^{(k)}\|^2} = \frac{\left\langle \mathbf{r}^{(k)}, \mathbf{A}^{-1}\mathbf{r}^{(k)} \right\rangle}{\|\mathbf{r}^{(k)}\|^2} \leq \lambda_{\max}(\mathbf{A}^{-1}) = \frac{1}{\lambda_{\min}}.$$

Part (b): For $n > 1$, $1 - \frac{1}{n \operatorname{cond}(\mathbf{A})} < 1$, and thus

$$\|\mathbf{d}^{(k+1)}\|_{\mathbf{A}} \leq \left(1 - \frac{1}{n \operatorname{cond}(\mathbf{A})} \right)^{\frac{k+1}{2}} \|\mathbf{d}^{(0)}\|_{\mathbf{A}} \to 0 \text{ for } n \to \infty.$$

That shows that $x^{(k)}$ converges to the solution $x = \mathbf{A}^{-1}\mathbf{b}$.

Final remarks:
1. Since $\mathbf{e}_i \in K = L$, it holds the Galerkin-orthogonality

$$\left\langle \mathbf{r}^{(k)} - \alpha \mathbf{A}\mathbf{e}_i, \mathbf{e}_i \right\rangle = 0,$$

where according to Example 2.3.10

$$\alpha = \frac{\langle \mathbf{r}^{(k)}, \mathbf{e}_i \rangle}{\langle \mathbf{A}\mathbf{e}_i, \mathbf{e}_i \rangle} = \frac{\langle \mathbf{r}^{(k)}, \mathbf{e}_i \rangle}{a_{ii}}.$$

This would have been used to show hint 1.

2. To show hint 2, notice that by the choice of i as maximal value of the residual vector

$$\|\mathbf{r}^{(k)}\|_2 \leq \sqrt{n}\|\mathbf{r}^{(k)}\|_\infty = \sqrt{n}|\mathbf{e}_i^T \mathbf{r}^{(k)}|$$

Solution 2. Part (a):

$$\mathbf{p} = \mathbf{A}\mathbf{r} - \frac{\langle \mathbf{A}\mathbf{r}, \mathbf{r} \rangle_{\mathbf{A}}}{\langle \mathbf{r}, \mathbf{r} \rangle_{\mathbf{A}}}\mathbf{r}.$$

Then $\langle \mathbf{A}\mathbf{r}, \mathbf{p} \rangle = \langle \mathbf{r}, \mathbf{p} \rangle_{\mathbf{A}} = 0$.

Part(b): As defined in Definition 3.3.3, we choose $\mathbf{v} \in K = L$, i.e. $\mathbf{v}$ is a linear combination of $\mathbf{p}$ and $\mathbf{r}$. From the second condition $\mathbf{r}^{(k)} - \mathbf{A}\mathbf{v} \perp L$, we can deduce the coefficients for v:

$$\mathbf{r}^{(k)} - \mathbf{A}\mathbf{v} \perp L \Rightarrow \langle \mathbf{r} - \mathbf{A}\mathbf{v}, \mathbf{r} \rangle = 0, \langle \mathbf{r} - \mathbf{A}\mathbf{v}, \mathbf{p} \rangle = 0.$$

Let $\mathbf{v} = \alpha\mathbf{r} + \beta\mathbf{p}$.

$$\begin{aligned}
\langle \mathbf{r} - \mathbf{A}\mathbf{v}, \mathbf{r} \rangle &= 0 \\
\Leftrightarrow \langle \mathbf{r}, \mathbf{r} \rangle - \langle \alpha\mathbf{A}\mathbf{r} + \beta\mathbf{A}\mathbf{p}, \mathbf{r} \rangle &= 0 \\
\Leftrightarrow \langle \mathbf{r}, \mathbf{r} \rangle - \alpha\langle \mathbf{A}\mathbf{r}, \mathbf{r} \rangle - \beta\langle \mathbf{A}\mathbf{p}, \mathbf{r} \rangle &= 0 \\
\Leftrightarrow \langle \mathbf{r}, \mathbf{r} \rangle - \alpha\langle \mathbf{A}\mathbf{r}, \mathbf{r} \rangle - \beta\langle \mathbf{p}, \mathbf{A}\mathbf{r} \rangle &= 0 \\
\Leftrightarrow \langle \mathbf{r}, \mathbf{r} \rangle - \alpha\langle \mathbf{A}\mathbf{r}, \mathbf{r} \rangle &= 0 \\
\Leftrightarrow \alpha &= \frac{\langle \mathbf{r}, \mathbf{r} \rangle}{\langle \mathbf{A}\mathbf{r}, \mathbf{r} \rangle}
\end{aligned}$$

$$\begin{aligned}
\langle \mathbf{r} - \mathbf{A}\mathbf{v}, \mathbf{p} \rangle &= 0 \\
\Leftrightarrow \langle \mathbf{r}, \mathbf{p} \rangle - \langle \alpha\mathbf{A}\mathbf{r} + \beta\mathbf{A}\mathbf{p}, \mathbf{p} \rangle &= 0 \\
\Leftrightarrow \langle \mathbf{r}, \mathbf{p} \rangle - \alpha\langle \mathbf{A}\mathbf{r}, \mathbf{p} \rangle - \beta\langle \mathbf{A}\mathbf{p}, \mathbf{p} \rangle &= 0 \\
\Leftrightarrow \langle \mathbf{r}, \mathbf{p} \rangle - \beta\langle \mathbf{A}\mathbf{p}, \mathbf{p} \rangle &= 0 \\
\Leftrightarrow \beta &= \frac{\langle \mathbf{r}, \mathbf{p} \rangle}{\langle \mathbf{A}\mathbf{p}, \mathbf{p} \rangle} \\
\Leftrightarrow \beta &= \frac{\langle \mathbf{r}, \mathbf{p} \rangle}{\langle \mathbf{A}^2\mathbf{r}, \mathbf{p} \rangle}.
\end{aligned}$$

In each step, first update $\mathbf{p}$, then compute α and β and then update $\mathbf{x}$.

Part (c): The method is similar to the steepest decent method. In the steepest decent method we choose $K_{\text{sd}} = L_{\text{sd}} = \text{span}\,\{\mathbf{r}\}$ and in the method described here we choose $K = L = \text{span}\,\{\mathbf{r}, \mathbf{A}\mathbf{r}\}$. In each step $K_{\text{sd}} \subset K$ and $L_{\text{sd}} \subset L$. Thus, the method is not converging worse than the steepest decent method. Therefore, it converges.

Solution 3 (Programming). C++ sample code ./programs/steepest−decent.cc without assembling the system matrix.

Convergence rate (Definition 2.2.18 in lecture notes). As can be seen in the plot, the convergence of the steepest decent method is sublinear.

Numerical Linear Algebra - Sheet 9

discussion in the first tutorials of January, 2023

This exercise sheet reviews some topics of this lecture. Your answers do not have to be handed in, but the exercises will be discussed in the tutorials of the first week of January.

Problem 1. Recapitulate the concept of an orthogonal projection and an oblique projection. What are use cases of both and why are they important for numerical linear algebra?

Problem 2. What methods presented in the lecture can be used for computing or estimating the eigenvalues of a matrix $\mathbf{A}$? Sort by properties of the methods, as well as by conditions on $\mathbf{A}$.

Problem 3. How can the QR factorization of a given matrix be computed? Discuss downsides and benefits of the different methods.

Problem 4. Householder reflections vs. Givens rotations: which one is more cost-efficient in the general case? When using the other one is advantageous?

Problem 5. Which of the discussed methods for solving eigenvalue problems can be implemented without explicitly forming a matrix?

Problem 6. Consider an $n \times n$ matrix that has n distinct eigenvalues such that $|\lambda_i| \neq |\lambda_j|$ for $i \neq j$. How can the eigenvalue with the second largest absolute value be computed?

Problem 7. When does one step of the Gauss-Seidel iteration provide the direct solution of a linear system? Consider an upper triangular matrix to visualize this matter.

Numerical Linear Algebra - Sheet 9 Solutions

Solution 1.

Solution 2.

Solution 3.

Solution 4.

Solution 5.

Solution 6.

Solution 7.

Numerical Linear Algebra - Sheet 10

to be handed in until January 17, 2024, 11am.

Problem 1. Show how GMRES (Algorithm 3.4.42 in the lecture notes) and Arnoldi with modified Gram-Schmidt (Algorithm 3.4.10 in the lecture notes) will converge on the linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ when

$$\mathbf{A} = \begin{pmatrix} & & & & 1 \\ 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

and $\mathbf{x}_0 = \mathbf{0}$.

Problem 2. Let $\mathbf{A}$ and $\mathbf{B}$ be symmetric positive-definite matrices in $\mathbb{R}^{n\times n}$. Let there be constants c_1, c_2 such that

$$c_1 \mathbf{x}^T \mathbf{A}\mathbf{x} \leq \mathbf{x}^T \mathbf{B}\mathbf{x} \leq c_2 \mathbf{x}^T \mathbf{A}\mathbf{x}.$$

Derive an estimate for the convergence of the conjugate gradient iteration for $\mathbf{A}\mathbf{x} = \mathbf{b}$ with preconditioner $\mathbf{B}^{-1}$ depending on c_1 and c_2.

Hints: Lemma 3.4.68 in the lecture notes, and Thereom 2.2.17 (Courant-Fischer min-max theorem) (note, that the definition of the Rayleigh quotient is independent of the choice of the inner product).

Problem 3. Consider again matrix $\mathbf{T}_\alpha$ as introduced in Problem 4 on sheet Sheet 6.

(a) Implement a method that computes the smallest eigenvalue of the matrix $\mathbf{T}_\alpha$ using the inverse iteration (Algorithm 2.6.1 in the lecture notes). You may reuse your code from Problem 5 on sheet Sheet 6. Do not calculate the inverse explicitly for solving the appearing linear system, but use the conjugate gradient iteration (Algorithm 3.4.28 in the lecture notes) for this matter.

(b) Use your implementation to calculate the smallest eigenvalue of $\mathbf{T}_\alpha$ with $\alpha = 2$ and $n = 20$.

Numerical Linear Algebra - Sheet 10 Solutions

Solution 1.

Solution 2.

Solution 3.

Solution 4 (Programming)**.**

Numerical Linear Algebra - Sheet 11

to be handed in until January 24, 2024, 11am.

Problem 1. Use your implementation of Problem 2(a) on Sheet 9 and augment it such that you can compute the eigenvalues of the projected matrix $\mathbf{H}_m$ in each step. For the computation of the eigenvalues you could, for example, use your implementation of the QR iteration (Problem 4 on Sheet 5). Test your implementation with $\mathbf{L}_2$ as in Problem 2(b) on Sheet 9 and observe the convergence of extremal eigenvalues $\lambda_{\min}$ and $\lambda_{\max}$.

The following problems review some topics of this lecture. They are a good preparation for the oral exam and they will also help with better understanding the programming problem and project. Please summarize your thoughts on those using bullet points.

Problem 2. Recapitulate the concept of an orthogonal projection and an oblique projection. What are use cases of both and why are they important for numerical linear algebra?

Problem 3. What methods presented in the lecture can be used for computing or estimating the eigenvalues of a matrix $\mathbf{A}$? Sort by properties of the methods, as well as by conditions on $\mathbf{A}$.

Problem 4. How can the QR factorization of a given matrix be computed? Discuss downsides and benefits of the different methods.

Numerical Linear Algebra - Sheet 11 Solutions

Solution 1.

Solution 2 (Programming)**.**

Numerical Linear Algebra - Sheet 12

to be handed in until January 31, 2024, 11am.

The following problems review some topics of this lecture. They are a good preparation for the oral exam and they will also help with better understanding the programming problem and project. Please summarize your thoughts on those using bullet points.

Problem 1. Householder reflections vs. Givens rotations: which one is more cost-efficient in the general case? When using the other one is advantageous?

Problem 2. Which of the discussed methods for solving eigenvalue problems can be implemented without explicitly forming a matrix?

Problem 3. Consider an $n \times n$ matrix that has n distinct eigenvalues such that $|\lambda_i| \neq |\lambda_j|$ for $i \neq j$. How can the eigenvalue with the second largest absolute value be computed?

Problem 4. When does one step of the Gauss-Seidel iteration provide the direct solution of a linear system? Consider an upper triangular matrix to visualize this matter.

Numerical Linear Algebra - Sheet 11 Solutions

Solution 1.

Solution 2 (Programming)**.**