

ETH Lectures 401-0663-00L Numerical Methods for Computer Science
401-2673-00L Numerical Methods for CSE

Homework Problems and Projects

Prof. R. Hiptmair, SAM, ETH Zurich

Autumn Term 2021, Version of October 2, 2022
(C) Seminar für Angewandte Mathematik, ETH Zürich

[Link](#) to the current version of this homework collection

0.1 General Information

0.1.1 Weekly Homework Assignments

All problems will be published in this single “.pdf” file. Every week, we publish a list of problem numbers on the lecture Moodle page. These are the current assignments for that week. Details are given on the lecture Moodle page.

0.1.2 Importance of Homework

Homework assignments are **not** mandatory. However, it is **very important** that you constantly exercise with the material you learn. “Solving” the homework assignments one week before the main exam by looking at the solutions will likely result in failure.

We provide hints and solutions from the beginning. It is your responsibility to look at the solutions only if you are stuck with a difficult problem and you tried to find a solution for a sufficient amount of time.

Most homework problems come with an estimate for the time required for its solution. These times are what we expect a bright and well-prepared student to require for that problem during an exam.

0.1.3 Corrections and Grading of Assignments

Homework assignments will not be graded.

However, you may submit your handwritten solutions by handing them to your tutor directly or uploading them via the Moodle upload interface, see the lecture Moodle page for instructions. The assistants will

Problem 3-2: Estimating a Tridiagonal Matrix

In [Lecture → Section 3.1] we learned that to determine the least squares solution of an overdetermined linear system of equations $\mathbf{Ax} = \mathbf{b}$ we minimize the residual norm $\|\mathbf{Ax} - \mathbf{b}\|_2$ w.r.t. $\mathbf{x}$. However, we also face a linear least squares problem when minimizing the residual norm w.r.t. the entries of $\mathbf{A}$. This is the unusual situation considered in this problem.

This is an exercise in converting a problem statement into a proper linear least squares problem in the form of an overdetermined linear system of equations as in [Lecture → Section 3.0.1] and then solving it through the normal equations, see [Lecture → Section 3.2].

The following least-squares problem arises in the estimation of material parameters in one-dimensional diffusion problems: Let two vectors $\mathbf{z}, \mathbf{c} \in \mathbb{R}^n$, $n > 2 \in \mathbb{N}$ of measured values be given. The two real numbers α^* and β^* are defined as:

$$(\alpha^*, \beta^*) = \operatorname{argmin}_{\alpha, \beta \in \mathbb{R}} \|\mathbf{T}_{\alpha, \beta} \mathbf{z} - \mathbf{c}\|_2, \quad (3.2.1)$$

where $\mathbf{T}_{\alpha, \beta} \in \mathbb{R}^{n \times n}$ is the following **tridiagonal matrix**

$$\mathbf{T}_{\alpha, \beta} = \begin{bmatrix} \alpha & \beta & 0 & \dots & 0 \\ \beta & \alpha & \beta & \ddots & \vdots \\ 0 & \beta & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \alpha & \beta \\ 0 & \dots & 0 & \beta & \alpha \end{bmatrix}. \quad (3.2.2)$$

(3-2.a) ☞ (20 min.) Reformulate (3.2.1) as a linear least squares problem in standard form according to [Lecture → Def. 3.1.1.1]

$$\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x} \in \mathbb{R}^k} \|\mathbf{Ax} - \mathbf{b}\|_2 \quad (3.2.3)$$

For appropriate choice of $m, k \in \mathbb{N}$ define suitable $\mathbf{A} \in \mathbb{R}^{m, k}$, $\mathbf{x} \in \mathbb{R}^k$ and $\mathbf{b} \in \mathbb{R}^m$, with $m, k \in \mathbb{N}$.

HINT 1 for (3-2.a): Compute the matrix×vector product $\mathbf{T}_{\alpha, \beta} \mathbf{z}$ for using $\mathbf{z} = [z_1, z_2, \dots, z_n]^\top$. Express the resulting vector as the product of a matrix and the vector $\begin{bmatrix} \alpha \\ \beta \end{bmatrix}$. ┐

SOLUTION of (3-2.a):

Clearly we set $\mathbf{x} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$ ($k = 2$) and $\mathbf{b} := \mathbf{c}$ ($m = n$). The vector $\mathbf{T}_{\alpha, \beta} \mathbf{z} - \mathbf{c}$ whose norm has to be minimized in ?? can be written as:

$$\mathbf{T}_{\alpha, \beta} \mathbf{z} - \mathbf{c} = \begin{bmatrix} \alpha z_1 + \beta z_2 \\ \alpha z_2 + \beta(z_1 + z_3) \\ \dots \\ \alpha z_{n-1} + \beta(z_{n-2} + z_n) \\ \alpha z_n + \beta z_{n-1} \end{bmatrix} - \mathbf{c} = \begin{bmatrix} z_1 & z_2 \\ z_2 & z_1 + z_3 \\ \vdots & \vdots \\ z_{n-1} & z_{n-2} + z_n \\ z_n & z_{n-1} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} - \mathbf{c} =: \mathbf{Ax} - \mathbf{b} \quad (3.2.4)$$

In this form, we have moved the unknown α and β into vector $\mathbf{x}$, while the data $\mathbf{z}$ and $\mathbf{c}$ are moved into matrix $\mathbf{A}$ and the right-hand side vector $\mathbf{b} \equiv \mathbf{c}$ (which is not affected by the transformation). The number k of unknowns is equal to 2 and the number m of equations is equal to n .



(3-2.b) 🕒 (15 min.) [depends on Sub-problem (3-2.a)]

Explicitly state the **normal equations** [Lecture → Thm. 3.1.2.1] associated with the linear least-squares problem (3.2.3).

SOLUTION of (3-2.b):

The normal equations are

$$\mathbf{A}^\top \mathbf{A} \mathbf{x} = \mathbf{A}^\top \mathbf{c}, \quad \text{for } \mathbf{A} := \begin{bmatrix} z_1 & z_2 \\ z_2 & z_1 + z_3 \\ \vdots & \vdots \\ z_{n-1} & z_{n-2} + z_n \\ z_n & z_{n-1} \end{bmatrix} \in \mathbb{R}^{n,2}.$$

Computing the matrix product, we find

$$\mathbf{A}^\top \mathbf{A} = \begin{bmatrix} \sum_{\ell=1}^n z_\ell^2 & 2 \sum_{\ell=1}^{n-1} z_\ell z_{\ell+1} \\ 2 \sum_{\ell=1}^{n-1} z_\ell z_{\ell+1} & z_1^2 + 2 \sum_{\ell=2}^{n-1} (z_\ell^2 + z_{\ell-1} z_{\ell+1}) + z_n^2 \end{bmatrix} \in \mathbb{R}^{2,2}.$$

▲

(3-2.c) 🕒 (30 min.) [depends on Sub-problem (3-2.b)]

Write a C++ function

```
Eigen::VectorXd lsqEst(const Eigen::VectorXd &z,
                      const Eigen::VectorXd &c);
```

that computes the optimal (in least-squares sense) parameters α^* and β^* according to ?? from data vectors $\mathbf{z}$ and $\mathbf{c}$ (i.e. $\mathbf{z}$ and $\mathbf{c}$ from ??). Use the *method of normal equations*.

HINT 1 for (3-2.c): Similarly to Problem 3-1, $\mathbf{A}\mathbf{x} = \mathbf{b}$ is an overdetermined problem: the number of equations, i.e. the number of rows of $\mathbf{A}$, n , is greater than the number of unknowns, i.e. the number of columns of $\mathbf{A}$, 2. We must therefore solve the normal equations $\mathbf{A}^\top \mathbf{A} \mathbf{x} = \mathbf{A}^\top \mathbf{b}$ with EIGEN. ┘

After the matrix $\mathbf{A}$ has been initialized, the solution just copies [Lecture → Code 3.2.0.1].

SOLUTION of (3-2.c):

C++ code 3.2.5: Solution of (3-2.c)

```
2 Eigen::VectorXd lsqEst(const Eigen::VectorXd& z, const Eigen::VectorXd& c) {
3   assert(z.size() == c.size() && "z and c must have same size");
4   Eigen::VectorXd x = Eigen::VectorXd::Zero(2); // (α, β).
5
6   // TODO: (3-2.c) Solve eq. (3.2.1) using the method of normal
7   // equations.
8   // START
9
10  // Reformulating  $T_{\alpha,\beta}z = c$  as  $Ax = b$  with  $x = (\alpha, \beta)^\top$ ,
11  // we have  $b = c$ , and  $A$  defined as follows.
```

```
11  const unsigned int n = z.size();
12  Eigen::MatrixXd A(n, 2);
13  // Initialize all entries of A:
14  A.col(0) = z;
15  A(0, 1) = 0;
16  A.col(1).tail(n - 1) = z.head(n - 1);
17  // Add remaining terms:
18  A.col(1).head(n - 1) += z.tail(n - 1);
19  // Normal equation linear system
20  Eigen::MatrixXd lhs = A.transpose() * A; // Left-hand side
21  Eigen::VectorXd rhs = A.transpose() * c; // Right-hand side
22  // Least squares estimate of  $(\alpha, \beta)$  as solution
23  // of normal equations.
24  x = lhs.lu().solve(rhs);
25  // END
26  return x;
27 }
```

Get it on  GitLab ([tridiagleastsquares.hpp](#)).



End Problem 3-2 , 65 min.