Prof. R. Hiptmair
G. Alberti,
F. Leonardi

AS 2015

Numerical Methods for CSE

ETH Zürich
D-MATH

Problem Sheet 0

These problems are meant as an introduction to EIGEN in the first tutorial classes of the new semester.

Problem 1. Gram-Schmidt orthogonalization with EIGEN

[1, Code 1.5.3] presents a MATLAB code that effects the Gram-Schmidt orthogonalization of the columns of an argument matrix.

(1a) ⚀ Based on the C++ linear algebra library EIGEN implement a function

```
template <class Matrix>
Matrix gramschmidt(const Matrix &A);
```

that performs the same computations as [1, Code 1.5.3].

(1b) ⚀ Test your implementation by applying it to a small random matrix and checking the orthonormality of the columns of the output matrix.

Problem 2. Fast matrix multiplication

[1, Rem. 1.4.9] presents Strassen's algorithm that can achieve the multiplication of two dense square matrices of size $n = 2^k$, $k \in \mathbb{N}$, with an asymptotic complexity better than $O(n^3)$.

(2a) ⚀ Using EIGEN implement a function

```
MatrixXd strassenMatMult(const MatrixXd & A, const
   MatrixXd & B)
```

that uses Strassen's algorithm to multiply the two matrices $\mathbf{A}$ and $\mathbf{B}$ and return the result as output.

(2b) ⚄ Validate the correctness of your code by comparing the result with EIGEN's built-in matrix multiplication.

(2c) ⚄ Measure the runtime of your function `strassenMatMult` for random matrices of sizes 2^k, $k = 4, \ldots, 10$, and compare with the matrix multiplication offered by the $*$-operator of EIGEN.

Problem 3. Householder reflections

This problem is a supplement to [1, Section 1.5.1] and related to Gram-Schmidt orthogonalization, see [1, Code 1.5.3]. Before you tackle this problem, please make sure that you remember and understand the notion of a QR-decomposition of a matrix, see [1, Thm. 1.5.7]. This problem will put to the test your advanced linear algebra skills.

(3a) ⚄ Listing 2 implements a particular MATLAB function.

Listing 1: MATLAB implementation for Problem 3. in file `houserefl.m`

```
function Z = houserefl(v)
% Porting of houserefl.cpp to Matlab code
% v is a column vector
    % Size of v
    n = size(v,1);

    w = v/norm(v);
    u = w + [1;zeros(n-1,1)];
    q = u/norm(u);
    X = eye(n) - 2*q*q';

    % Remove first column X(:,1) \in span(v)
    Z = X(:,2:end);
end
```

Write a C++ function with declaration:

```
void houserefl(const VectorXd &v, MatrixXd &Z);
```

that is equivalent to the MATLAB function `houserefl()`. Use data types from EIGEN.

(3b) ⚂ Show that the matrix $\mathbf{X}$, defined at line 10 in Listing 2, satisfies:

$$\mathbf{X}^\top \mathbf{X} = \mathbf{I}_n$$

HINT: $\|\mathbf{q}\|^2 = 1$.

(3c) ⚃ Show that the first column of $\mathbf{X}$, after line 9 of the function `houserefl`, is a multiple of the vector $\mathbf{v}$.

HINT: Use the previous hint, and the facts that $\mathbf{u} = \mathbf{w} + \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$ and $\mathbf{w} = 1$.

(3d) ⚁ What property does the set of columns of the matrix $\mathbf{Z}$ have? What is the purpose of the function `houserefl`?

HINT: Use (3b) and (3c).

(3e) ⚀ What is the asymptotic complexity of the function `houserefl` as the length n of the input vector $\mathbf{v}$ goes to ∞?

(3f) ⚂ Rewrite the function as MATLAB function and use a *standard function* of MATLAB to achieve the same result of lines 5-9 with a single call to this function.

HINT: It is worth reading [1, Rem. 1.5.10] before mulling over this problem.

Issue date: 21.09.2015
Hand-in: — (in the boxes in front of HG G 53/54).

Version compiled on: September 20, 2015 (v. 1.1).