Lennart Edsberg

Numerisk Analys, KTH

Applied Numerical Methods, SF2520, Numerical Algebra part

Lab 7, QR-factorization and Linear Least Squares

A. Stable QR-factorization

QR-factorization can be used to solve both *underdetermined*, *square* and *overdetermined* linear systems of equations. It can also be used to compute a numerically stable solution of an *ill-conditioned* system.

With QR-factorization a matrix can be factorized into

$$AP = QR$$

where P is a permutation matrix, making the diagonal elements $r_{i,i}$ in the uppertriangular matrix R be ordered in decending order, i.e. $|r_{i,i}| \leq |r_{i-1,i-1}|$. Q is orthonormal, i.e. $Q^T Q = I$. Assume the matrix A is $n \times n$. The solution of a linear system of equations $A\mathbf{x} = \mathbf{b}$ is obtained from

$$R\mathbf{y} = Q^T\mathbf{b}, \quad \textit{and} \quad \mathbf{x} = P\mathbf{y}$$

If the matrix A is ill-conditioned R can be written on block form

$$R = \begin{pmatrix} R_{11} & R_{12} \\ 0 & E \end{pmatrix}$$

where R_{11} and E are uppertriangular matrices of dimension $r \times r$ and $n-r \times n-r$ respectively, and where the elements of E are very small. The smaller the elements of E are the more ill-conditioned is the matrix A. If the elements of E are exactly zero the matrix A is rank-deficient with $rank(A) = r$.

By setting small but nonzero elements in E to zero we regard the matrix A as rank-deficient even if it isn't that mathematically. Then we cannot find a unique solution, but instead a minimum norm solution to $\hat{A}\hat{\mathbf{x}} = \mathbf{b}$, where

$$\hat{A} = Q \begin{pmatrix} R_{11} & R_{12} \\ 0 & 0 \end{pmatrix} P^T$$

according to the following algorithm:

1) Compute the vectors $\hat{\mathbf{d}}$ and $\mathbf{d}_E$ from

$$Q^T\mathbf{b} = \begin{pmatrix} \hat{\mathbf{d}} \\ \mathbf{d}_E \end{pmatrix}$$

2) Solve the system

$$R_{11}\hat{\mathbf{y}} = \hat{\mathbf{d}}$$

3) Finally compute

$$\hat{\mathbf{x}} = P \begin{pmatrix} \hat{\mathbf{y}} \\ 0 \end{pmatrix}$$

The goal of this lab is to illustrate the following *numerical relations*

1) the sensitivity of the solution $\hat{\mathbf{x}}$ from the experimentally computed condition number of R_{11} (see below) as a function of r

2) the condition number of R_{11} computed with MATLABs function `cond` as a function of r

3) the norm of the matrix E as a function of r

4) the norm of the residual $\mathbf{res} = A\hat{\mathbf{x}} - \mathbf{b}$ as a function of r

What is important in a practical case is to find the rank r for which both the residual **res** and the sensitivity of disturbancies in $\mathbf{b}$ are “small”. Often this is a compromize; the norm of the residual increases while the sensitivity (condition number) decreases as the rank r decreases.

Now to the lab. Download the MATLAB-file `illposed.m` from the homepage where lab 7 is mentioned. The file contains a MATLAB-function with the same name and creates an ill-conditioned system $A\mathbf{x} = \mathbf{b}$. Compute a system of size $n = 8$, i.e. A is 8×8 and $\mathbf{b}$ is 8×1.

a) Estimate the condition number $\kappa(A)$ experimentally from the relation

$$\frac{\|\Delta\mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \kappa(A)\frac{\|\Delta b\|_2}{\|\mathbf{b}\|_2}$$

by disturbing $\mathbf{b}$ with some small perturbances $\Delta\mathbf{b}$ and see how $\mathbf{x}$ is changed, $\Delta\mathbf{x}$. Compare your result with MATLAB’s library function `cond`.

b) Write a program showing a table of the following quantities as functions of the rank $r = 8, 7, 6, 5$: `norm(E)`, `cond(`$\mathtt{R_{11}}$`)`, the experimentally computed condition number of R_{11}, `norm(x)`, and `norm(res)`. Express your results in a way that describes the behaviour of the *numerical relations* above as functions of the rank r.

B. Least Squares with QR-factorization

The least squares method is used to fit functions to measured data. In the file `maunaloa8000.dat`, available from the homepage where lab 7 is mentioned, measurements of the CO_2-concentration is the athmosphere is collected. The concentration is measured in parts per million and is available in the file for each month from 1980 to 2003. The format of the file is as follows:

```
1980 338.01 338.36 340.08  ........................ 338.21
1981 339.23 340.47 341.38  ........................ 339.61
```

Hence each row starts with the year, then follows the 12 measurements for each month. Read this information to a program in MATLAB, e.g. in the following way

```
load maunaloa8000.dat -ascii
%pick out the monthly values (column 2:13) and store columnwise in Q
Q=maunaloa8000(:,2:13)';
b=Q(:); %all measurements are in time increasing order in one column
m=length(b);
t=[1:m]';
plot(t,b,'x')
```

A suitable function to fit to the given data could be

$$y(t) = c_1 + c_2e^{\alpha t} + \sum_{k=1}^{n}(a_k cos(\frac{2\pi kt}{12}) + b_k sin(\frac{2\pi kt}{12}))$$

where t is in months, $\alpha = 0.00037$ and the other parameters to be computed with the least squares method. Use the MATLAB function `qr` and solve the least squares problem for $n = 1, 2, 3$. Present the computed values of the parameters in each case. Compute also the residual and write out its 2-norm in the three cases. Did you get the smallest residual norm for $n = 3$? Make a graph of the model function $y(t)$ giving the smallest residual together with the x’s representing the measured values.