Lab Class (Least Squares Problems)

1 Least squares problems

1.1 Recall on least-square problems

A (linear) *least squares problem* is the problem of minimizing $\|Ax - b\|_2$ for a given tall thin matrix $A \in \mathbb{R}^{m \times n}$, $m > n$, and a vector $b \in \mathbb{R}^m$.

Recall that during the lecture we have seen two methods to solve least-square problems:

Normal equations Solve the linear system $(A^T A)x = A^T b$.

QR factorization Compute the QR factorization

$$A = QR, \quad Q = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix}, \quad R = \begin{bmatrix} R_1 \\ 0 \end{bmatrix},$$

with $Q_1 \in \mathbb{R}^{m \times n}$ and $R_1 \in \mathbb{R}^{n \times n}$, and solve $R_1 x = Q_1^T b$.

1.2 Salary estimation

The CSV file `salaries.csv` published on the course's webpage contains the total number of rebounds taken, fouls committed and points scored by 399 NBA players during the season 2015-2016, and the salaries guaranteed by their contract for season 2016-2017 (source: `basketball-reference.com`). As in the example seen in the lecture, we want to find the most appropriate coefficients for the model (salary) $\approx x_1$(rebounds) $+ x_2$(fouls) $+ x_3$(points), by minimizing

$$\sum_{p \in \text{players}} \Big(x_1 (\text{rebounds})_p + x_2 (\text{fouls})_p + x_3 (\text{points})_p - (\text{salary})_p \Big)^2$$

We expect x_1 and x_3 to be positive and x_2 to be negative, intuitively.

1. Load the numerical data in the file in Matlab with

   ```
   M = dlmread('salaries_formatted.csv', ',', 1, 1)
   ```

   The two last parameters indicate that we want to skip the first row (headers) and the first column (player names).

   Form the matrices A, b associated to this problem. Which entries go in which column of the matrix?

2. Compute using normal equations the solution to the least-square problem.

3. Compute `[Q, R] = qr(A)` and use these factors to compute the solution using the QR method. Are the two approximations equal? What is the order of magnitude of their difference?

4. Compute the relative residual $\frac{\|b-Ax\|}{\|b\|}$. Do you expect this quantity to be small like it was for linear systems?

5. Using our model, can you find out who are the most overpaid and the most underpaid player, i.e., those for which the difference $b - Ax$ between the actual and expected salary is higher/lower?

1.3 Data fitting

In this section, we want to use least square problem to construct a polynomial curve that fits given data.

1. Generate a sorted vector t of 1000 random numbers in $[-10, 10]$ with the command

   ```
   t = sort(20*rand(1000,1) - 10);
   ```

   Generate a vector y such that $y_i \approx 0.02t_i^3 - t_i + 1$ for each i, using the command

   ```
   sigma = 0.4
   y = 0.02*t.^3 - t + 1 + sigma*randn(1000,1);
   ```

   Plot the result with `plot(t, y, '.')`.

   (Recall that `rand` returns numbers in $[0, 1]$, while `randn` returns numbers drawn from a Gaussian distribution with mean 0 and variance 1.)

2. We now look for the unknown coefficients a, b, c, d so that $f(t) = at^3 + bt^2 + ct + d$ better approximates the data points that we have generated, in the sense that

   $$\sum_{i=1}^{1000} (at_i^3 + bt_i^2 + ct_i + d - y_i)^2 \tag{1}$$

   is minimized. Construct the matrix

   $$A = \begin{bmatrix} t_1^3 & t_1^2 & t_1 & 1 \\ t_2^3 & t_2^2 & t_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ t_{1000}^3 & t_{1000}^2 & t_{1000} & 1 \end{bmatrix},$$

   and convince yourself that (1) can be written as $\|Ax - y\|^2$, with $x = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$.

3. Find the solution x of the least squares problem $\min\|Ax - y\|^2$ using each of the two methods described above. This time, use Matlab's function `[Q1, R1]= qr(A, 0)`, which returns directly the factors Q_1, R_1 (*thin QR decomposition*).

4. Plot the sets of points (t_i, y_i) and $(t_i, (Ax)_i)$ onto the same graph with

```
plot(t, y, '.', t, A*x);
```

   What does the second set of data points represent?

5. How do the plots change if you alter the value of the 'noise' parameter σ? What happens at the ratio $\frac{\|b-Ax\|}{\|b\|}$?

2 QR factorization of Hessenberg matrices

Remember that a *(upper) Hessenberg matrix* is a matrix $A \in \mathbb{C}^{n\times n}$ such that $A_{ij} = 0$ whenever $i > j + 1$; i.e., it is almost like an upper triangular matrix, but it has one more nonzero diagonal; for instance, for $n = 5$,

$$\begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{bmatrix}.$$

1. Consider the first step of the QR factorization of a Hessenberg matrix A using Householder matrices. Show that the vector u_1 which defines the first Householder reflector H_1 has only two nonzero entries.

2. Show that, if u_1 has the special form found in the previous point, then the matrix-vector product $u_1^T A$ needed in the first step of the algorithm can be performed in $O(n)$. Moreover, replacing A with $H_1 A$ can be done in time $O(n)$ by updating only $2n$ elements of the matrix (which ones?).

3. Show that the same properties hold for the Householder reflectors needed at the next steps of the algorithm: $u_2, u_3, \ldots, u_{n-1}$ and H_2, H_3, H_{n-1}.

4. Use this fact to build a specialized algorithm to compute the QR factorization of a Hessenberg matrix using only $O(n^2)$ operations.

5. You can generate a random Hessenberg matrix with `H = triu(randn(n, n), -1)`. Generate one, for instance for $n = 5$, and use it to test the algorithm that you have implemented.

3 Extra exercises

If you still have time, you can try one of these ones.

- Experiment with the stability of linear least squares problems. Construct ill-conditioned random matrices with the following instructions

```
[Q, R] = qr(randn(50, 30));
R = triu(randn(50, 30));
A = Q * R;
b = randn(50, 1);
C = A + 1e-14*randn(size(A))*norm(A);
```

  As you can see, C is a tiny perturbation of A. How much do the solution of the least squares problems $\min\|Ax - b\|$ and $\min\|Cy - b\|$ differ (in terms of their relative error, $\frac{\|y-x\|}{\|x\|}$)? Is one of the two algorithms that we have used more sensitive than the other to small perturbations?

- We have seen that $\frac{\|b-Ax\|}{\|b\|}$ is not small for a least-squares problems, in general. What would you use instead as a residual measure?

- Can you modify the QR factorization function that you have written so that it returns the thin QR factors $Q_1 \in \mathbb{C}^{m\times n}$, $R_1 \in \mathbb{C}^{n\times n}$ instead of the full one? Your algorithm should require only $O(mn)$ location of memory, not $O(m^2)$ — in particular, you cannot construct the full QR factorization and then truncate the matrices.