Numerical Analysis Hilary Term 2025
Lecture 4: Least-squares problem

Recall from last lecture that we are interested in solving the *least-squares problem*:

$$\min_{x\in\mathbb{R}^n} \|Ax-b\|, \qquad A\in\mathbb{R}^{m\times n}, b\in\mathbb{R}^m, m\geq n, \tag{1}$$

where $\|y\| := \sqrt{y_1^2+y_2^2+\cdots+y_m^2} = \sqrt{y^\top y}$, and such problems (also known as *overdetermined* systems) are ubiquitous in applied mathematics and data science; linear regression is a basic example.

Solution of least-squares by the QR factorisation

Recall from last lecture that every matrix admits a QR factorisation:

$$A = Q_F R_F = \begin{bmatrix} Q & Q_\perp \end{bmatrix} \begin{bmatrix} R \\ 0 \end{bmatrix},$$

where $Q_F R_F$ is the full QR factorisation with $Q_F\in\mathbb{R}^{m\times m}$ and $R_F\in\mathbb{R}^{m\times n}$ and $A=QR$ is the thin QR factorisation with $Q\in\mathbb{R}^{m\times n}$ and $R\in\mathbb{R}^{n\times n}$. In particular, Q_F is orthogonal an R is upper triangular.

Recall that $\|Q_F^\top y\| = \|y\|$ for all $y\in\mathbb{R}^m$, and so

$$\|Ax-b\|^2 = \|Q_F^\top(Ax-b)\|^2 = \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} x - \begin{bmatrix} Q^\top b \\ Q_\perp^\top b \end{bmatrix} \right\|^2 = \|Rx - Q^\top b\|^2 + \|Q_\perp^\top b\|^2.$$

The second term does not involve x, so we have that the solution vector x satisfies

$$\arg\min_{x\in\mathbb{R}^n} \|Ax-b\|^2 = \arg\min_{x\in\mathbb{R}^n} \|Rx-Q^\top b\|^2.$$

For now, assume that A has full rank, which then implies that R is invertible. Then,

$$x = R^{-1}Q^\top b$$

is a solution to (1) since for any $y\in\mathbb{R}^n$, there holds

$$\|Ax-b\|^2 = \|Q_\perp^\top b\|^2 \leq \|Ry-Q^\top b\|^2 + \|Q_\perp^\top b\|^2 = \|Ay-b\|^2 \implies \|Ax-b\| \leq \|Ay-b\|.$$

It remains to show that the solution is unique. Suppose that $y\in\mathbb{R}^n$ is another solution; i.e. $\|Ay-b\| = \|Ax-b\|$. Then,

$$0 = \|Ay-b\|^2 - \|Ax-b\|^2 = \|Ry-Q^\top b\|^2,$$

and so $y = R^{-1}Q^\top b = x$. We have proved the following result:

Theorem. Let $A\in\mathbb{R}^{m\times n}$ and $b\in\mathbb{R}^n$ with $m\geq n$, and suppose that A has full column rank. Then, the least-squares problem (1) is uniquely solvable, and the solution is given by

$$x = R^{-1}Q^\top b,$$

where $A=QR$ is the thin QR factorisation of A.

Notes:

- The arguments we used suggest an algorithm: compute the "thin" QR factorization $A = QR$, then solve $Rx = Q^\top b$ for x, which is obtained by backward substitution as R is upper triangular.
- We only used the full QR for the derivation.
- The only properties of the QR factorisation that we used were that Q is orthogonal and that R is invertible.
- In a later lecture, will see that a general linear least-squares problem has solution characterised by the orthogonality condition, which in our context reduces to $A^\top(Ax-b) = 0$, so $x = (A^\top A)^{-1}A^\top b$; one can verify this is the same as $R^{-1}Q^\top b$ obtained above.

A is rank-deficient (non-examinable): If A does not have full rank (i.e. A is rank-deficient), then R will not be invertible. Nevertheless, one can still find a solution to the least-squares problem (1), but solutions will not be unique.

Underdetermined case

One might wonder, what if we have *fewer* equations than variables? That is, we wish to solve $Ax = b$ with $A \in \mathbb{R}^{m\times n}$, $m < n$. This *underdetermined* system of equations has infinitely many solutions (if there is one). The natural question becomes, which one should we look for? One possibility is to find the minimum-norm solution minimize $\|x\|$ subject to $Ax = b$, which as connections to the hot topic of *deep learning*.

More precisely, assume that A has full rank and that the set $\mathcal{K}(b) := \{x \in \mathbb{R}^n : Ax = b\}$ is non-empty. Then, the least-squares problem reads

$$\min_{x\in\mathcal{K}(b)} \|x\|. \tag{2}$$

To solve (2), we want to perform similar manipulations to the over-determined case. The first step above was to say that $\|Ax - b\| = \|Q_F^\top(Ax - b)\|$, and so we want to do the same here.

If we begin with the full QR factorisation of A, then the factor Q_F has dimension $m\times m$, which cannot be used to multiply $x \in \mathbb{R}^n$. Instead, we take the full QR factorisation of $A^\top$:

$$A^\top = Q_F R_F = \begin{bmatrix} Q & Q_\perp \end{bmatrix} \begin{bmatrix} R \\ 0 \end{bmatrix},$$

where $Q_F \in \mathbb{R}^{n\times n}$, $Q \in \mathbb{R}^{n\times m}$, and $R \in \mathbb{R}^{m\times m}$. Then, we can write

$$x = Q_F w = \begin{bmatrix} Q & Q_\perp \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

for some $w \in \mathbb{R}^n$ with two components: $u \in \mathbb{R}^m$ and $v \in \mathbb{R}^{n-m}$. By orthogonality, we have

$$\|x\|^2 = \|Q_F^\top x\|^2 = \|w\|^2 = \|u\|^2 + \|v\|^2.$$

We now need to incorporate the constraint that $x \in \mathcal{K}(b)$; i.e. $Ax = b$. Note that $A = R_F^\top Q_F^\top$, and so we can write $Ax = b$ as

$$b = R_F^\top Q_F^\top Q_F w = R_F^\top w = \begin{bmatrix} R^\top & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = R^\top u,$$

where we used that Q_F is orthogonal. Since A has full rank, $A^\top$ has full rank, and so R and $R^\top$ are invertible. Consequently, the condition $Ax = b$ is equivalent to

$$u = R^{-\top} b.$$

Putting things together, the least-squares problem (2) then reads

$$\min_{x \in \mathcal{K}(b)} \|x\|^2 = \min_{\substack{u \in \mathbb{R}^m \\ v \in \mathbb{R}^{n-m} \\ u = R^{-\top} b}} \left(\|u\|^2 + \|v\|^2\right) = \|R^{-\top} b\|^2 + \min_{v \in \mathbb{R}^{n-m}} \|v\|^2.$$

Clearly, the minimum is achieved for $v = 0$, and so we have $x = Qu = QR^{-\top} b$. We leave the uniqueness of solutions as an exercise. In summary, we have the following result:

Theorem. Let $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^n$ with $m < n$, and suppose that A has full rank. Then, the least-squares problem (2) is uniquely solvable, and the solution is given by

$$x = QR^{-\top} b,$$

where $A^\top = QR$ is the thin QR factorisation of $A^\top$.

Notes (non-examinable):

- If A is rank deficient, then the solution to the least-squares problem (2) still exists and, unlike the over-determined case, is still unique. In this case, the solution is more easily expressed in terms of the SVD decomposition of A.
- In (2), we minimised the 2-norm. Another fascinating approach that has had enormous impact is to minimise the 1-norm $\|x\|_1$ subject to $Ax = b$, where $\|x\|_1 = \sum_{i=1}^n |x_i|$. It turns out that the solution x then tends to be sparse, i.e., most of its entries are 0. This is the basis of the exciting field of *compressed sensing*.

Illustration of least-squares for polynomial approximation (non-examinable)

We treated Lagrange interpolation in Lecture 1. While Lagrange polynomials give a clean expression for the interpolating polynomial, the interpolating polynomial is not always a good approximation to the original underlying function f. For example, suppose $f(x) = 1/(25x^2 + 1)$ (this is a famous function called the *Runge function*), and take a degree-n polynomial interpolant p_n at $n+1$ equispaced points in $[-1, 1]$. The interpolating polynomials for varying n are shown in Figure 1.

As we increase n, we hope that $p_n \to f$—but this is far from the truth! p_n is diverging as n grows near the endpoints ± 1, and the divergence is actually exponential (very bad); note the vertical scales of the final plots! This is called Runge's phenomenon.

How can we avoid the divergence, and get $p_n \to f$ as we hope? One approach is to *oversample*: take (many) more points than the degree n. With $m(> n+1)$ data

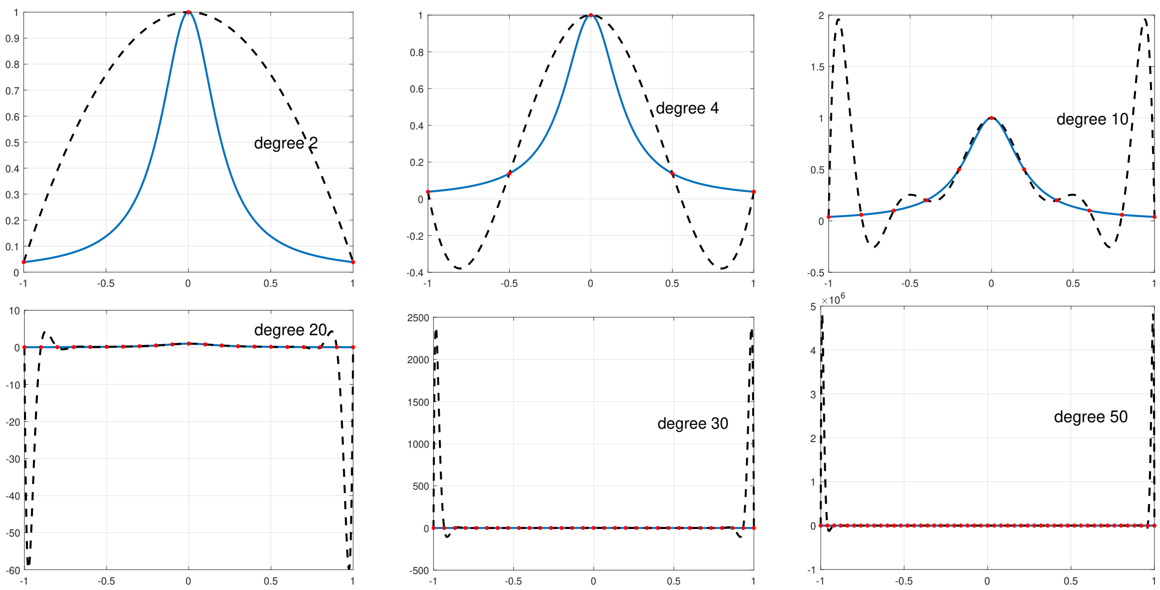

Figure 1: Polynomial interpolants (dashed black curves) of $f(x) = 1/(25x^2 + 1)$ (blue). The red dots are the interpolation points.

points $x_1, \ldots, x_m$, this will lead to the least-squares problem $\min_c \|Ac - b\|$, wherein $c = [c_0, c_1, \ldots, c_n]^\top$ represents the coefficients of the polynomial $p_n(x) = \sum_{j=0}^{n} c_j x^j$, $A \in \mathbb{R}^{m\times(n+1)}$ with $A_{ij} = (x_i)^{j-1}$ and $b = [f(x_1), \ldots, f(x_m)]^\top$.

We illustrate this in Figure 2 with the example above, but now fixing $n = 20$ and varying the number of data points m. This time, for large enough m the polynomial p_n is close to f across the whole interval $[-1, 1]$.

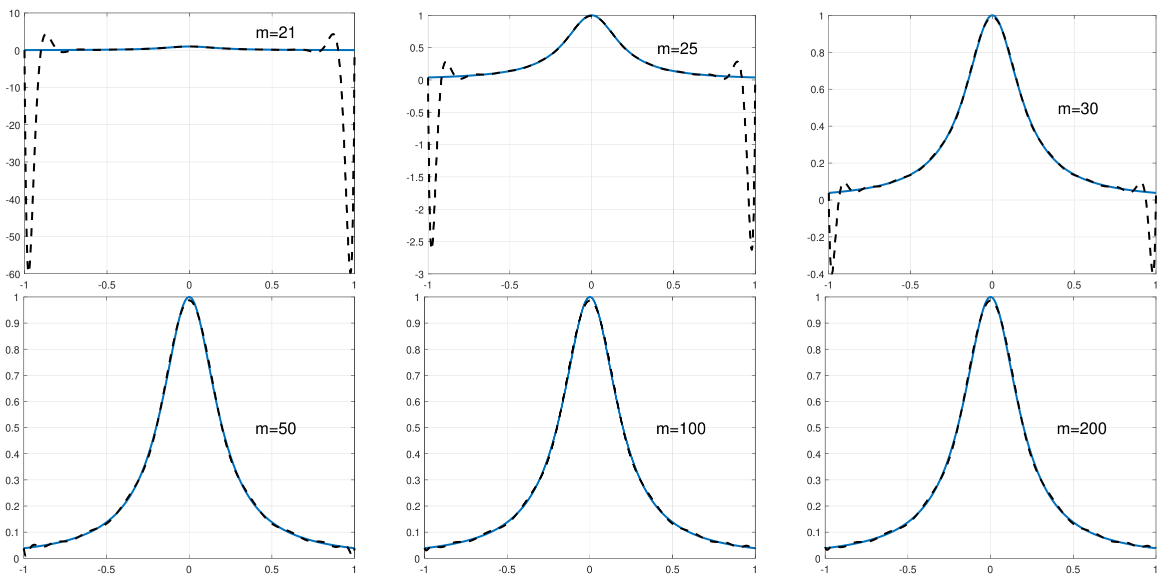

Figure 2: Least-squares polynomial fits of degree 20 (black dashed curves) of $f(x) = 1/(25x^2+1)$ (blue).

Extensions and related facts (Non-examinable)

- Instead of $p_n(x) = \sum_{j=0}^{n} c_j x^j$, it is actually much better to use a different polynomial basis involving *orthogonal polynomials* $\{\phi_i\}_{i=0}^{n}$ such as the Chebyshev polynomials, a topic discussed later. Then we would express $p_n(x) = \sum_{j=0}^{n} c_j \phi_j(x)$ and $A_{ij} = (\phi_{j-1}(x_i))$, and the least-squares problem will be beter-conditioned (easier to solve accurately). However, Runge's phenomenon still persists unless $m \gg n$.

- Note that we do not have $p_n \to f$ in Figure 2 as $m \to \infty$ because the polynomial degree $n = 20$ is fixed; to get $p_n \to f$ one needs to increase n together with m. It can be shown that if one takes $m = n^2$, we do have $p_n \to f$ for any analytic function f (the convergence is exponential in n).

- Another—more elegant—solution to overcome the instability in Figure 1 is to change the interpolation points. If one chooses them to be the so-called Chebyshev points $x_j = \cos(j\pi/n)$ for $j = 0, 1, \ldots, n$, the interpolating polynomial can be shown to be an excellent approximation to f, in fact nearly the best-possible polynomial approximation for any continuous f. This is a fundamental fact in approximation theory; for a rigorous and extended discussions (including an explanation of Runge's phenomenon), check out the Part C course Approximation of Functions.