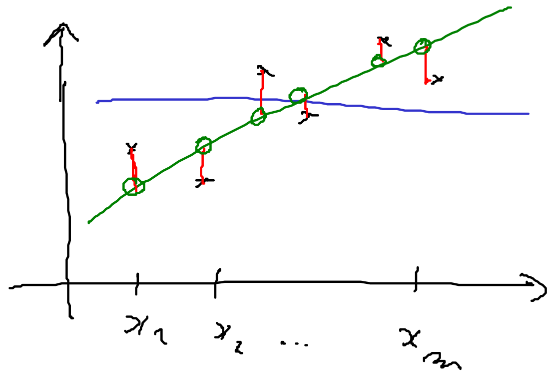
Least Squares Problems



measurements

$y_1, y_2, \dots, y_m \in \mathbb{R}$

corresponds to

$\underline{x}_1, \underline{x}_2, \dots \underline{x}_m \in \mathbb{R}^n$

Model (assumption): linear dependence between y and x:

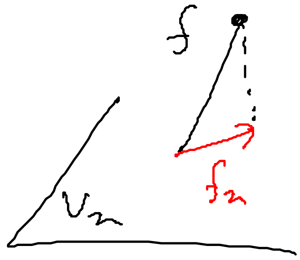
$$y = \underline{a}^T \underline{x} + \underset{\substack{\cap \\ \mathbb{R}}}{c} \qquad \underline{a} \in \mathbb{R}^n$$

Question: given measurements, what are $\underline{a}, \underline{c}$?
(parameters of the model.

least squares solution is

$$\min_{\substack{\underline{p} \in \mathbb{R}^n \\ q \in \mathbb{R}}} \sum_{i=1}^{m} \left| y_i - \underline{p}^T \underline{x}_i - q \right|^2$$

Ex Want an approximation of function



$$f \in V$$
$$\approx \qquad \cup$$
$$f_n \in V_n \text{ of dimension } n < \infty$$
$$\Downarrow$$

$b_1, b_2, \dots b_n$ Basis in V_n

$$f_n(t) = \sum_{i=1}^{n} x_i \, b_i(t)$$

Example $b_i(t) = t^i$ polynomial of degree i

f complicated or even not know, but we evaluate /measure $y_i \approx f(t_i)$

Have: $(t_i, y_i) \in \mathbb{R}^2$, $i = 1, 2, \dots, m$

Find coefficients $x_1, \dots, x_m$ such that

$$\underbrace{\sum_{i=1}^{m} \left| f_n(t_i) - y_i \right|^2}_{\| f_n(\underline{t}) - \underline{y} \|_2^2} = \min!$$

again a (linear) least squares problem!

matlab: polyfit !!

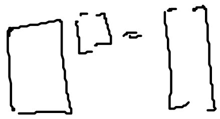
(LSP) linear least squares problem:

Given $\underline{\underline{A}} \in \mathbb{C}^{m \times n}$, $\underline{b} \in \mathbb{C}^m$, find $\underline{x} \in \mathbb{C}^n$

such that

$$\| \underline{\underline{A}}\underline{x} - \underline{b} \|_2 = \inf \{ \| \underline{\underline{A}}\underline{y} - \underline{b} \|, \ y \in \mathbb{C}^n \}$$

another way: $\underline{\underline{A}}\underline{x} = \underline{b}$ define sol. $\| \underbrace{Ax - b}_{\text{residual } r} \|_2^2 = \min!$



∃ unique solution, function in matlab used to compute pseudo inverse of a matrix A

$A^+ \in \mathbb{C}^{n \times m}$

$y \mapsto x = \text{sol of (LSP)}$

$Ax = b$ $\qquad \begin{bmatrix} A \end{bmatrix} [x] = [b]$

A^T

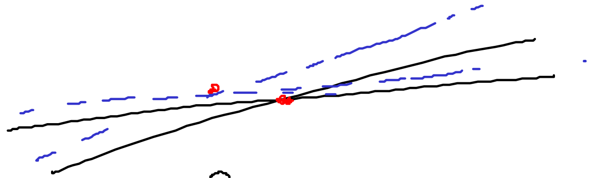
$\Rightarrow$

$\boxed{A^T A x} = A^T b \Rightarrow$ compute x via $(A^TA) \backslash b$

quadratic, symmetric, pos. def $\Rightarrow$ invertible

this way of solving is called ~~the~~ normal equation

instability: bad conditioning of A

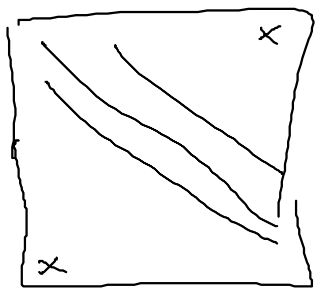


1) $\text{cond}(A^T A) = (\text{cond } A)^2$

$\text{cond } A = 10^3 \Rightarrow \text{cond}(A^T A) = 10^6$

$\text{cond } A = \frac{\sigma_1}{\sigma_n}$

2) typically $Ax = b$ from discretizations of partial diff. equations $\Rightarrow$ $\underline{A}$ large, sparse matrix



$A^T A \Rightarrow$ not anymore sparse!

$\Rightarrow A^T A x = b$ might be very expensive.

Note Trick to be used when A is sparse:

$r = Ax - b$; $\|r\|$ = measure of the validity of the linear model

ideally: $r = 0$; practically normal equation requires $A^T r = 0$

$$\begin{cases} A^T A x = A^T b \\ -r + Ax = b \end{cases} \iff \underbrace{\begin{bmatrix} -I & A \\ A^T & 0 \end{bmatrix}}_{B} \begin{bmatrix} r \\ x \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}$$

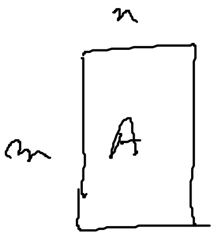
A sparse $\Rightarrow$ B sparse

What about the condition of this system?

$r := \frac{1}{\alpha}(Ax - b)$, parameter $\alpha > 0$ free

choose α such that $\mathrm{cond}(B_\alpha)$ is small!

When A is not sparse:



*) rank $A = n$

$$\|Ax - b\|_2 = \|QRx - b\|_2 = \|Q(Rx - Q^H b)\|_2 = \|Rx - \underbrace{Q^H b}_{\tilde{b}}\|_2 = \|Rx - \tilde{b}\|_2$$

$$\min \left\| \begin{bmatrix} \diagdown \\ \hline 0 \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} - \begin{bmatrix} \tilde{b}_1 \\ \vdots \\ \tilde{b}_n \\ \hline \tilde{b}_{n+1} \\ \vdots \\ \tilde{b}_m \end{bmatrix} \right\| \Rightarrow \text{choose } x \text{ such that } \begin{bmatrix} \diagdown \\ 0 \end{bmatrix} x = \begin{bmatrix} \tilde{b}_1 \\ \vdots \\ \tilde{b}_n \end{bmatrix}$$

$\Rightarrow x$ unique, easy computed via backward substitution

Residual: $\quad r = Q \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \tilde{b}_{n+1} \\ \vdots \\ \tilde{b}_m \end{bmatrix}$

$$\|r\|_2^2 = |\tilde{b}_{n+1}|^2 + \dots + |\tilde{b}_m|^2$$

no round-off errors are introduced

price to pay: costs twice as long to do QR-decomposition than LU-decomposition

(*) rank $A = r \le \min(m,n)$ use SVD (matlab is doing)

$\sigma_1, \sigma_2, \dots, \sigma_r$

SVD:

$$A = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \left[\begin{array}{c|c} \Sigma_r & 0 \\ \hline 0 & 0 \end{array}\right] \begin{bmatrix} V_1^H \\ V_2^H \end{bmatrix}$$

$$\|Ax - b\|_2 = \left\| \begin{bmatrix} U_1 & U_2 \end{bmatrix} \left[\begin{array}{c|c} \Sigma_r & 0 \\ \hline 0 & 0 \end{array}\right] \begin{bmatrix} V_1^H \\ V_2^H \end{bmatrix} x - \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \right\|_2 =$$

$$= \left\| \begin{bmatrix} \Sigma_r V_1^H x \\ \hline 0 \end{bmatrix} - \begin{bmatrix} U_1^H b_1 \\ \hline U_2^H b_2 \end{bmatrix} \right\|_2$$

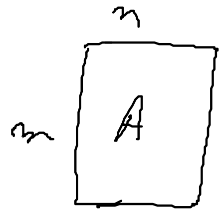
$\Rightarrow$ take $x = V_1 \Sigma_r^{-1} U_1^H b_1$

and have $\|r\|_2 = \|U_2^H b_2\|_2$

* expensive, no advantage from a structure of A can be taken

Constrained Least Squares Problems

$\underline{\underline{A}} \in \mathbb{R}^{m \times n}$, $m \geq n = \text{rank}(A)$



$\underline{b} \in \mathbb{R}^{m}$

$\underline{\underline{C}} \in \mathbb{R}^{p \times n}$, $n > p = \text{rank}(C)$

$\underline{d} \in \mathbb{R}^{p}$

Find $\underline{x} \in \mathbb{R}^{n}$ such that $\| \underline{\underline{A}}\,\underline{x} - \underline{b} \|_2 = \min$

and $\underline{\underline{C}}\,\underline{x} = \underline{d}$ linear constraint

1) use Lagrange-multipliers $\underline{m} \in \mathbb{R}^{p}$

$$x = \operatorname*{argmin}_{x \in \mathbb{R}^n} \max_{m \in \mathbb{R}^p} L(x, m)$$

$$L(x, m) = \frac{1}{2} \| Ax - b \|_2^2 + \underline{m}^T (\underline{\underline{C}}\,\underline{x} - \underline{d})$$

$$\frac{\partial L}{\partial x}(x, m) \overset{!}{=} 0 \Rightarrow \underline{\underline{A}}^T (\underline{\underline{A}}\,\underline{x} - \underline{b}) + \underline{\underline{C}}^T \underline{m} = 0 \Rightarrow$$

$$\frac{\partial L}{\partial m}(x, m) \overset{!}{=} 0 \Rightarrow \underline{\underline{C}}\,\underline{x} - \underline{d} = 0$$

normal equations

$$\begin{bmatrix} A^T A & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ m \end{bmatrix} = \begin{bmatrix} A^T b \\ d \end{bmatrix}$$

Trick

$$\begin{bmatrix} -I & A & 0 \\ A^T & 0 & C^T \\ 0 & C & 0 \end{bmatrix} \begin{bmatrix} r \\ x \\ m \end{bmatrix} = \begin{bmatrix} b \\ 0 \\ d \end{bmatrix}$$

usefull for sparse matrix A, C

2) singular value decomposition:

$$C = \underset{p \times p}{U} \begin{bmatrix} \Sigma & 0 \end{bmatrix} \begin{bmatrix} V_1^H \\ V_2^H \end{bmatrix} \begin{matrix} \}p \\ \}n-p \end{matrix}$$

$\operatorname{Ker} C = \operatorname{Im}(V_2) \Rightarrow$

define $\boxed{x_0 = V_1 \Sigma^{-1} U^H d}$

search for $\boxed{x = x_0 + V_2 y}$ with $y \in \mathbb{R}^{n-p}$

$$\| A(x_0 + V_2 y) - b \|_2 = \| A x_0 + A V_2 y - b \|_2 =$$

$$= \| A V_2 y - (b - A x_0) \|_2 \text{ standard LSP}$$

Nonlinear Least Squares Problems

Ex data: (t_i, y_i) for $i = 1, 2, \dots, m$ with errors

non-linear model.

$$y = f(t, x) = x_1 + x_2 e^{-x_3 t}$$

x: parameters, to be fitted to the measurements

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in \mathbb{R}^3$$

$$\text{(NlLSP)} \qquad x^* = \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \sum_{i=1}^{m} | f(t_i, x) - y_i |^2$$

$$F(\underline{x}) = \begin{bmatrix} f(t_1, \underline{x}) - y_1 \\ \vdots \\ f(t_m, \underline{x}) - y_m \end{bmatrix} \Rightarrow x^* = \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \boxed{\frac{1}{2} \| F(x) \|_2^2}$$

$$\frac{1}{2} \| F(x) \|_2^2 = \Phi(x)$$

Note not clear at all, if a solution exists
it is unique?

1) Newton-method:

$$\Phi(x^*) = \min \Rightarrow \operatorname{grad} \Phi(x^*) = 0$$

Nonlinear algebraic equation
$\Downarrow$
use Newton-Method for $\operatorname{grad} \Phi(x^*) = 0$

$$\operatorname{grad} \Phi(x) = \operatorname{grad}\left(\frac{1}{2} F(x)^T F(x)\right) = DF(x)^T F(x)$$

Newton-method for

$$\boxed{\text{find } x^* \text{ such that } \quad DF(x^*)^T F(x^*) = 0}$$

$$\underbrace{D\,\mathrm{grad}\,\Phi(x)}_{H\Phi(x)} = DF(x)^T DF(x) + \sum_{j=1}^{m} F_j(x)\, D^2 F_j(x)$$

$\hookrightarrow$ Hesse matrix of Φ

Newton:

$$x^{(k+1)} = x^{(k)} + s$$

with Newton update

$$H\Phi(x^{(k)})\, s = -\, DF(x^{(k)})^T F(x^{(k)})$$

start near $x^{(*)}$ $\Longrightarrow$ quadratic convergence

2) Gauss-Newton method

Idea: linearise locally $F(x) \approx F(y) + DF(y)(x-y)$

then solve a series of linear LSPs:

$$\underset{x \in \mathbb{R}^n}{\mathrm{argmin}} \|F(x)\|_2 \approx \underset{x \in \mathbb{R}^n}{\mathrm{argmin}} \| F(x_0) + DF(x_0) \underbrace{(x - x_0)}_{s} \| =$$

approximation of x^*

$$= \mathrm{argmin} \|Ax - b\|_2$$

$$A = DF(x_0), \quad b = F(x_0) - DF(x_0)x_0$$

fast!

$$x_0 \xmapsto{LSP} x_1 \xmapsto{LSP} x_2 \to \dots \xrightarrow{LSP} x_k$$

$$x^{(k+1)} = x^{(k)} + s_k$$

$$s_k = \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \| F(x^{(k)}) + DF(x^{(k)}) s \|_2$$

faster, but only linear convergence!

In both cases: starting value is very important

3) Levenberg-Marquardt-method

Gauss-Newton with penalization of large s and special choice of initial value

$$\min_s \{ \| F(x_k) + DF(x^{(k)}) s \|_2^2 + \lambda \|s\|_2^2 \}$$

with parameter $\lambda > 0$ clever chosen!