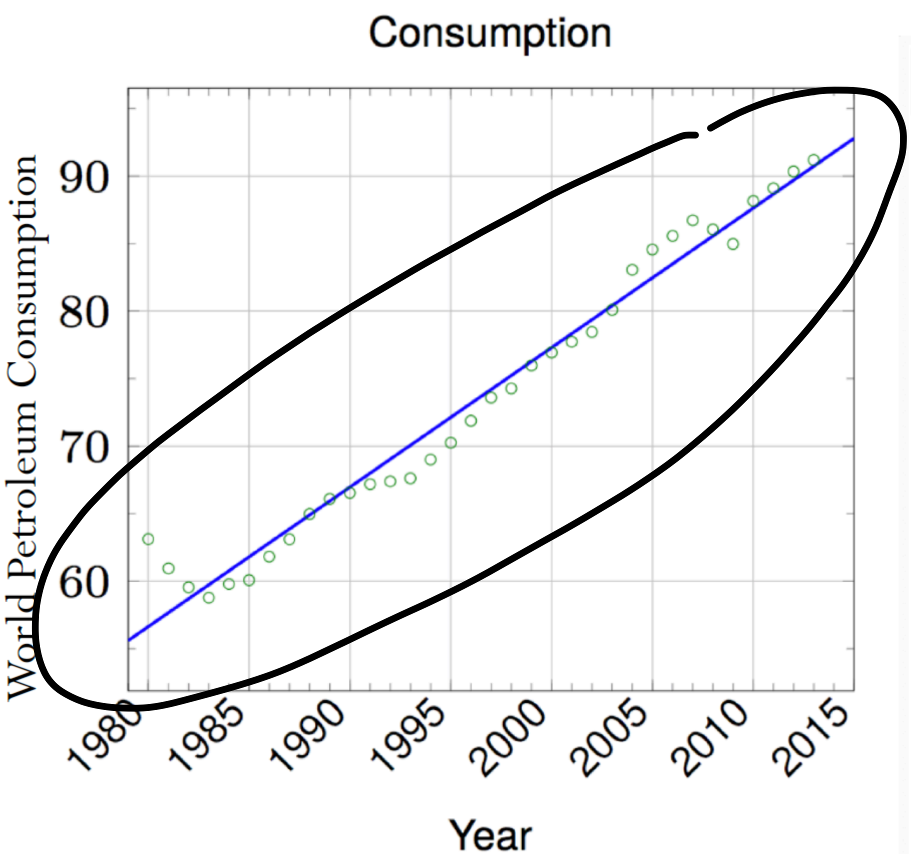
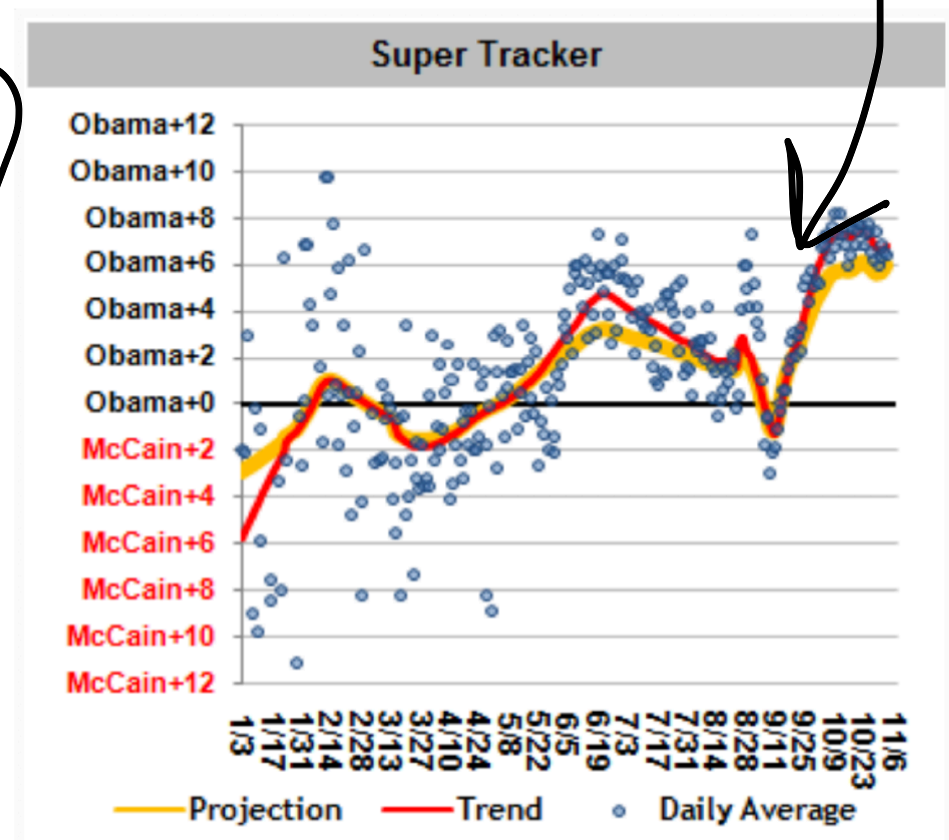
Least Squares and Data Fitting

Data fitting

How do we best fit a set of data points?





Linear Least Squares

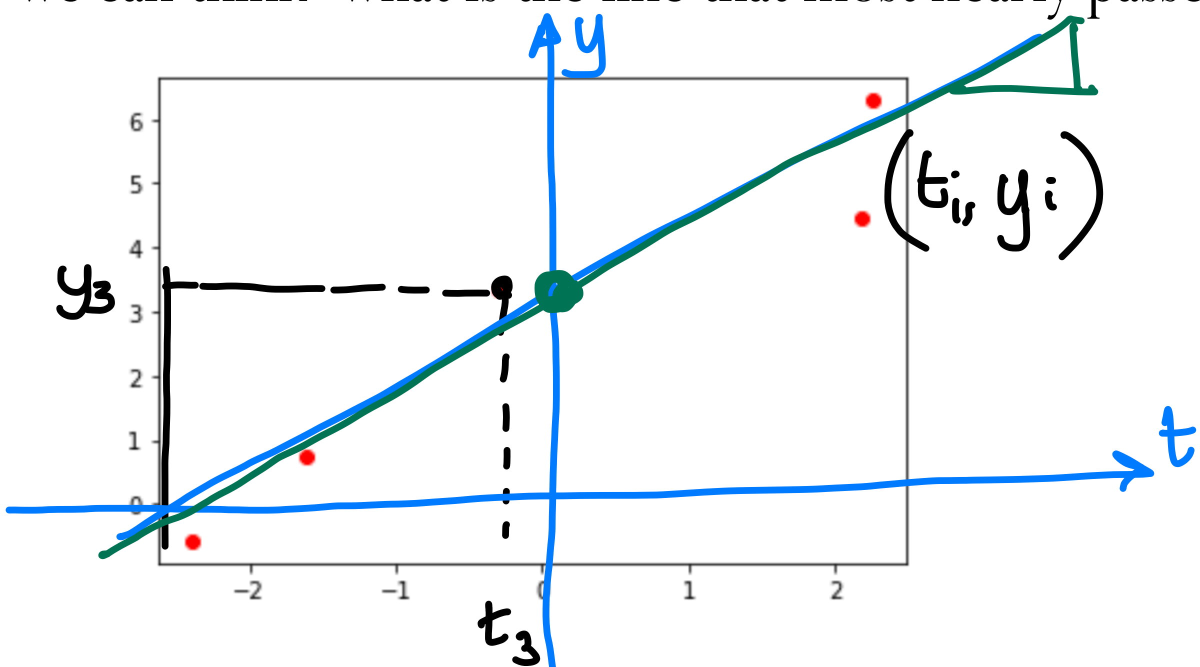
1) Fitting with a line

Given m data points $\{\{t_1, y_1\}, \ldots, \{t_m, y_m\}\}$, we want to find the function

$$y = x_o + x_1 t$$

that best fit the data (or better, we want to find the coefficients x_o, x_1).

Thinking geometrically, we can think "what is the line that most nearly passes through all the points?"



Given m data points $\{\{t_1, y_1\}, \dots, \{t_m, y_m\}\}$, we want to find x_o and x_1 such that

$$\boxed{y_i = x_o + x_1\, t_i} \qquad \forall i \in [1, m]$$

(t_1, y_1)

$$
\left.
\begin{aligned}
y_1 &= x_0 + x_1 t_1 \\
y_2 &= x_0 + x_1 t_2 \\
y_3 &= x_0 + x_1 t_3 \\
&\ \vdots \\
y_m &= x_0 + x_1 t_m
\end{aligned}
\right\}
\quad
\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}}_{\underset{\sim}{b}\ m\times 1}
=
\underbrace{\begin{bmatrix} 1 & t_1 \\ 1 & t_2 \\ \vdots & \\ 1 & t_m \end{bmatrix}}_{\underset{\approx}{A}\ m\times n}
\underbrace{\begin{bmatrix} x_0 \\ x_1 \end{bmatrix}}_{\underset{\sim}{x}\ n\times 1}
$$

$\underset{\sim}{b} = \underset{\approx}{A}\, \underset{\sim}{x}$

overdetermined! $\boxed{m > n}$

$$\underset{\sim}{b} = \underset{\approx}{A}\, \underset{\sim}{x}$$

given   given   find

Given m data points $\{\{t_1, y_1\}, \dots, \{t_m, y_m\}\}$, we want to find x_o and x_1 such that

$$y_i = x_o + x_1\, t_i \qquad \forall i \in [1, m]$$

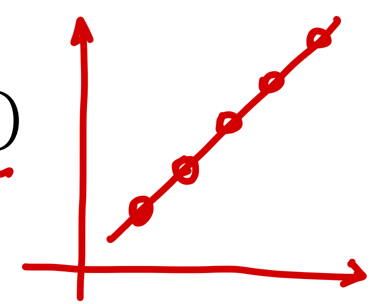
or in matrix form:

$$\underbrace{\begin{bmatrix} 1 & t_1 \\ \vdots & \vdots \\ 1 & t_m \end{bmatrix}}_{\boldsymbol{m\times n}} \underbrace{\begin{bmatrix} x_o \\ x_1 \end{bmatrix}}_{\boldsymbol{n\times 1}} = \underbrace{\begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}}_{\boldsymbol{m\times 1}} \qquad \boxed{\boldsymbol{A\,x = b}}$$

Note that this system of linear equations has more equations than unknowns – OVERDETERMINED SYSTEMS

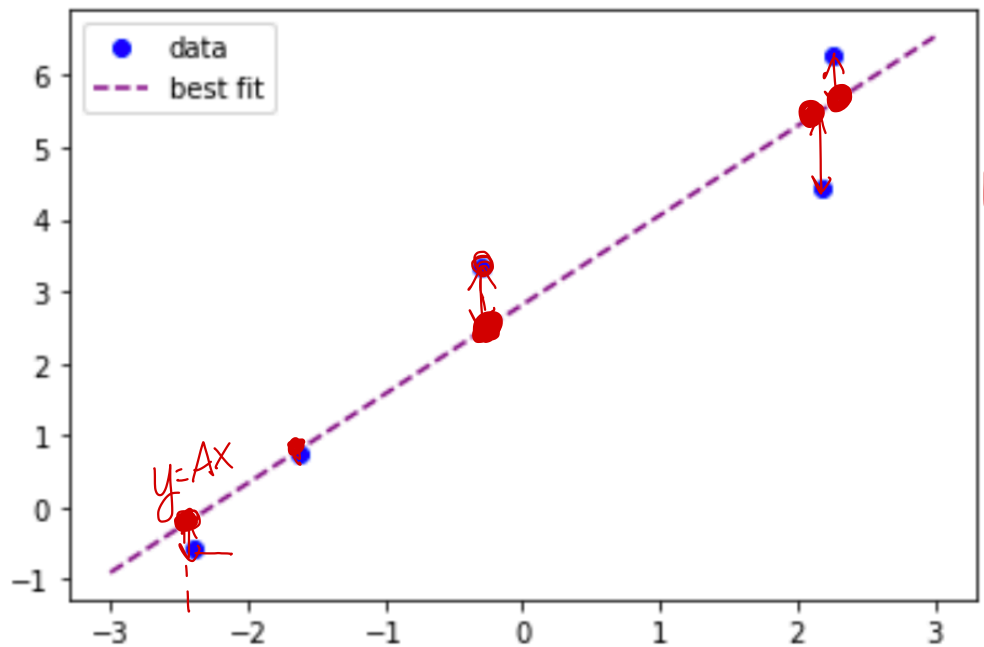
We want to find the appropriate linear combination of the columns of $\boldsymbol{A}$ that makes up the vector $\boldsymbol{b}$.

If a solution exists that satisfies $\boldsymbol{A\,x = b}$ then $\boldsymbol{b} \in range(\boldsymbol{A})$



Linear Least Squares

- In most cases, $\boldsymbol{b} \notin range(\boldsymbol{A})$ and $\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{b}$ **does not have an exact solution**!



we want to find $\underset{\sim}{x}$ s.t $\underset{=}{y} = \underset{=}{A}\underset{\sim}{x}$ better approximates $\underset{\sim}{b}$

- Therefore, an overdetermined system is better expressed as

$$\boxed{\boldsymbol{A}\,\boldsymbol{x} \cong \boldsymbol{b}} \qquad \boxed{m > n}$$

Linear Least Squares

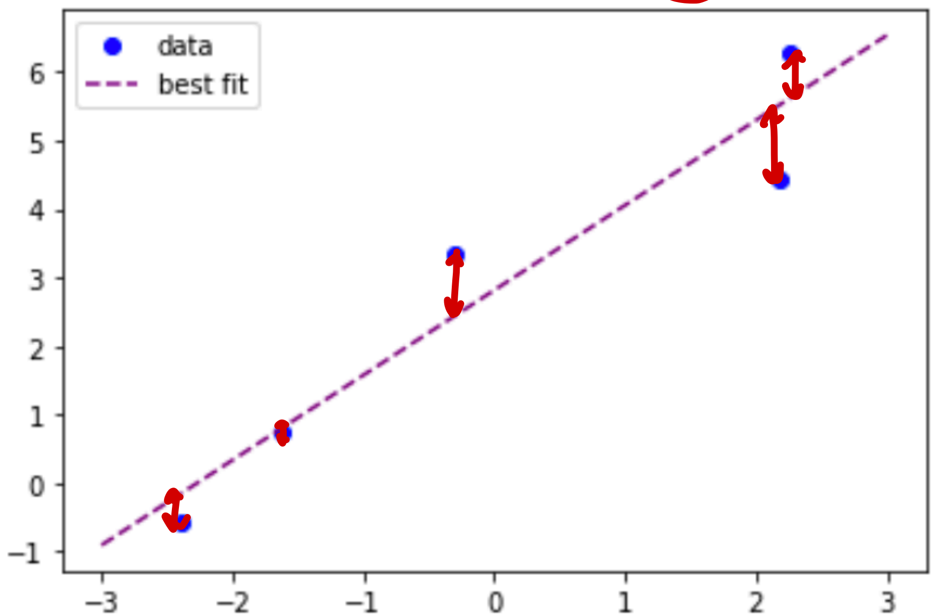
$Ax \cong b$

$= y - b$

- **Least Squares**: find the solution $\boxed{\boldsymbol{x}}$ that minimizes the residual

min

$$\boldsymbol{r} = \boldsymbol{b} - \boldsymbol{A}\,\boldsymbol{x} = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_5 \end{bmatrix} = \begin{bmatrix} b_1 - y_1 \\ b_2 - y_2 \\ \vdots \\ b_5 - y_5 \end{bmatrix}$$



r is a vector

min $\|r\|$

- Let's define the function ϕ as the square of the 2-norm of the residual

$$\phi(\boldsymbol{x}) = \|\boldsymbol{b} - \boldsymbol{A}\,\boldsymbol{x}\|_2^2$$

Linear Least Squares

- **Least Squares**: find the solution $\boldsymbol{x}$ that minimizes the residual

$$\boldsymbol{r} = \boldsymbol{b} - \boldsymbol{A}\,\boldsymbol{x}$$

- Let's define the function ϕ as the square of the 2-norm of the residual

$$\phi(\boldsymbol{x}) = \|\boldsymbol{b} - \boldsymbol{A}\,\boldsymbol{x}\|_2^2$$

- Then the least squares problem becomes

$$\min_{\boldsymbol{x}} \phi\,(\boldsymbol{x})$$

1d opt
1st order
2nd order

- Suppose $\phi: \mathcal{R}^m \to \mathcal{R}$ is a smooth function, then $\phi(\boldsymbol{x})$ reaches a (local) maximum or minimum at a point $\boldsymbol{x}^* \in \mathcal{R}^m$ only if

$$\boxed{\nabla\phi(\boldsymbol{x}^*) = 0}$$

How to find the minimizer?

stationary

$\Uparrow$

$\tilde{x}$

- To minimize the 2-norm of the residual vector

1st order necessary cond

$$\min_{x} \phi(x) = \|b - A\,x\|_2^2 = (b - A\,x)^T (b - A\,x)$$

$$\nabla \phi = -A^T(b - Ax) + (b - Ax)^T(-A)$$

$$= -A^Tb + A^TAx - A^Tb + A^TAx$$

$$= 2(A^TAx - A^Tb)$$

$$\nabla\phi = 0 \Rightarrow \underbrace{\underbrace{A^T}_{n\times m}\underbrace{A}_{m\times n}\underbrace{x}_{n\times 1}}_{n\times 1} = \underbrace{\underbrace{A^T}_{n\times m}\underbrace{b}_{m\times 1}}_{n\times 1}$$

$\nabla(\nabla\phi) = ?$

Find

$$\underbrace{A^TA}_{\text{given}}\,x = \underbrace{A^Tb}_{\text{given}}$$

Normal Equations

$\Downarrow$

linsys

$A^TAx = A^Tb$ → Normal equation

$\nabla\phi = 2(A^TAx - A^Tb)$

$\boxed{H = 2A^TA}$ → x is minimizer

x : solution?

*if matrix A is full rank:

$A = \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix}_{m\times n}$ $m > n$ → $\text{rank}(A) = n$

$\text{rank}(A) = n \Rightarrow$ A has n L.I. columns

$\Rightarrow$ has n singular values > 0

A^TA has n eigenvalues > 0

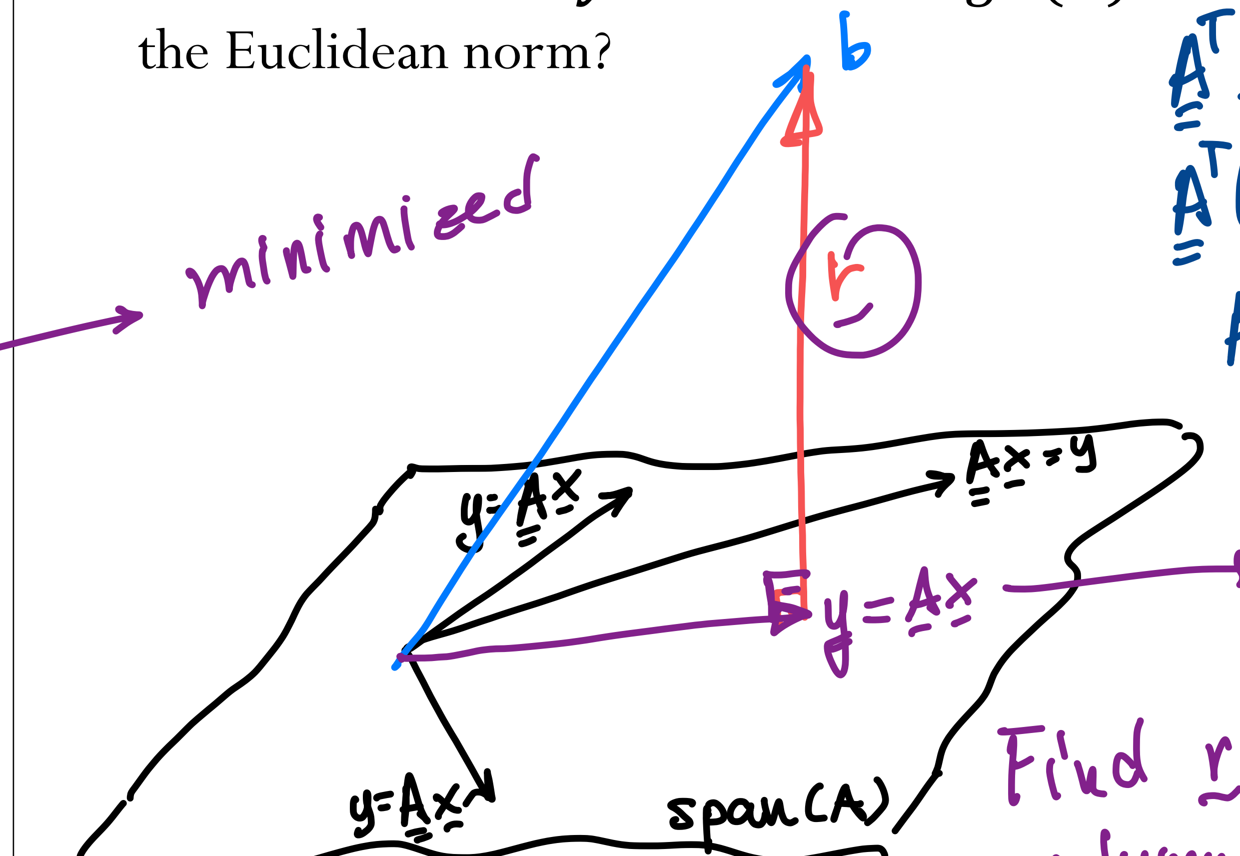
$y^TA^TAy > 0$ for any $y \neq 0 \Rightarrow$ A^TA is positive def. & symmetric

↳ A^TA is invertible

$\boxed{x = (A^TA)^{-1}A^Tb}$ ⇒ unique

Linear Least Squares (another approach)

- Find $\boldsymbol{y} = \boldsymbol{A}\,\boldsymbol{x}$ which is closest to the vector $\boldsymbol{b}$
- What is the vector $\boldsymbol{y} = \boldsymbol{A}\,\boldsymbol{x} \in range(\boldsymbol{A})$ that is closest to vector $\boldsymbol{y}$ in the Euclidean norm?



$$\underline{\underline{A}}^T \underline{r} = 0$$

$$\underline{\underline{A}}^T(\underline{b} - \underline{\underline{A}}\,\underline{x}) = 0$$

$$A^T b - A^T A x = 0$$

$$\boxed{A^T A x = A^T b}$$ ✓

projection of $\underline{b}$ in the column space of $\underline{\underline{A}}$

Find $\underline{r}$ orthogonal to $\underline{\underline{\text{ALL}}}$ columns of $\underline{\underline{A}}$.

$\Rightarrow \underline{\underline{A}}^T \underline{r} = 0$

Summary:

- $\boldsymbol{A}$ is a $m \times n$ matrix, where $m > n$.
- m is the number of data pair points. n is the number of parameters of the "best fit" function.
- Linear Least Squares problem $\boldsymbol{A}\,\boldsymbol{x} \cong \boldsymbol{b}$ ***always*** has solution.
- The Linear Least Squares solution $\boldsymbol{x}$ minimizes the square of the 2-norm of the residual:

$$\min_{\boldsymbol{x}} \|\boldsymbol{b} - \boldsymbol{A}\,\boldsymbol{x}\|_2^2$$

- One method to solve the minimization problem is to solve the system of **Normal Equations**

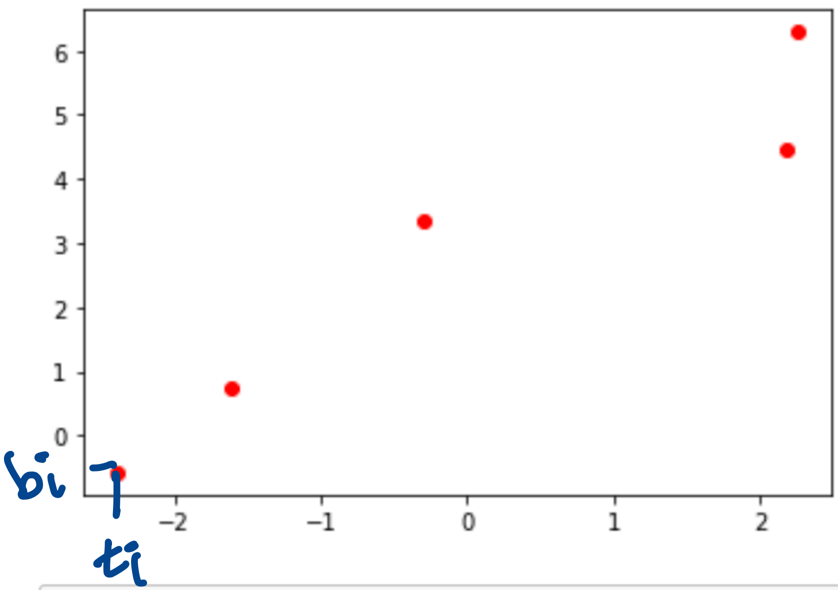
$$\boxed{\boldsymbol{A}^T\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{A}^T\,\boldsymbol{b}} \rightarrow$$

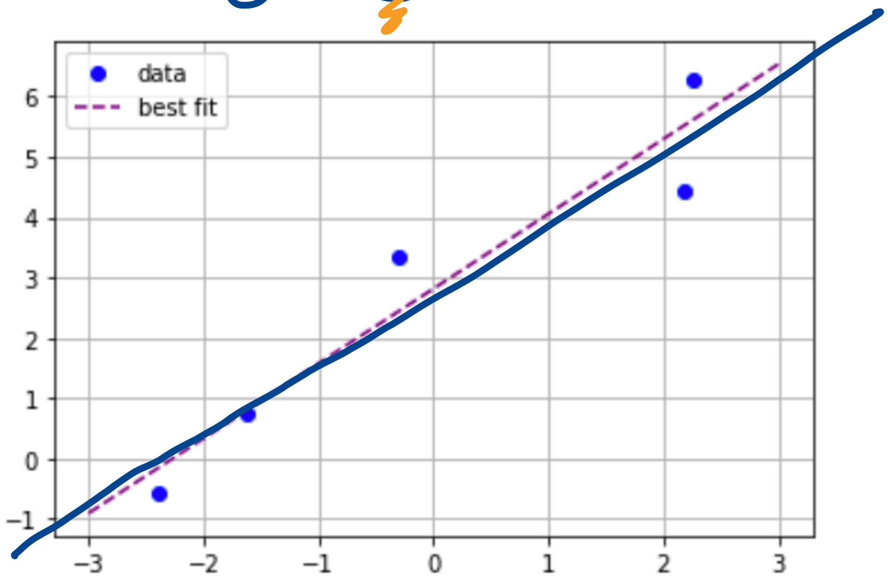
x is unique
matrix is full
rank

- Let's see some examples and discuss the limitations of this method.

Example:

$y = x_0 + x_1 t$






```
t
```

```
array([-1.61477467, -2.3970584 , -0.30372944,  2.26304537,  2.188127  ])
```

```
b
```

```
array([ 0.74112251, -0.57768693,  3.33523097,  6.29377547,  4.44786481])
```

Solve: $\boldsymbol{A}^T \boldsymbol{A}\, \boldsymbol{x} = \boldsymbol{A}^T\, \boldsymbol{b}$

$$\underset{5\times 2}{\underset{\sim}{A}} = \begin{bmatrix} 1 & t_1 \\ 1 & t_2 \\ \vdots & \vdots \\ 1 & t_5 \end{bmatrix} \qquad \underset{\sim}{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_5 \end{bmatrix}$$

```
x
```

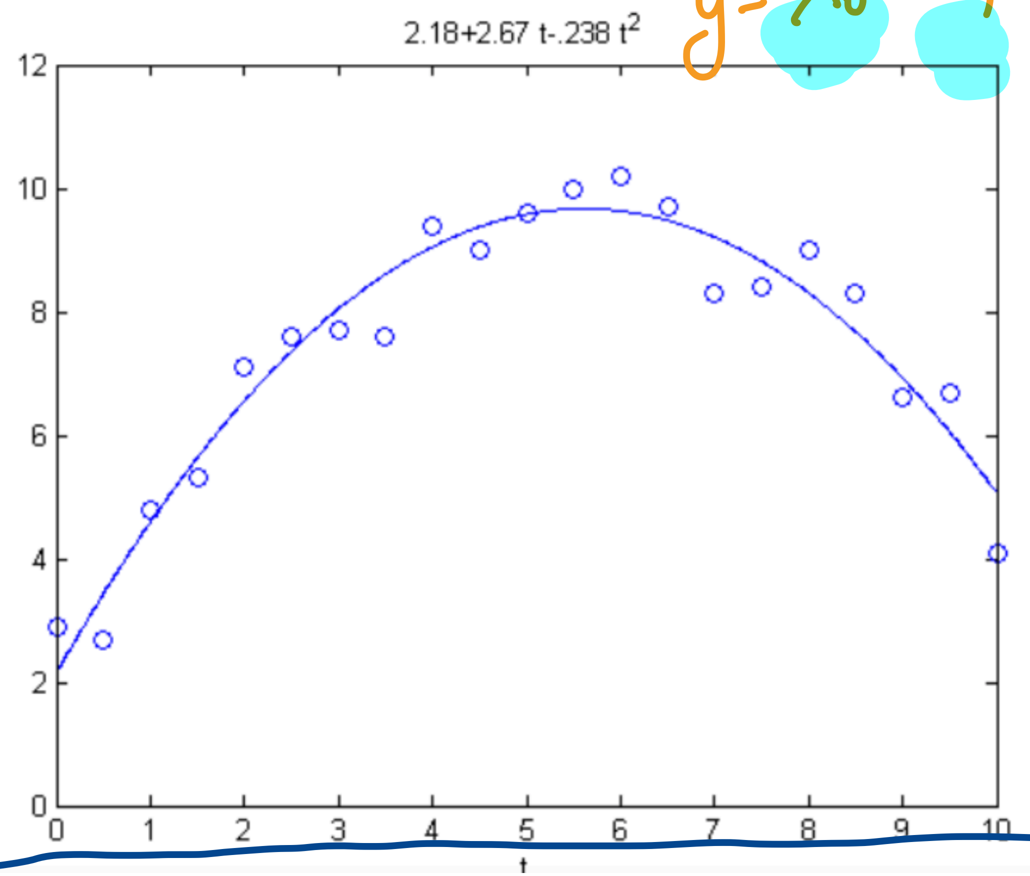
x = la.solve($A^T A$, $A^T b$)

```
array([2.81441707, 1.24048133])
```

Data fitting - not always a line fit!

- Does not need to be a line! For example, here we are fitting the data using a quadratic curve.

$$y = x_0 + x_1 t + x_2 t^2$$



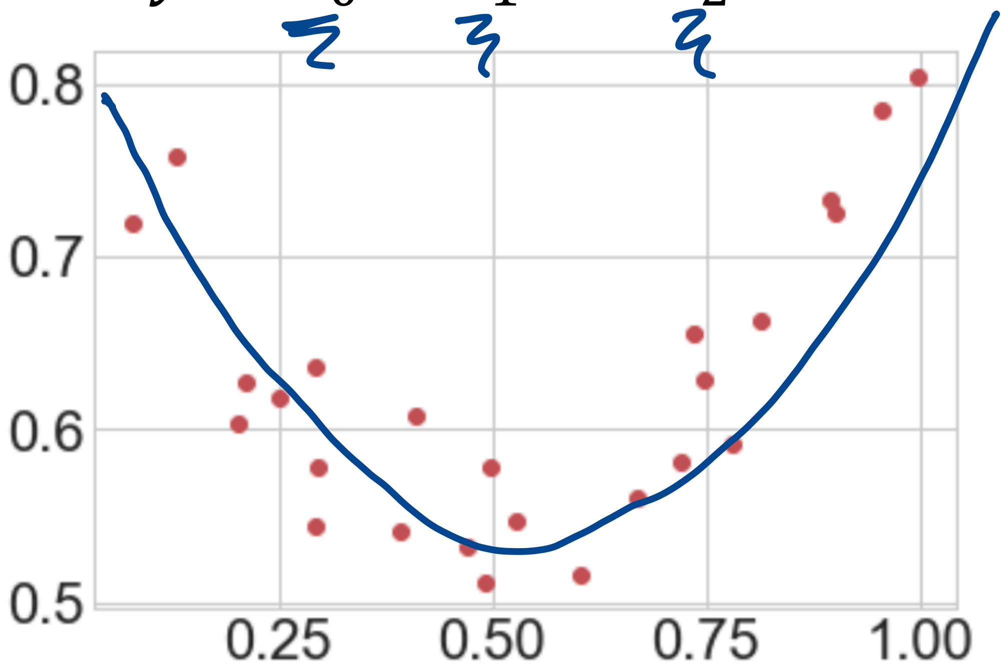
$m \times 3$

<u>Linear</u> Least Squares: The problem is **linear in its coefficients!**

Another example

We want to find the coefficients of the quadratic function that best fits the data points:

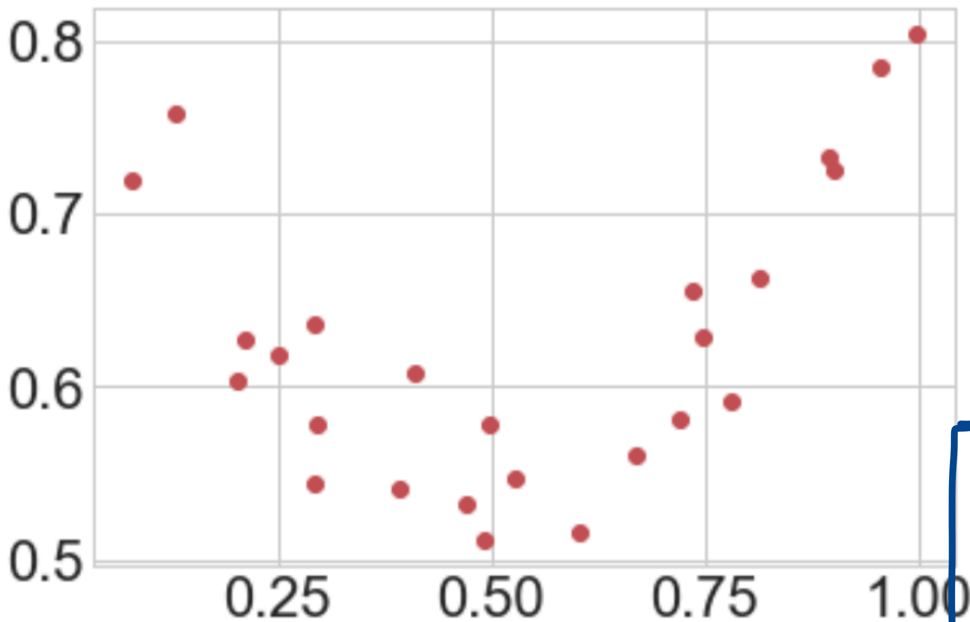
$$y = x_0 + x_1\, t + x_2\, t^2$$



We would not want our "fit" curve to pass through the data points exactly as we are looking to model the general trend and not capture the noise.

Data fitting

$$y = x_0 + x_1\, t + x_2\, t^2$$



$$\begin{aligned}
y_1 &= x_0 + x_1 t_1 + x_2 t_1^2 \\
y_2 &= x_0 + x_1 t_2 + x_2 t_2^2 \\
y_3 &= x_0 + x_1 t_3 + x_2 t_3^2 \\
&\vdots \\
y_m &= x_0 + x_1 t_m + x_2 t_m^2
\end{aligned}$$

$$\underset{m\times 1}{\underset{\sim}{b}} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} = \overset{\underset{=}{A}}{\begin{bmatrix} 1 & t_1 & t_1^2 \\ 1 & t_2 & t_2^2 \\ 1 & t_3 & t_3^2 \\ \vdots & \vdots & \vdots \\ 1 & t_m & t_m^2 \end{bmatrix}} \overset{\underset{\sim}{x}}{\begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}}$$

3×1

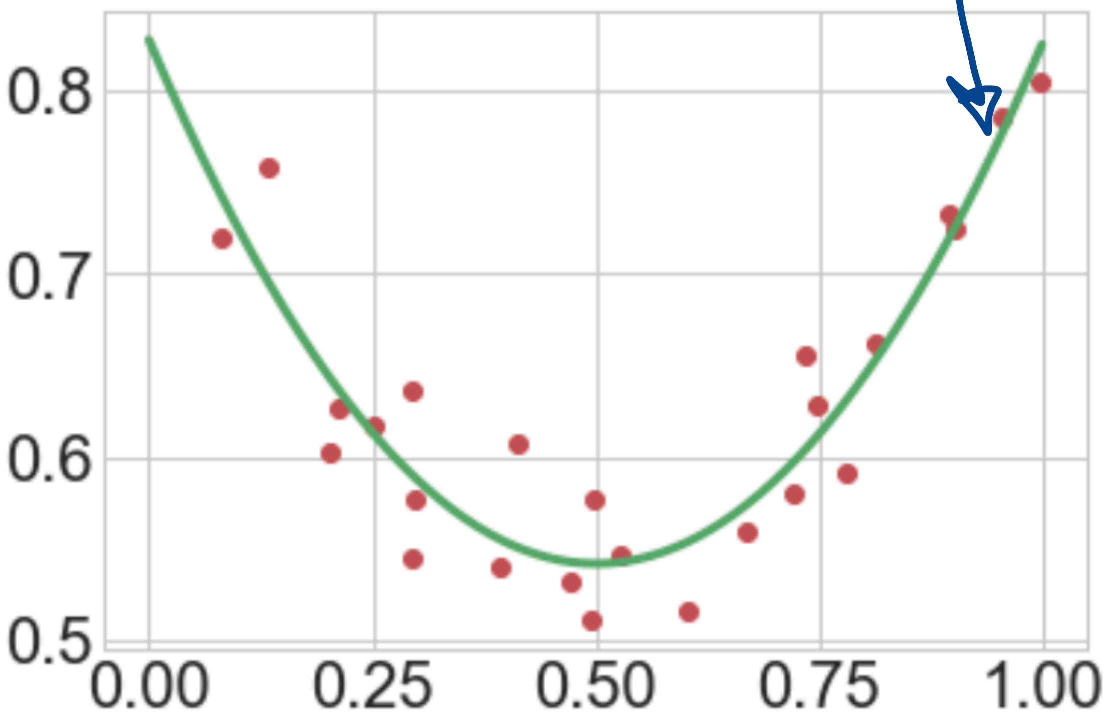
$(n\times 1)$

Data fitting

$A \quad b \rightarrow Ax \cong b$

$$\begin{bmatrix} 1 & t_1 & t_1^2 \\ \vdots & \vdots & \vdots \\ 1 & t_m & t_m^2 \end{bmatrix} \begin{bmatrix} x_o \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}$$

Solve: $\boxed{\boldsymbol{A}^T \boldsymbol{A}\, \boldsymbol{x} = \boldsymbol{A}^T\, \boldsymbol{b}}$



Which function is not suitable for linear least squares?

A) $y = a + b\,x + c\,x^2 + d\,x^3$

B) $y = x(a + b\,x + c\,x^2 + d\,x^3)$

C) $y = a\sin(x) + b/\cos(x)$

D) $y = a\sin(x) + x/\cos(bx)$

E) $y = a\,e^{-2x} + b\,e^{2x}$

$$y = x_0 + x_1 t$$

$$y = x_0 + x_1 t + x_2 t^2$$

$$Ax = b$$

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} e^{-2x_1} & e^{2x_1} \\ e^{-2x_2} & e^{2x_2} \\ \vdots & \vdots \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} \ \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$$

Computational Cost

$A_{m\times n}$

$$\boxed{A^T A x = A^T b}$$

① Construct A^TA → $\left[\; n\times m \;\right]\left[\; m\times n \;\right] = \left[\; n\times n \;\right]$

inner product $O(m)$

n^2 times

$\boxed{O(mn^2)}$

② Factorize $\underbrace{A^TA}_{n\times n} \Rightarrow \boxed{O(n^3)}$

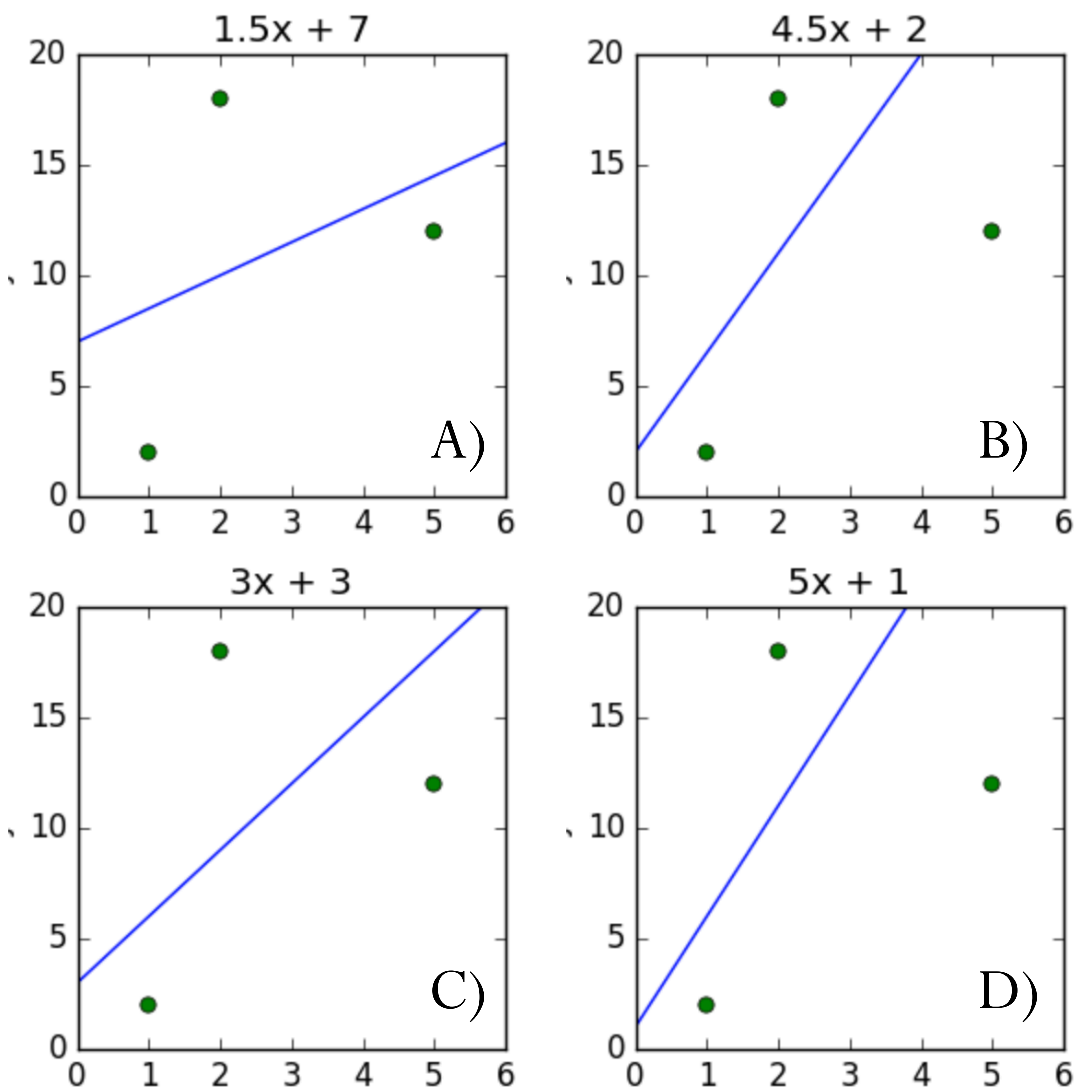
③ Solve $\Rightarrow \boxed{O(n^2)}$

overall cost: $O(mn^2)$ $m>n$

Short questions

Given the data in the table below, which of the plots shows the line of best fit in terms of least squares?

x	1	2	5
y	2	18	12



Short questions

Given the data in the table below, and the least squares model

$$y = c_1 + c_2 \sin(t\pi) + c_3 \sin(t\pi/2) + c_4 \sin(t\pi/4)$$

written in matrix form as

$$A \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} \cong \mathbf{y}$$

determine the entry A_{23} of the matrix $\boldsymbol{A}$.
Note that indices start with 1.

A) -1.0
B) 1.0
C) -0.7
D) 0.7
E) 0.0

t_i	y_i
0.5	0.72
1.0	0.79
1.5	0.72
2.0	0.97
2.5	1.03
3.0	0.96
3.5	1.00

Solving Linear Least Squares with SVD

What we have learned so far…

$\boldsymbol{A}$ is a $m \times n$ matrix where $m > n$

(more points to fit than coefficient to be determined)

Normal Equations: $\boldsymbol{A}^T \boldsymbol{A}\, \boldsymbol{x} = \boldsymbol{A}^T\, \boldsymbol{b}$

- The solution $\boldsymbol{A}\, \boldsymbol{x} \cong \boldsymbol{b}$ is unique if and only if $rank(\mathbf{A}) = n$

($\boldsymbol{A}$ is full column rank)

- $rank(\mathbf{A}) = n$ → columns of $\boldsymbol{A}$ are ***linearly independent*** → n non-zero singular values → $\boldsymbol{A}^T\ \boldsymbol{A}$ has only positive eigenvalues → $\boldsymbol{A}^T \boldsymbol{A}$ is a symmetric and positive definite matrix → $\boldsymbol{A}^T \boldsymbol{A}$ is invertible

$$\boldsymbol{x} = (\boldsymbol{A}^T \boldsymbol{A})^{-1} \boldsymbol{A}^T\ \boldsymbol{b}$$

- If $rank(\mathbf{A}) < n$, then $\boldsymbol{A}$ is rank-deficient, and solution of linear least squares problem is ***not unique***.

Condition number for Normal Equations

Finding the least square solution of $\boldsymbol{A}\,\boldsymbol{x} \cong \boldsymbol{b}$ (where $\boldsymbol{A}$ is full rank matrix) using the Normal Equations

$$\boldsymbol{A}^T\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{A}^T\,\boldsymbol{b}$$

has some advantages, since we are solving a square system of linear equations with a symmetric matrix (and hence it is possible to use decompositions such as Cholesky Factorization)

However, the normal equations tend to worsen the conditioning of the matrix.

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

How can we solve the least square problem without squaring the condition of the matrix?

SVD to solve linear least squares problems

$\boldsymbol{A}$ is a $m \times n$ rectangular matrix where $m > n$, and hence the SVD decomposition is given by:

$$\boldsymbol{A} = \begin{pmatrix} \vdots & \cdots & \vdots \\ \boldsymbol{u}_1 & \cdots & \boldsymbol{u}_m \\ \vdots & \cdots & \vdots \end{pmatrix} \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \\ & & 0 \\ & & \vdots \\ & & 0 \end{pmatrix} \begin{pmatrix} \cdots & \mathbf{v}_1^T & \cdots \\ \vdots & \vdots & \vdots \\ \cdots & \mathbf{v}_n^T & \cdots \end{pmatrix}$$

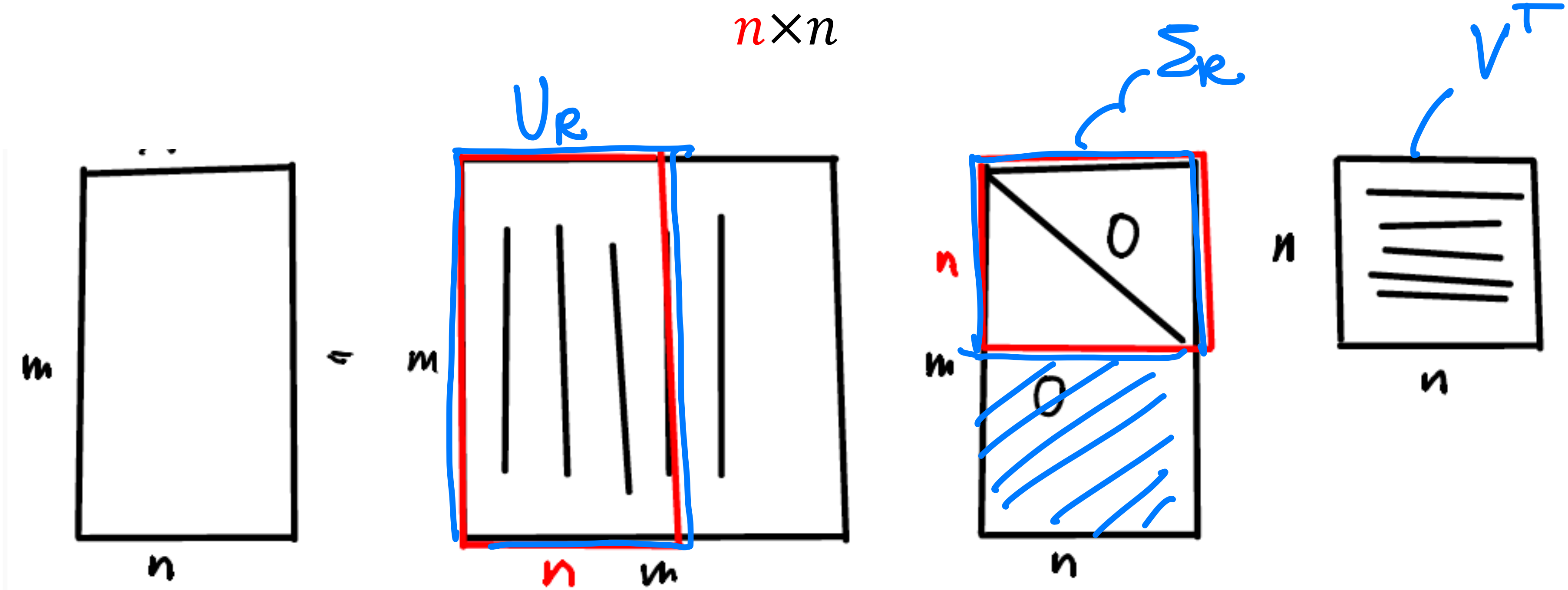
We want to find the least square solution of $\boldsymbol{A}\,\boldsymbol{x} \cong \boldsymbol{b}$, where $\boldsymbol{A} = \boldsymbol{U}\,\boldsymbol{\Sigma}\,\boldsymbol{V}^{\boldsymbol{T}}$

or better expressed in reduced form: $\boldsymbol{A} = \boldsymbol{U}_R\;\boldsymbol{\Sigma}_{\boldsymbol{R}}\;\boldsymbol{V}^{\boldsymbol{T}}$

Recall Reduced SVD $\boxed{m > n}$

$$A = U_R \, \Sigma_R \, V^T$$

A: $m \times n$ — U_R: $m \times n$ — Σ_R: $n \times n$ — V^T: $n \times n$



Shapes of the Reduced SVD

Suppose you compute a reduced SVD $A = U\Sigma V^T$ of a 10×14 matrix A. What will the shapes of U, Σ, and V be?
Hint: Remember the transpose on V!

The shape of U will be ____ $\times$ ____ .

The shape of Σ will be ____ $\times$ ____ .

The shape of V will be ____ $\times$ ____ .

SVD to solve linear least squares problems

$$\boldsymbol{A} = \boldsymbol{U}_R\ \boldsymbol{\Sigma}_R\ \boldsymbol{V}^T$$

$$\boldsymbol{A} = \begin{pmatrix} \vdots & \cdots & \vdots \\ \boldsymbol{u}_1 & \cdots & \boldsymbol{u}_n \\ \vdots & \cdots & \vdots \end{pmatrix} \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{pmatrix} \begin{pmatrix} \cdots & \mathbf{v}_1^T & \cdots \\ \vdots & \vdots & \vdots \\ \cdots & \mathbf{v}_n^T & \cdots \end{pmatrix}$$

$$A x = b \longrightarrow A^T A x = A^T b$$

$$(U_R \Sigma_R V^T)^T (U_R \Sigma_R V^T) x = (U_R \Sigma_R V^T)^T b$$

$$(V^T)^T \Sigma_R^T \underbrace{U_R^T U_R} \Sigma_R V^T x = (V^T)^T \Sigma_R^T U_R^T b$$

$$\Sigma_R^{-1} = ?$$

$$V \Sigma_R^T \Sigma_R V^T x = V \Sigma_R^T U_R^T b$$

$$V \Sigma_R^2 V^T x = V \Sigma_R U_R^T b \Longrightarrow \boxed{\Sigma_R^2 V^T x = \Sigma_R U_R^T b}$$

① Full rank A $\quad A_{m\times n}$: $\mathrm{rank}(A) = n$

$$\Sigma_R^2 V^T x = \Sigma_R U_R^T b \Longrightarrow V^T x = \Sigma_R^{-1} U_R^T b$$

$$\boxed{\underbrace{x}_{n\times 1} = \underbrace{V}_{n\times n}\, \underbrace{\Sigma_R^{-1}}_{n\times n}\, \underbrace{U_R^T}_{n\times m}\, \underbrace{b}_{m\times 1}}$$

unique solution ✓

② Rank deficient $\quad A_{m\times n}$: $\mathrm{rank}(A) = r < n$

$\Sigma_R^2 V^T x = \Sigma_R U_R^T b \longrightarrow$ solution is not unique

① Find x s.t $\min_x \|Ax - b\|_2^2$ ✓

\+ $\min_x \|x\|_2$ ✓

$$\Sigma_R^2 V^T x = \Sigma_R U_R^T b$$

$\text{rank}(A) = r < n$

Change of variable $y = V^T x \rightarrow x = Vy$

Let's solve $\Sigma_R\, y = \boxed{U_R^T b}$

$$U_R = \begin{bmatrix} | & & | \\ u_1 & \cdots & u_n \\ | & & | \end{bmatrix} \quad m \times n$$

$$\underbrace{\begin{bmatrix} \sigma_1 & & & & & \\ & \sigma_2 & & & & \\ & & \ddots & & & \\ & & & \sigma_r & & \\ & & & & 0 & \\ & & & & & \ddots \; 0 \end{bmatrix}}_{\Sigma_R \; n\times n} \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_r \\ y_{r+1} \\ \vdots \\ y_n \end{bmatrix}}_{y} = \underbrace{\begin{bmatrix} u_1^T b \\ u_2^T b \\ \vdots \\ u_r^T b \\ \vdots \\ u_n^T b \end{bmatrix}}_{U_R^T b}$$

$$y_1 \sigma_1 = u_1^T b = u_1 \cdot b$$

$$\boxed{\begin{aligned} y_1 &= u_1^T b / \sigma_1 \\ y_2 &= u_2^T b / \sigma_2 \\ &\vdots \\ y_r &= u_r^T b / \sigma_r \end{aligned}}$$

set it to zero!

$$y_i = \frac{u_i^T b}{\sigma_i} \qquad i = 1, \dots, r$$

$$y_i = \cancel{0} \qquad i = r+1, \dots, n \; ?$$

$y_{r+1} = ?$

$\min \|x\|$

$\min \|Vy\|_2$

anything!! $\rightarrow$ solution not unique

In summary:

$$y_i = \begin{cases} \dfrac{u_i^T b}{\sigma_i}, & \text{if } i = 1, \dots, r \\ 0, & \text{if } i = r+1, \dots, n \end{cases} \Bigg\} \; \underset{\sim}{y}$$

Compute $\underset{\sim}{x} \longrightarrow \underset{\sim}{x} = \underset{=}{V}\, \underset{\sim}{y} = \sum_{i=1}^{n} y_i \underset{\sim}{v_i}$

$$\boxed{\underset{\sim}{x} = \sum_{\substack{i=1 \\ \sigma_i \neq 0}}^{n} \left(\frac{u_i^T b}{\sigma_i} \right) \underset{\sim}{v_i}}$$

$\underset{=}{A}\, \underset{\sim}{x} = \underline{b}$

$\underset{=}{A}$ is <u>rank deficient</u>

$$\boxed{\underset{\sim}{x} = \sum_{i=1}^{n} \left(\frac{u_i^T b}{\sigma_i}\right) \underset{\sim}{v_i}}$$

$\sigma_i \neq 0$

$U_R^T b$ — $n \times m$, $m \times 1$ $\longrightarrow$ $O(mn)$

$\Sigma_R^T z$ — $n \times n$, $n \times 1$ $\xrightarrow{\Sigma \text{ is diag}}$ $O(n)$

$$\Sigma_R^2 V^T x = \Sigma_R U_R^T b \longrightarrow \Sigma_R^+$$

$$V^T x = \Sigma_R^+ U_R^T b$$

$\Sigma_R^+ = \Sigma_R^{-1}$ full rank

$$\boxed{\underset{\sim}{x} = \underset{=}{V} \underbrace{\underset{=}{\Sigma_R^+} \underbrace{\underset{=}{U_R^T}\, \underline{b}}_{z}}_{y}}$$

$V y$ — $n \times n$, $n \times 1$ $\longrightarrow$ $O(n^2)$

SVD $\Rightarrow O(mn^2)$

$m \geq n$

overall $O(mn)$

Example:

Consider solving the least squares problem $\boldsymbol{A\,x} \cong \boldsymbol{b}$, where the singular value decomposition of the matrix $\boldsymbol{A} = \boldsymbol{U\,\Sigma\,V^T x}$ is:

$$\begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 14 & 0 & 0 \\ 0 & 14 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x} \cong \begin{bmatrix} 12 \\ 9 \\ 9 \\ 10 \end{bmatrix}$$

Determine $\|\boldsymbol{b} - \boldsymbol{A\,x}\|_2$

Example

Suppose you have $\boldsymbol{A} = \boldsymbol{U}\,\boldsymbol{\Sigma}\,\boldsymbol{V}^{\boldsymbol{T}}\boldsymbol{x}$ calculated. What is the cost of solving

$$\min_{\boldsymbol{x}} \|\boldsymbol{b} - \boldsymbol{A}\,\boldsymbol{x}\|_2^2\ ?$$

A) $O(n)$
B) $O(n^2)$
C) $O(mn)$
D) $O(m)$
E) $O(m^2)$