Linear least squares problems (§12.1)

- Least squares: a special class of optimization problems.
- Well studied because:
  - Many applications
  - Easy to solve
- Basic idea: want to solve
$$\boldsymbol{Ax} = \boldsymbol{b}$$
even when there is no solution (!).

- Consider the system of linear equations
$$\boldsymbol{Ax} = \boldsymbol{b},$$
where $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $m \geq n$, and rank $\boldsymbol{A} = n$ (full rank).
- Number of unknowns $\leq$ number of equations (overdetermined).
- If $\boldsymbol{b} \notin \mathcal{R}(\boldsymbol{A})$, then no solution (inconsistent).
- Alternative goal:
$$\text{minimize } \frac{1}{2}\|\boldsymbol{Ax} - \boldsymbol{b}\|^2.$$
- The reason for the factor of $1/2$ will be clear later.

Examples of linear least squares problems

- Linear regression (line fitting)
- Discrete Fourier series
- Linear system identification
- Optimal filtering
- ... others
- Detailed examples later.

A note about quadratics

- Consider a quadratic
$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} - \boldsymbol{x}^T\boldsymbol{b} + c,$$
where $\boldsymbol{Q} > 0$.

- We can write f as
$$f(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}^*)^T\boldsymbol{Q}(\boldsymbol{x}-\boldsymbol{x}^*) + \left(c - \frac{1}{2}\boldsymbol{x}^{*T}\boldsymbol{Q}\boldsymbol{x}^*\right),$$
where $\boldsymbol{x}^* = \boldsymbol{Q}^{-1}\boldsymbol{b}$.

- Hence, $\boldsymbol{x}^* = \boldsymbol{Q}^{-1}\boldsymbol{b}$ is the unique global minimizer.

Quadratic formulation of least squares problem

- Rewrite $f(\boldsymbol{x}) = \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2/2$ as
$$\begin{aligned} f(\boldsymbol{x}) &= \frac{1}{2}(\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b})^T(\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}) \\ &= \frac{1}{2}\boldsymbol{x}^T(\boldsymbol{A}^T\boldsymbol{A})\boldsymbol{x} - \boldsymbol{x}^T(\boldsymbol{A}^T\boldsymbol{b}) + \frac{\boldsymbol{b}^T\boldsymbol{b}}{2}. \end{aligned}$$

- Thus, f is a quadratic.

- It is clear that $\boldsymbol{A}^T\boldsymbol{A}$ is positive semidefinite. (Why?)

- But, is $\boldsymbol{A}^T\boldsymbol{A} > 0$? For if it is, then the unique global minimizer is $\boldsymbol{x}^* = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}$.

Lemma (12.1): rank $\boldsymbol{A} = n \Leftrightarrow \boldsymbol{A}^T\boldsymbol{A} > 0$.

Proof:

- Given $\boldsymbol{x} \in \mathbb{R}^n$, we have
$$\boldsymbol{x}^T(\boldsymbol{A}^T\boldsymbol{A})\boldsymbol{x} = \|\boldsymbol{A}\boldsymbol{x}\|^2.$$

- By property of norm,
$$\|\boldsymbol{A}\boldsymbol{x}\|^2 = 0 \qquad \Leftrightarrow \qquad \boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}.$$

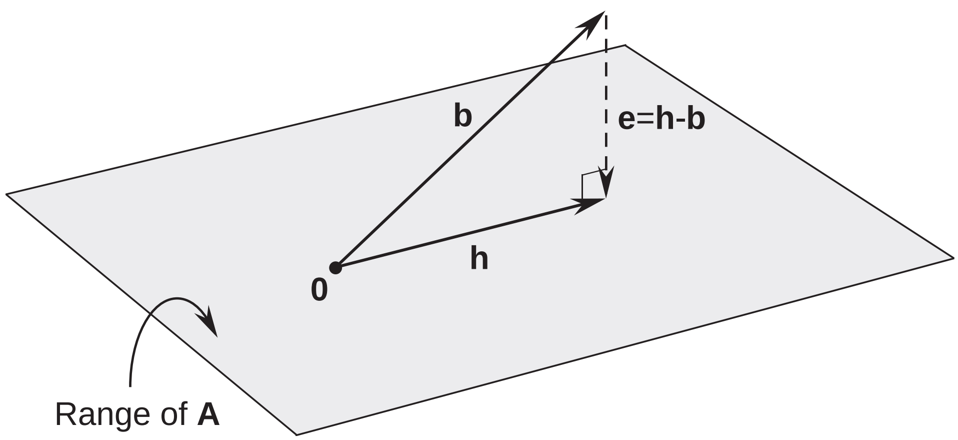
- Hence,
$$\begin{aligned} \boldsymbol{A}^T\boldsymbol{A} \not> 0 &\Leftrightarrow \exists \boldsymbol{x} \neq \boldsymbol{0}:\ \boldsymbol{x}^T(\boldsymbol{A}^T\boldsymbol{A})\boldsymbol{x} = 0 \\ &\Leftrightarrow \exists \boldsymbol{x} \neq \boldsymbol{0}:\ \boldsymbol{A}\boldsymbol{x} = \boldsymbol{0} \\ &\Leftrightarrow \text{rank } \boldsymbol{A} < n. \end{aligned}$$

Solution to least squares problem

- Theorem: The unique (global) minimizer to the least squares problem is $\boldsymbol{x}^* = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}$.
- Proof: Follows immediately from the quadratic formulation of the problem.

Orthogonality principle

- Prop. (12.1): The vector $\boldsymbol{A}\boldsymbol{x}^*$ is orthogonal to $\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{b}$.
- Proof: By simple algebra.
- Least squares problem $\equiv$ approximating $\boldsymbol{b}$ by a point in $\mathcal{R}(\boldsymbol{A})$.
- $\boldsymbol{h} = \boldsymbol{A}\boldsymbol{x}^*$ is the projection of $\boldsymbol{b}$ onto $\mathcal{R}(\boldsymbol{A})$.
- The projection $\boldsymbol{A}\boldsymbol{x}^*$ is orthogonal to the error in the approximation $\boldsymbol{e} = \boldsymbol{h} - \boldsymbol{b}$.



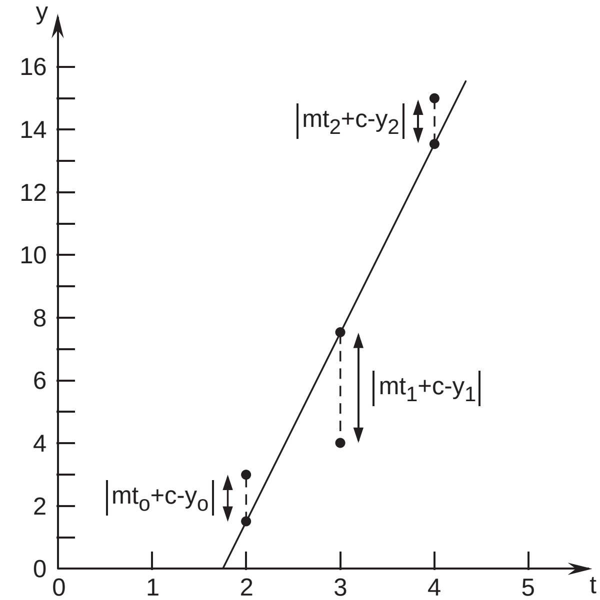
- Let $\boldsymbol{a}_i$ represent the ith column of $\boldsymbol{A}$;
  i.e., $\boldsymbol{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \dots, \boldsymbol{a}_n]$.
- We have
$$\boldsymbol{A}^T\boldsymbol{A} = \begin{bmatrix} \langle \boldsymbol{a}_1, \boldsymbol{a}_1 \rangle & \cdots & \langle \boldsymbol{a}_n, \boldsymbol{a}_1 \rangle \\ \vdots & & \vdots \\ \langle \boldsymbol{a}_1, \boldsymbol{a}_n \rangle & \cdots & \langle \boldsymbol{a}_n, \boldsymbol{a}_n \rangle \end{bmatrix}$$
  Name: *Gram matrix* (or Grammian).
- Also,
$$\boldsymbol{A}^T\boldsymbol{b} = \begin{bmatrix} \langle \boldsymbol{a}_1, \boldsymbol{b} \rangle \\ \vdots \\ \langle \boldsymbol{a}_n, \boldsymbol{b} \rangle \end{bmatrix}.$$

Example (12.1): Linear Regression

- Given data:

i	0	1	2
t_i	2	3	4
y_i	3	4	15

- Want to find straight line of best fit.



- Equation of straight line: $y = mt + c$.
- If a straight line passed through all three points, we would have

$$\begin{aligned} 2m + c &= 3 \\ 3m + c &= 4 \\ 4m + c &= 15. \end{aligned}$$

- The above is an inconsistent (overdetermined) set of equations.
- The total squared error between the line and the given data points is

$$\begin{aligned} f(m, c) &= \sum_{i=0}^{2} (mt_i + c - y_i)^2 \\ &= \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2 \end{aligned}$$

where

$$\boldsymbol{A} = \begin{bmatrix} 2 & 1 \\ 3 & 1 \\ 4 & 1 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} 3 \\ 4 \\ 15 \end{bmatrix}, \qquad \boldsymbol{x} = \begin{bmatrix} m \\ c \end{bmatrix}.$$

- The above is a least squares problem. Note that rank $\boldsymbol{A} = 2$.

- We compute

$$(\boldsymbol{A}^T\boldsymbol{A})^{-1} = \frac{1}{6}\begin{bmatrix} 3 & -9 \\ -9 & 29 \end{bmatrix}, \qquad \boldsymbol{A}^T\boldsymbol{b} = \begin{bmatrix} 78 \\ 22 \end{bmatrix}.$$

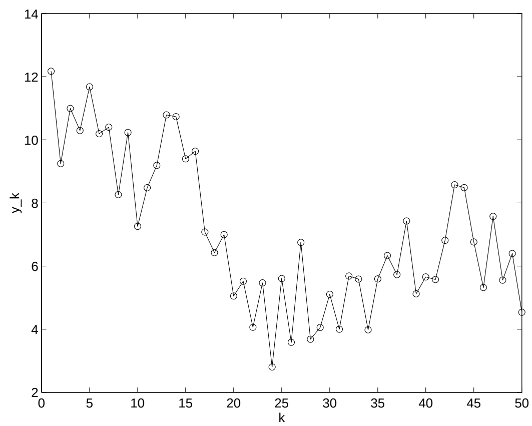
- Solution:

$$\boldsymbol{x}^* = \begin{bmatrix} m^* \\ c^* \end{bmatrix} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b} = \begin{bmatrix} 6 \\ -32/3 \end{bmatrix}.$$

Example: Signal representation

- Given: a discrete time signal

$$\boldsymbol{b} = [b_1, b_2, \ldots, b_m]^T.$$



- Want to "represent" this signal using other prespecified signals.

- Specifically, want to write this signal as

$$\boldsymbol{b} \approx x_1\boldsymbol{s}^{(1)} + x_2\boldsymbol{s}^{(2)} + \cdots + x_n\boldsymbol{s}^{(n)},$$

  where $\boldsymbol{s}^{(1)}, \ldots, \boldsymbol{s}^{(n)}$ are prespecified signals.

- What coefficients $x_1, \ldots, x_n$ should we use?

- Examples of prespecified signals: sinusoids (Fourier), square waves, wavelets, sigmoids (neural nets).

- Specific example: (Haar basis)

$$\begin{aligned} \boldsymbol{s}^{(1)} &= [1, -1, 1, -1, 1, -1, 1, -1]^T \\ \boldsymbol{s}^{(2)} &= [1, 1, -1, -1, 1, 1, -1, -1]^T \\ \boldsymbol{s}^{(3)} &= [1, 1, 1, 1, -1, -1, -1, -1]^T. \end{aligned}$$

- Note that the above signals are mutually orthogonal.

- Formulate as least squares problem.

- Want to find $x_1, \ldots, x_n$ to minimize
$$\left\| \left(x_1 \boldsymbol{s}^{(1)} + x_2 \boldsymbol{s}^{(2)} + \cdots + x_n \boldsymbol{s}^{(n)} \right) - \boldsymbol{b} \right\|^2 .$$

- Define
$$\begin{aligned} \boldsymbol{A} &= \left[\boldsymbol{s}^{(1)}, \boldsymbol{s}^{(2)}, \ldots, \boldsymbol{s}^{(n)} \right], \\ \boldsymbol{x} &= \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}. \end{aligned}$$

- Objective function becomes $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2$.

- To find the solution, we first compute $\boldsymbol{A}^T \boldsymbol{A}$.

- Note that the (i, j)th entry of $\boldsymbol{A}^T \boldsymbol{A}$ is
$$\left\langle \boldsymbol{s}^{(i)}, \boldsymbol{s}^{(j)} \right\rangle = \boldsymbol{s}^{(i)T} \boldsymbol{s}^{(j)}.$$

- We use the following identities:
$$\begin{aligned} \boldsymbol{s}^{(i)T} \boldsymbol{s}^{(j)} = \sum_{k=1}^{m} s_k^{(i)} s_k^{(j)} &= 0 \text{ for } i \neq j \\ \boldsymbol{s}^{(i)T} \boldsymbol{s}^{(i)} = \sum_{k=1}^{m} (s_k^{(i)})^2 &= m. \end{aligned}$$

- With the aid of the previous identities, we find that
$$\boldsymbol{A}^T \boldsymbol{A} = m \boldsymbol{I}_n = \begin{bmatrix} m & & 0 \\ & \ddots & \\ 0 & & m \end{bmatrix}$$
which is clearly nonsingular, with inverse
$$(\boldsymbol{A}^T \boldsymbol{A})^{-1} = \frac{1}{m} \boldsymbol{I}_n.$$

- We next compute $\boldsymbol{A}^T \boldsymbol{b}$:
$$\boldsymbol{A}^T \boldsymbol{b} = \begin{bmatrix} \boldsymbol{s}^{(1)T} \boldsymbol{b} \\ \vdots \\ \boldsymbol{s}^{(n)T} \boldsymbol{b} \end{bmatrix}.$$

- The solution is

$$\boldsymbol{x}^* = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b} = \frac{1}{m}\begin{bmatrix} \boldsymbol{s}^{(1)T}\boldsymbol{b} \\ \vdots \\ \boldsymbol{s}^{(n)T}\boldsymbol{b} \end{bmatrix}.$$

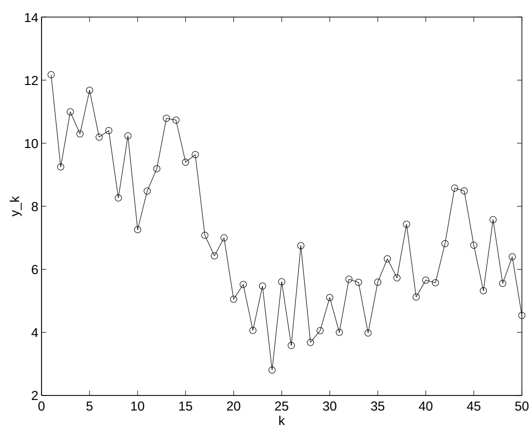
- Hence, the ith component is:

$$x_i^* = \frac{1}{m}\boldsymbol{s}^{(i)T}\boldsymbol{b} = \frac{1}{m}\sum_{k=1}^{m} s_k^{(i)} b_k.$$

- Interpretation: x_i^* is the "correlation" between the signals $\boldsymbol{b}$ and $\boldsymbol{s}^{(i)}$.

Example: Linear system identification

- Consider a discrete time signal, $\{y_k\}$. (E.g., speech signal, stock price, etc.)



- We model the signal using a linear model:

$$y_k = a_1 y_{k-1} + a_2 y_{k-2}$$

Name: second order autoregressive (or AR(2)) model.

- Given measurements of y, we want to determine (estimate) the values of a_1 and a_2.
- Estimate $\equiv$ parameter values that minimize total squared error.

- Suppose we are given $y_1, y_2, \ldots, y_{m+2}$.
- To formulate as a least squares problem, we write objective function as

$$f(a_1, a_2) = \sum_{k=3}^{m+2}(a_1 y_{k-1} + a_2 y_{k-2} - y_k)^2.$$

- So, define

$$\boldsymbol{x} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}, \qquad \boldsymbol{A} = \begin{bmatrix} y_2 & y_1 \\ y_3 & y_2 \\ \vdots & \vdots \\ y_{m+1} & y_m \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} y_3 \\ y_4 \\ \vdots \\ y_{m+2} \end{bmatrix}.$$

- Objective function becomes $f(\boldsymbol{x}) = \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2$.

- Solution: $\boldsymbol{x}^* = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}$.

- We have

$$\begin{aligned} \boldsymbol{A}^T\boldsymbol{A} &= \begin{bmatrix} \sum_{i=1}^m y_{i+1}^2 & \sum_{i=1}^m y_i y_{i+1} \\ \sum_{i=1}^m y_i y_{i+1} & \sum_{i=1}^m y_i^2 \end{bmatrix}, \\ \boldsymbol{A}^T\boldsymbol{b} &= \begin{bmatrix} \sum_{i=1}^m y_{i+1} y_{i+2} \\ \sum_{i=1}^m y_i y_{i+2} \end{bmatrix}. \end{aligned}$$

- Solution:

$$\begin{bmatrix} a_1^* \\ a_2^* \end{bmatrix} = \frac{1}{(\overline{Y_0^2})(\overline{Y_1^2}) - (\overline{Y_0 Y_1})^2} \begin{bmatrix} (\overline{Y_0^2})(\overline{Y_1 Y_2}) - (\overline{Y_0 Y_1})(\overline{Y_0 Y_2}) \\ (\overline{Y_1^2})(\overline{Y_0 Y_2}) - (\overline{Y_0 Y_1})(\overline{Y_1 Y_2}) \end{bmatrix},$$

  where

$$\overline{Y_j Y_k} = \sum_{i=1}^m y_{i+j} y_{i+k}.$$

- Can easily generalize the previous technique to autoregressive models of higher order:

$$y_n = a_1 y_{n-1} + a_2 y_{n-2} + \cdots + a_p y_{n-p}$$

  AR(p) model.

- We can even include external inputs:

$$\begin{aligned} y_n &= a_1 y_{n-1} + a_2 y_{n-2} + \cdots + a_p y_{n-p} \\ &\quad + b_1 u_{n-1} + \cdots + b_q u_{n-q} \end{aligned}$$

  (autoregressive moving average (ARMA(p,q)) model).

Recursive least squares (RLS) algorithm (§12.2)

- Consider the line fitting problem.

- Given data $(t_0, y_0), (t_1, y_1), (t_2, y_2)$, we can find line of best fit: m^*, c^*. We use

$$\boldsymbol{A} = \begin{bmatrix} t_0 & 1 \\ t_1 & 1 \\ t_2 & 1 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \end{bmatrix}.$$

- Solution: $[m^*, c^*]^T = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}$.
- Suppose we are now given another data point (t_3, y_3). Wish to find the line of best fit through all the data points so far.
- New matrices have extra row:

$$\boldsymbol{A} = \begin{bmatrix} t_0 & 1 \\ t_1 & 1 \\ t_2 & 1 \\ t_3 & 1 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix}.$$

- We have already calculated m^* and c^* for the first three data points.
- RLS: we can update the previous m^* and c^* to incorporate the new data point (without having to calculate the new m^* and c^* from scratch).

Formulation of RLS algorithm

- Given $\boldsymbol{A}_0$ and $\boldsymbol{b}^{(0)}$, suppose we solve the least squares problem

$$\text{minimize } \|\boldsymbol{A}_0\boldsymbol{x} - \boldsymbol{b}^{(0)}\|^2.$$

- Solution: $\boldsymbol{x}^{(0)} = (\boldsymbol{A}_0^T\boldsymbol{A}_0)^{-1}\boldsymbol{A}_0^T\boldsymbol{b}^{(0)}$.
- For simplicity, write $\boldsymbol{G}_0 = \boldsymbol{A}_0^T\boldsymbol{A}_0$.
- So, solution is

$$\boldsymbol{x}^{(0)} = \boldsymbol{G}_0^{-1}\boldsymbol{A}_0^T\boldsymbol{b}^{(0)}.$$

- Suppose we now add an extra "data point": $(\boldsymbol{a}_1, b_1)$, where $\boldsymbol{a}_1 \in \mathbb{R}^n$, $b_1 \in \mathbb{R}$.
- New matrices have extra row:

$$\boldsymbol{A}_1 = \begin{bmatrix} \boldsymbol{A}_0 \\ \boldsymbol{a}_1^T \end{bmatrix}, \qquad \boldsymbol{b}^{(1)} = \begin{bmatrix} \boldsymbol{b}^{(0)} \\ b_1 \end{bmatrix}.$$

- New problem

$$\text{minimize } \|\boldsymbol{A}_1\boldsymbol{x} - \boldsymbol{b}^{(1)}\|^2.$$

- New solution:

$$\boldsymbol{x}^{(1)} = \boldsymbol{G}_1^{-1}\boldsymbol{A}_1^T\boldsymbol{b}^{(1)},$$

  where $\boldsymbol{G}_1 = \boldsymbol{A}_1^T\boldsymbol{A}_1$.

- Goal: write $\boldsymbol{x}^{(1)}$ in terms of $\boldsymbol{x}^{(0)}$, $\boldsymbol{G}_0$, and the new data $(\boldsymbol{a}_1, b_1)$.

- Now,

$$\begin{aligned} \boldsymbol{G}_1 &= \begin{bmatrix} \boldsymbol{A}_0^T & \boldsymbol{a}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_0 \\ \boldsymbol{a}_1^T \end{bmatrix} \\ &= \boldsymbol{A}_0^T \boldsymbol{A}_0 + \boldsymbol{a}_1 \boldsymbol{a}_1^T \\ &= \boldsymbol{G}_0 + \boldsymbol{a}_1 \boldsymbol{a}_1^T. \end{aligned}$$

- Hence, we have a formula for $\boldsymbol{G}_1$ in terms of $\boldsymbol{G}_0$ and $\boldsymbol{a}_1$.

- Also,

$$\begin{aligned} \boldsymbol{A}_1^T \boldsymbol{b}^{(1)} &= \begin{bmatrix} \boldsymbol{A}_0^T & \boldsymbol{a}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{b}^{(0)} \\ b_1 \end{bmatrix} \\ &= \boldsymbol{A}_0^T \boldsymbol{b}^{(0)} + \boldsymbol{a}_1 b_1. \end{aligned}$$

- We now write

$$\begin{aligned} \boldsymbol{A}_0^T \boldsymbol{b}^{(0)} &= \boldsymbol{G}_0 \boldsymbol{G}_0^{-1} \boldsymbol{A}_0^T \boldsymbol{b}^{(0)} \\ &= \boldsymbol{G}_0 \boldsymbol{x}^{(0)} \\ &= (\boldsymbol{G}_1 - \boldsymbol{a}_1 \boldsymbol{a}_1^T) \boldsymbol{x}^{(0)} \\ &= \boldsymbol{G}_1 \boldsymbol{x}^{(0)} - \boldsymbol{a}_1 \boldsymbol{a}_1^T \boldsymbol{x}^{(0)}. \end{aligned}$$

- Hence,

$$\begin{aligned} \boldsymbol{x}^{(1)} &= \boldsymbol{G}_1^{-1} \boldsymbol{A}_1^T \boldsymbol{b}^{(1)} \\ &= \boldsymbol{G}_1^{-1} \left(\boldsymbol{A}_0^T \boldsymbol{b}^{(0)} + \boldsymbol{a}_1 b_1 \right) \\ &= \boldsymbol{G}_1^{-1} \left(\boldsymbol{G}_1 \boldsymbol{x}^{(0)} - \boldsymbol{a}_1 \boldsymbol{a}_1^T \boldsymbol{x}^{(0)} + \boldsymbol{a}_1 b_1 \right) \\ &= \boldsymbol{x}^{(0)} + \boldsymbol{G}_1^{-1} \boldsymbol{a}_1 \left(b_1 - \boldsymbol{a}_1^T \boldsymbol{x}^{(0)} \right), \end{aligned}$$

  where $\boldsymbol{G}_1$ can be calculated using

$$\boldsymbol{G}_1 = \boldsymbol{G}_0 + \boldsymbol{a}_1 \boldsymbol{a}_1^T.$$

- Ideally, we need an update formula for $\boldsymbol{G}_1^{-1}$.

- Use Sherman-Morrison formula:

$$(\boldsymbol{A} + \boldsymbol{u}\boldsymbol{v}^T)^{-1} = \boldsymbol{A}^{-1} - \frac{(\boldsymbol{A}^{-1}\boldsymbol{u})(\boldsymbol{v}^T \boldsymbol{A}^{-1})}{1 + \boldsymbol{v}^T \boldsymbol{A}^{-1} \boldsymbol{u}}.$$

- We have

$$\begin{aligned} \boldsymbol{G}_1^{-1} &= (\boldsymbol{G}_0 + \boldsymbol{a}_1 \boldsymbol{a}_1^T)^{-1} \\ &= \boldsymbol{G}_0^{-1} - \frac{\boldsymbol{G}_0^{-1} \boldsymbol{a}_1 \boldsymbol{a}_1^T \boldsymbol{G}_0^{-1}}{1 + \boldsymbol{a}_1^T \boldsymbol{G}_0^{-1} \boldsymbol{a}_1}. \end{aligned}$$

  For simplicity of notation, we rewrite $\boldsymbol{G}^{-1}$ as $\boldsymbol{P}$.

- We now have an update formula for updating from $\boldsymbol{x}^{(0)}$ and $\boldsymbol{P}_0$ to $\boldsymbol{x}^{(1)}$ and $\boldsymbol{P}_1$, given $\boldsymbol{a}_1$ and b_1.
- We can generalize the above to $k = 2, 3, 4, \ldots$.

Summary of RLS algorithm

$$\begin{aligned}\boldsymbol{P}_{k+1} &= \boldsymbol{P}_k - \frac{\boldsymbol{P}_k \boldsymbol{a}_{k+1} \boldsymbol{a}_{k+1}^T \boldsymbol{P}_k}{1 + \boldsymbol{a}_{k+1}^T \boldsymbol{P}_k \boldsymbol{a}_{k+1}} \\ \boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} + \boldsymbol{P}_{k+1} \boldsymbol{a}_{k+1} \left(b_{k+1} - \boldsymbol{a}_{k+1}^T \boldsymbol{x}^{(k)} \right).\end{aligned}$$

- The term $b_{k+1} - \boldsymbol{a}_{k+1}^T \boldsymbol{x}^{(k)}$ is called the *innovation*.
- If innovation is zero, then $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)}$.

Example: Line fitting

- Recall line fitting example. Original data:

i	0	1	2
t_i	2	3	4
y_i	3	4	15

- We have

$$\boldsymbol{A} = \begin{bmatrix} 2 & 1 \\ 3 & 1 \\ 4 & 1 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} 3 \\ 4 \\ 15 \end{bmatrix}, \qquad \boldsymbol{x} = \begin{bmatrix} m \\ c \end{bmatrix}.$$

- We compute

$$(\boldsymbol{A}^T \boldsymbol{A})^{-1} = \frac{1}{6} \begin{bmatrix} 3 & -9 \\ -9 & 29 \end{bmatrix}, \qquad \boldsymbol{A}^T \boldsymbol{b} = \begin{bmatrix} 78 \\ 22 \end{bmatrix}.$$

- Hence,

$$\boldsymbol{x}^* = \begin{bmatrix} m^* \\ c^* \end{bmatrix} = (\boldsymbol{A}^T \boldsymbol{A})^{-1} \boldsymbol{A}^T \boldsymbol{b} = \begin{bmatrix} 6 \\ -32/3 \end{bmatrix}.$$

- Given a new data point: $(t_3, y_3) = (5, 18)$.
- Recursive formulation:

$$\boldsymbol{P}_0 = \frac{1}{6} \begin{bmatrix} 3 & -9 \\ -9 & 29 \end{bmatrix}, \qquad \boldsymbol{x}^{(0)} = \begin{bmatrix} 6 \\ -32/3 \end{bmatrix}$$
$$\boldsymbol{a}_1 = \begin{bmatrix} 5 \\ 1 \end{bmatrix}, \qquad b_1 = 18.$$

- Hence,

$$\boldsymbol{P}_1 = \boldsymbol{P}_0 - \frac{\boldsymbol{P}_0 \boldsymbol{a}_1 \boldsymbol{a}_1^T \boldsymbol{P}_0}{1 + \boldsymbol{a}_1^T \boldsymbol{P}_0 \boldsymbol{a}_1} = \begin{bmatrix} 0.2 & -0.7 \\ -0.7 & 2.7 \end{bmatrix},$$

  and

$$\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \boldsymbol{P}_1 \boldsymbol{a}_1 \left(b_1 - \boldsymbol{a}_1^T \boldsymbol{x}^{(0)} \right) = \begin{bmatrix} 5.6 \\ -9.6 \end{bmatrix}.$$

Matlab demo

Related to Kalman filtering.