Applications of Least Squares

Practical Linear Algebra | Lecture 8

Statistical Interpretation

$$y = Ax$$

$$A \in \mathbb{R}^{m \times n}$$

- If A is skinny, we have an overdetermined system of equations
- Each row of A could be measurements we obtain from sensor readings
- y are target values we want to reach by combining the sensor readings
- x represents the way we combine the readings to obtain y

Statistical Interpretation

$$y = Ax + v$$

$$\hat{y} = Ax_{ls} = A\hat{x}$$

- More precise equation: add noise vector v
- v accounts for the mismatch between y and Ax
- Now the equals sign really is an equals sign

Statistical Interpretation

- Consider the difference between the true value and our estimate of x

$$
\begin{aligned}
x - \hat{x} = x - A^{\dagger}y &= x - A^{\dagger}(Ax + v) \\
&= x - A^{\dagger}Ax - A^{\dagger}v \\
&= x - (A^T A)^{-1}A^T Ax - A^{\dagger}v \\
&= x - x - A^{\dagger}v \\
&= -A^{\dagger}v
\end{aligned}
$$

Statistical Interpretation

$$x - \hat{x} = -A^{\dagger}v$$

- If we have no noise or measurement error, then $\hat{x} = x$
- We say $\hat{x}$ is an unbiased estimator of x

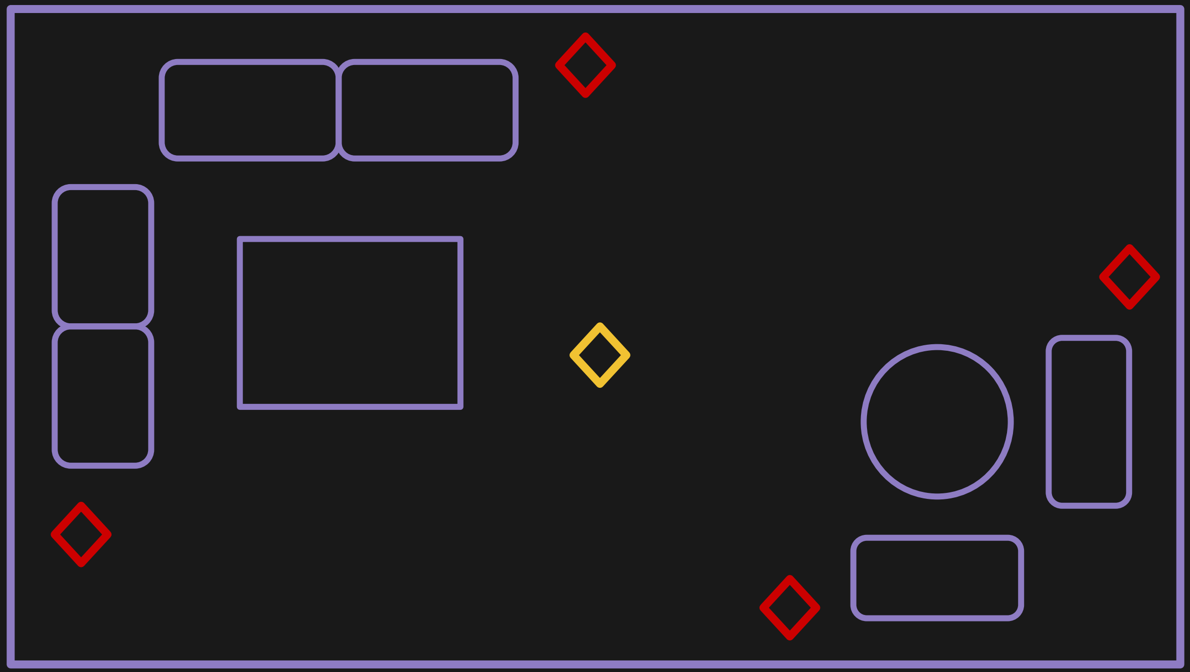
Least Squares is BLUE

$$BA = I$$

$$\sum_{i,j} B_{ij}^2 \geq \sum_{i,j} A_{ij}^{\dagger 2}$$

- $A^\dagger$ is the “smallest” matrix that can invert A (any other matrix B is bigger)
- $A^\dagger$ is the matrix that propagates noise the least
- The least squares solution is the best linear unbiased estimator of x
- Gauss-Markov theorem

Application: Temperature Estimation



Application: Temperature Estimation

$$y = x_1 a_1 + x_2 a_2 + x_3 a_3 + x_4 a_4$$

- Red sensors are cheap but not very accurate temperature sensors
- Yellow sensor is an accurate but very expensive sensor
- Problem: Can we combine several cheap sensors to form an accurate temperature estimate?

Application: Temperature Estimation

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

- Collect measurements from all sensors at many different times!
- We can stack all these measurements into one system of equations

Polynomial Fitting

$$y = at^3 + bt^2 + ct + d$$

- Imagine we have some time series data (input is time, output is data)
- We can try fitting a polynomial to the data – for example, a cubic
- We want to find the best values for the coefficients a, b, c, d
- We can compute good estimates if we have data at several times

Polynomial Fitting

$$y = at^3 + bt^2 + ct + d$$

$$\Rightarrow \quad y = d + ct + bt^2 + at^3$$

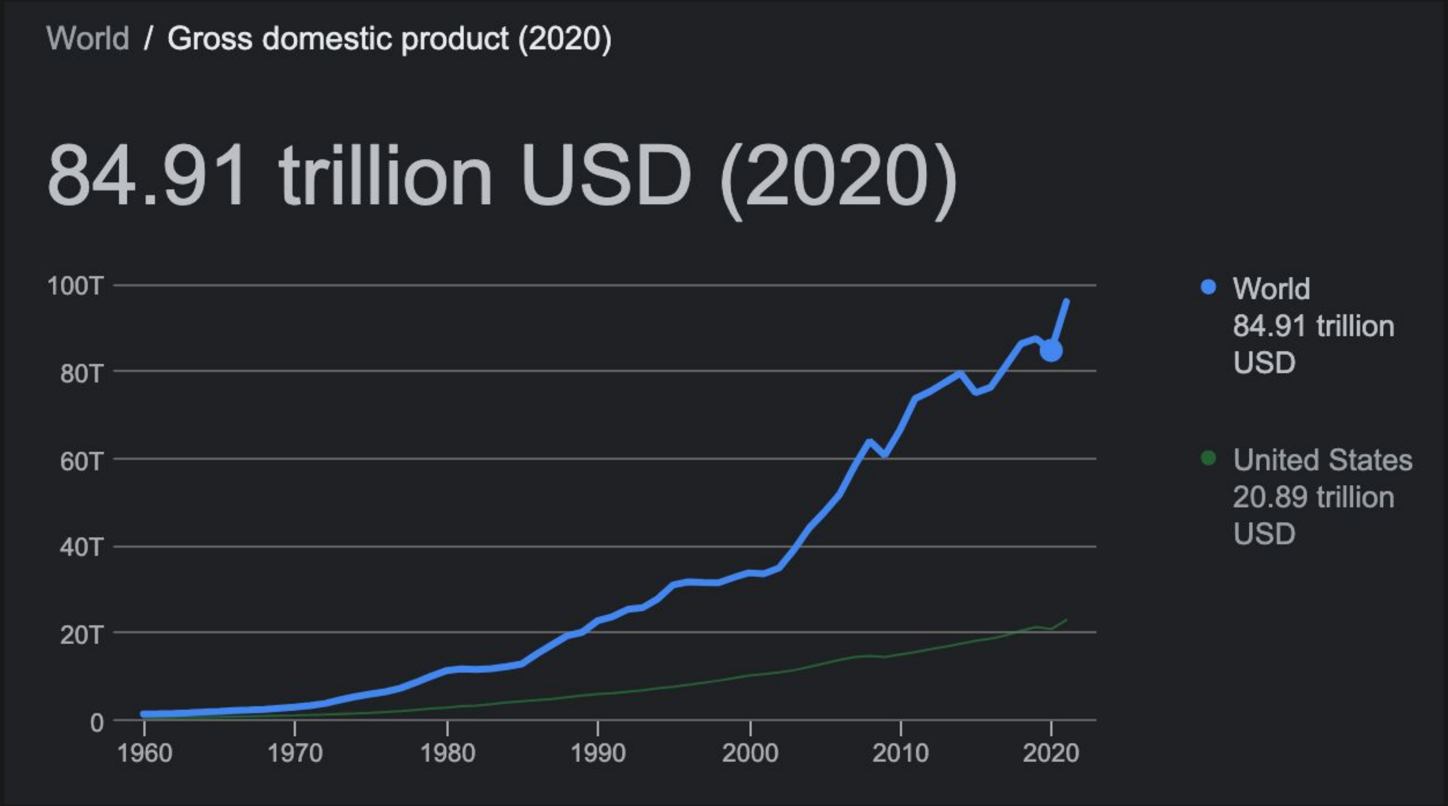
$$\Rightarrow \quad y = 1d + ct + bt^2 + at^3$$

$$\Rightarrow \quad y = \begin{bmatrix} 1 & t & t^2 & t^3 \end{bmatrix} \begin{bmatrix} d \\ c \\ b \\ a \end{bmatrix}$$

Polynomial Fitting

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & t_1 & t_1^2 & t_1^3 & \cdots & t_1^{n-1} \\ 1 & t_2 & t_2^2 & t_2^3 & \cdots & t_2^{n-1} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & t_m & t_m^2 & t_m^3 & \cdots & t_m^{n-1} \end{bmatrix}}_{\text{Vandermonde matrix}} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Application: GDP Forecasting



Application: Image Alignment

Image credit:
https://paperswithcode.com/task/image-matching

- We can only align images using least squares for affine cameras (simple model)
- For projective cameras (more realistic model), we need SVD – we'll come back!

Next Time

- Underdetermined systems and least norm