

Lecture 14 - B

§9.1 Least-Squares

T. Takahashi

Department of Mathematics and Statistics
University of New Mexico



We will answer these questions:

- How is SVD used for semantic analysis?
- What is an overdetermined system and when does it arise in applications?
- What is the standard way to solve an overdetermined system?
- How can you use the SVD to solve overdetermined systems and why?



Semantic Analysis



Too Much Text

- Suppose we collect hundreds of thousands of customer reviews:

Examples

- "game is fun for kids"
- "does not work on mac"
- "installation failed on windows"
- "kids love this game"



Too Much Text

- Suppose we collect hundreds of thousands of customer reviews:

Examples

- "game is fun for kids"
- "does not work on mac"
- "installation failed on windows"
- "kids love this game"

Question

Can we automatically understand what people are talking about?



- We want to find patterns in how words appear
- Words that appear together may represent a "topic"

Key Idea

We will turn text into numbers and let linear algebra find structure.



Document-Word Matrix

	game	kids	mac	install
Doc 1	1	1	0	0
Doc 2	0	0	1	0
Doc 3	0	0	0	1
Doc 4	1	1	0	0

- Load every document and create a list of unique words
- Then create a matrix of 1s and 0s
- Each row = a document
- Each column = a word



$$A = U\Sigma V^T$$

- V^T contains combinations of words
- Σ tells us how important each combination is
- U tells us how documents use those combinations



Example



- 2018 Amazon customer reviews in the Video Games category
- 497,577 total reviews
- Source: <https://nijianmo.github.io/amazon/index.html>

Result

- Each eigenvector contains an ordered pattern of words
- Top 7 words in v_1 :

"game" "like" "games" "just" "get" "play" "time"

- Top 7 in v_2

"game" "like" "games" "just" "get" "really" "controller"

- Top 7 in v_3

"games" "wii" "just" "play" "get" "like" "mario"

Inference

We can't really glean much from this. We have to omit filler words.



New Result

- Removed filler/obscure words

game, games, play, like, get, really, time

- New result

```
1 Topic 1: good even first well story fun
2
3 Topic 2: wii controller story link xbox characters
4
5 Topic 3: link mario good class characters normal
6
7 Topic 4: mario wii fun link super new
8
9 Topic 5: great good fun story graphics even
```

Inference

Gamers appreciate a well-written story. Link and Mario games are popular. Good graphics are appreciated.

- We never told the computer what a "topic" is
- It discovered them from patterns in the data

Key Idea

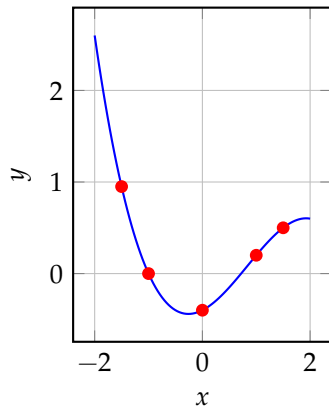
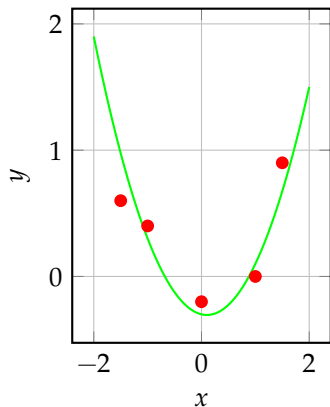
SVD turns raw text into meaningful ordered structure.



Linear Regression



Regression vs Interpolation

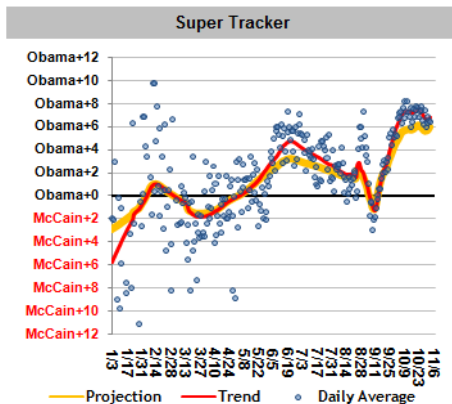


- Both extract continuous functions $f(x)$ from discrete data
- **Regression:** find a function that “best” fits the data
- **Interpolation:** find a function that passes through all data points



Noise in Real data

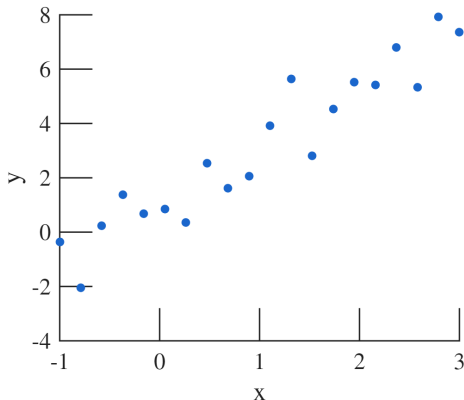
- Real data $\{(x_1, y_1), \dots, (x_n, y_n)\}$ contains noise (random error)
- Regression is more suitable than interpolation for noisy data



Linear Regression

Question

Here is a set of 20 noisy data points. How do you fit a line on the data?



Interpolation?

Interpolation Problem

- Given: n data points $\{(x_1, y_1), \dots, (x_n, y_n)\}$
- Goal: Find a line $y = mx + b$ that fits through all n points

In matrix form

$$\underbrace{\begin{bmatrix} x_1 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{bmatrix}}_A \underbrace{\begin{bmatrix} m \\ b \end{bmatrix}}_x = \underbrace{\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}}_b$$

Systems is **overdetermined**, solution almost never exists.

Key Idea

Trying to fit a degree **one** polynomial through **twenty** points is a pipe dream.



Overdetermined Problem

The issue: A^{-1} does not exist because A is not square.

Key Idea

- **Give up** on trying to pass a line through all points
- Find the **line of best fit** instead

Let's rework the problem

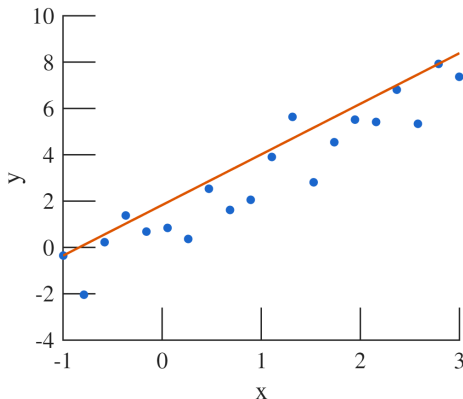
- We still want a line $\hat{y} = mx + b$
- We want our line to be optimal in some way

Question

What makes a line “best fit”?



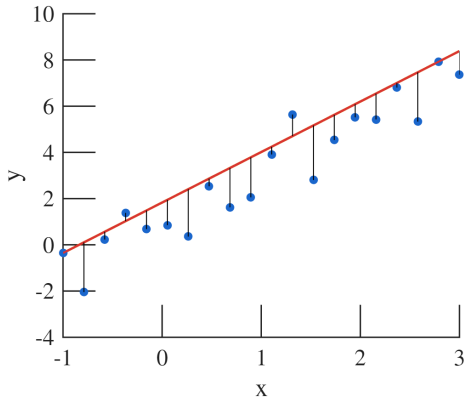
What is the best line?



Question

- Do you think this is the best line?
- How can you tell?
- How can we quantify the goodness of the fit?

Quantify the Error



- For any line, we can add up the vertical error at each point
- We can define error to be

$$E = \sum_i |i^{th} \text{ vertical bar}|^2$$



The Residual

- The length of the vertical lines is called the **residual**

$$r_i = y_i - (mx_i + b)$$

- r_i can be positive or negative. That's why we square them.
- The error is

$$E = \sum_i (r_i)^2$$

- Since the matrix form is

$$\begin{bmatrix} x_1 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{bmatrix} \begin{bmatrix} m \\ b \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

we can write the residual in **vector form**

$$r = y - Ax$$



Least Squares

- Let x represent a particular "fit"

$$x = \begin{bmatrix} m \\ b \end{bmatrix}$$

for the line $y = mx + b$

Definition: Backward error

The **residual** vector is

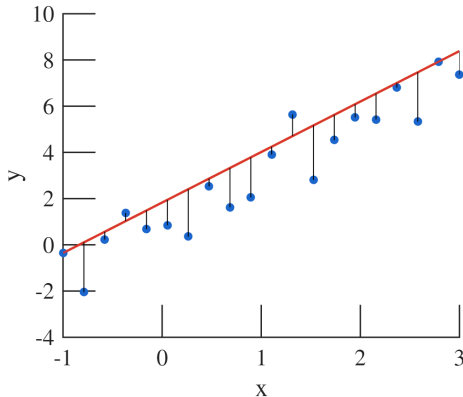
$$r = b - Ax.$$

The norm $\|r\|^2$ represents the **square error**.

Method: Least Squares

- ▶ **Given:** Overdetermined system $Ax = b$
- ▶ **Output:** Best approximate solution x
- ▶ **Idea:** The solution is chosen so that the backward error is minimized.

Optimize



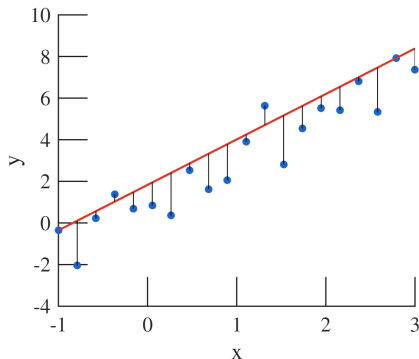
- For any fit $y = mx + b$, we can compute the least squares error

$$E = \sum_i |i^{th} \text{ vertical bar}|^2 = \sum_i (r_i)^2 = r^T r = \|r\|^2$$

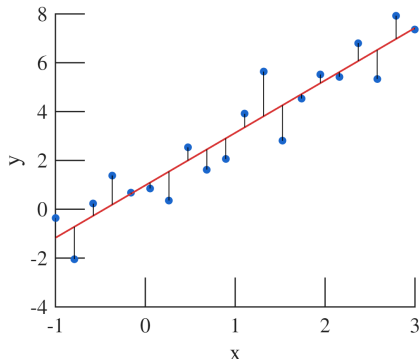
- Goal:** find the values m and b that minimizes the error



Visual



Residual $\|r\|^2 = 31.8624$



Residual $\|r\|^2 = 16.0863$

Key Idea

Minimizing $\|r\|^2$ generates the line that best represents the data.



Normal Equations



How do you find the optimal line?

Objective

Find the line of best fit $y = mx + b$ that represents the data (x_i, y_i) .

Least Squares

Find the **least** value of the sum of **squares** $E = \sum_i (y_i - (mx_i + b))^2$



How do you find the optimal line?

Objective

Find the line of best fit $y = mx + b$ that represents the data (x_i, y_i) .

Least Squares

Find the **least** value of the sum of **squares** $E = \sum_i (y_i - (mx_i + b))^2$

Question

How do accomplish this algorithmically?



- Suppose you want to find the local minimum of $y = f(x)$
- First, find $f'(x)$
- Set $f'(x) = 0$ and then solve for x
- That value of x is the local minimum



- Suppose you want to find the local minimum of $y = f(x)$
- First, find $f'(x)$
- Set $f'(x) = 0$ and then solve for x
- That value of x is the local minimum
- We apply this same strategy to $\|r\|^2$



Least Squares

- We can express $\|r\|_2^2$ in terms of x :

$$\|r\|_2^2 = r^T r = (b - Ax)^T (b - Ax) = \underbrace{b^T b - 2x^T A^T b + x^T A^T A x}_{\phi(x)}$$

- Take the derivative and set it zero: $\nabla \phi(x) = 0$.

$$\nabla \phi(x) = -2A^T b + 2A^T A x = 0$$

- Rearrange

$$2A^T A x = 2A^T b$$

$$A^T A x = A^T b$$

Definition: Normal Equations

For a system $Ax = b$, the **normal equations** is defined to be

$$A^T A x = A^T b.$$

Summary of Least Squares

- 1 Solving an overdetermined system cannot be done

$$Ax = b$$

- 2 We can find the line of best fit

Make the square of the backward error $\|r\|^2$ the least

- 3 This amounts to solving the normal equations

$$A^T Ax = A^T b$$

Advantages:

- Well-defined problem
- Inverse always exists
- We can use CG to solve for x



Example

Example

Find the line of best fit for the data points $(-3, -2)$, $(1, 0)$, $(2, 4)$.

- Let the fit be $y = mx + b$
- Then the system is

$$\begin{array}{rcl} -2 & = & -3m + b \\ 0 & = & m + b \\ 4 & = & 2m + b \end{array} \quad \Rightarrow \quad \begin{bmatrix} -3 & 1 \\ 1 & 1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} m \\ b \end{bmatrix} = \begin{bmatrix} -2 \\ 0 \\ 4 \end{bmatrix}$$

- Calculate $A^T A = \begin{bmatrix} -3 & 1 & 2 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} -3 & 1 \\ 1 & 1 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} 14 & 0 \\ 0 & 3 \end{bmatrix}$
- Calculate $A^T b = \begin{bmatrix} -3 & 1 & 2 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} -2 \\ 0 \\ 4 \end{bmatrix} = \begin{bmatrix} 14 \\ 2 \end{bmatrix}$



Example

- The normal equations are $A^T Ax = A^T b$:

$$\begin{bmatrix} 14 & 0 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} m \\ b \end{bmatrix} = \begin{bmatrix} 14 \\ 2 \end{bmatrix}$$

- Solving the system gives

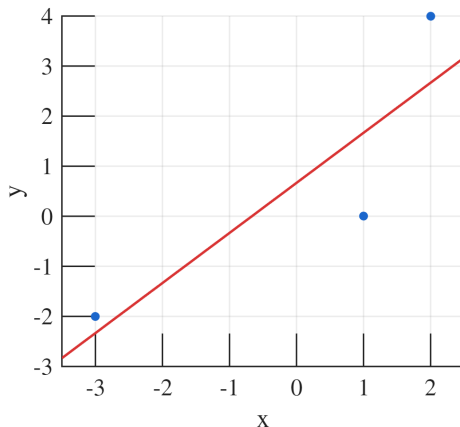
$$m = 1, \quad b = 2/3$$

- The line of best fit is

$$y = x + \frac{2}{3}$$



Example



Example

Line of best fit for $(-3, -2)$, $(1, 0)$, $(2, 4)$ is $y = x + \frac{2}{3}$.

An issue...

Disadvantage:

- The normal equations tend to worsen the condition of the matrix.

Theorem

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

- Recall that $\text{cond}(A) \geq 1$

```
1 >> A = rand(10,10);  
2 >> cond(A)  
3    43.4237  
4 >> cond(A'*A)  
5    1.8856e+03
```

Key Idea

Regression involves solving $A^T A x = A^T b$ which unfortunately has a high condition number. The coefficients of the line may be highly inaccurate.

Least Squares with SVD



Using SVD for least squares

- Recall that a **singular value decomposition** is given by

$$A = \begin{bmatrix} \vdots & \vdots & \vdots \\ u_1 & \dots & u_m \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \sigma_r & \\ & & & \ddots \\ & & & & 0 \end{bmatrix} \begin{bmatrix} \dots & v_1^T & \dots \\ \dots & \vdots & \dots \\ \dots & v_n^T & \dots \end{bmatrix}$$

where σ_i are the singular values.

- U and V are orthogonal $\implies$ they are norm-preserving
- Instead of minimizing $\|r\|^2$, we can minimize $\|U^T r\|^2$



Using SVD for Least Squares

- Assume that A has rank k (and hence k nonzero singular values σ_i) and recall that we want to minimize

$$\|r\|_2^2 = \|b - Ax\|_2^2.$$

- Substituting the SVD for A we find that

$$\|r\|_2^2 = \|b - Ax\|_2^2 = \|b - USV^T x\|_2^2$$

- We leverage the orthogonality of U^T

$$\|b - USV^T x\|_2^2 = \|U^T b - U^T USV^T x\|_2^2 = \|U^T b - SV^T x\|_2^2$$

The New Problem

By using the SVD, we have changed the least squares problem into

$$SV^T x = U^T b$$

The Method

- Let $c = U^T b$ and $y = V^T x$ so the problem is to solve

$$Sy = c$$

- S is a diagonal matrix. It only has k non-zero entries by assumption.
- Solving this system can be done extremely quickly

$$y_i = \frac{c_i}{\sigma_i}, \quad 1 \leq i \leq k$$

Method: SVD Least Squares

- Given:** Overdetermined system $Ax = b$
- Output:** Best approximate solution $\tilde{x}$
- Idea:** Using the SVD let's us avoid the ill-conditioned normal equations.



Theorem

Let A be an $m \times n$ matrix of rank k and let $A = USV^T$, the singular value decomposition. The least squares solution of the system $Ax = b$ is

$$x = \sum_{i=1}^k (\sigma_i^{-1} c_i) v_i$$

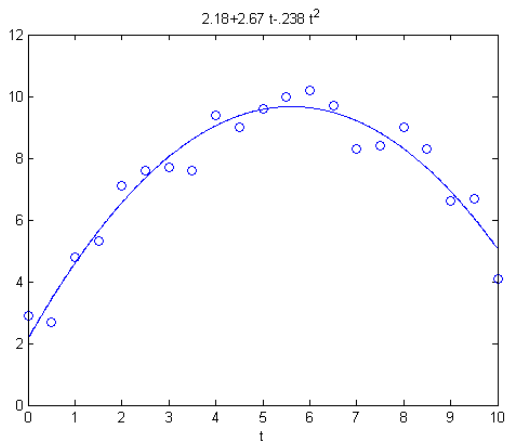
where $c_i = u_i^T b$.



Other Fitting



Fitting a Quadratic



How can we find the quadratic of best fit?



Fitting a Quadratic

- Let's assume our data is quadratic $y = a + bx + cx^2$
- Our given data points are

$$(x_1, y_1)$$

$$(x_2, y_2)$$

$$\vdots$$

$$(x_n, y_n)$$

- Our interpolation system would be

$$y_1 = a + bx_1 + cx_1^2$$

$$y_2 = a + bx_2 + cx_2^2$$

$$\vdots$$

$$y_n = a + bx_n + cx_n^2$$



Fitting a Quadratic

- This is non-square *Vandermonde matrix*

$$\underbrace{\begin{bmatrix} 1 & x_1 & x_2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{bmatrix}}_A \underbrace{\begin{bmatrix} a \\ b \\ c \end{bmatrix}}_x = \underbrace{\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}}_b$$

- Since this is an *over-determined system* we solve the normal equation

$$A^T A x = A^T b$$

Key Idea

- ▶ The actual model does not matter
- ▶ It always boils down to solving $A^T A x = A^T b$



Common Regression Models

1. Linear Regression

$$y = ax + b$$

2. Quadratic Regression

$$y = ax^2 + bx + c$$

3. Cubic Regression

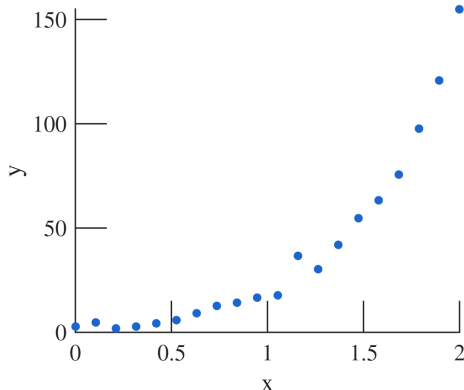
$$y = ax^3 + bx^2 + cx + d$$



Nonlinear Models

Question

What do you do if your data appears to be exponential?



- There's an issue with the model

$$y = ae^{bx}$$



Nonlinear Models

- Fitting data on the model $y = ae^{bx}$ yields

$$y_1 = ae^{bx_1}$$

$$y_2 = ae^{bx_2}$$

$$\vdots$$

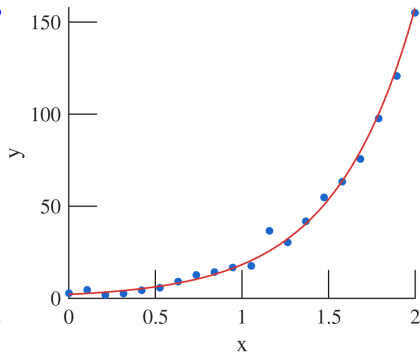
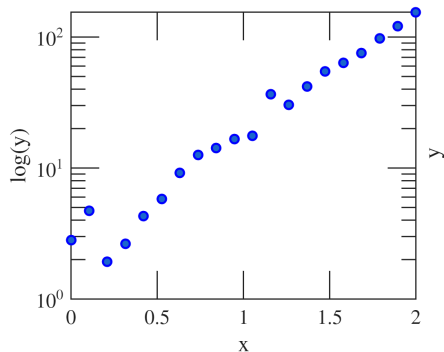
$$y_n = ae^{bx_n}$$

- **Problem:** this is *not* a linear system
- **Fix:** Use a log transformation

$$\ln(y) = \ln(ae^{bx}) = \ln a + bx$$



Transform the Data



- Fit the data on

$$\ln(y) = \ln a + bx$$

- Undo the transformation

$$y = ae^{bx}$$



Summary

- Fitting a function to noisy data leads to an overdetermined system

$$Ax = b,$$

where A is $m \times n$ and m is much larger than n

- We can instead solve

$$A^T Ax = A^T b$$

to find the "best-fit" equation

- In practice, it's better to compute $A = USV^T$ and solve

$$SV^T x = U^T b$$

