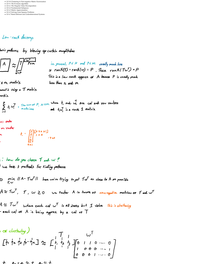


3.1 Low-rank decomp

- perfect pattern. By taking up rank multiplies

A is an $M \times N$ matrix
and is symmetric: $M = N$ matrix
and v_i matrix

$v_i v_i^T = \frac{1}{\|v_i\|^2} v_i v_i^T$ — projection of v_i onto v_i (line v_i and all v_i are orthogonal)

λ_1 is an eigenvalue
v_1 is an eigenvector
$\lambda_2 = \lambda_3 = \lambda_4$
v_2, v_3, v_4 — λ_2 eigenvalue

$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$

Question: how do you choose λ and v?
the SVD can be a method for finding solutions

1) find λ_1 λ_2 λ_3 λ_4 v_1 v_2 v_3 v_4 — choose the 4 most popular

2) $A = U \Sigma V^T$, $U = I$ $\Sigma = D$ — we make A a lower-rank matrix or U and V^T

3) $A = U \Sigma V^T$, where each row of V^T is the popular part of Σ value. U is identity
U and V are $M \times N$ being square by Σ and V^T

Example or identity:

$A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \approx \frac{1}{2} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}$

$\overset{U}{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}} \overset{V^T}{\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}$

So $A = I_4$ $\lambda_1 = 3$ $\lambda_2 = 1$ $\lambda_3 = 1$ $\lambda_4 = 1$

- A matrix is rank-1 if it is any linear combination of rank-1 matrix
- $A = \lambda_1 v_1 v_1^T + \lambda_2 v_2 v_2^T + \lambda_3 v_3 v_3^T + \lambda_4 v_4 v_4^T$
- $v_1 v_1^T$ — rank-1 matrix
- λ_1 — weight

$A_{M \times N} = U_{M \times M} \Sigma_{M \times N} V^T_{N \times N}$

$U^T U = I_M$
$V^T V = I_N$

$X = U \Sigma V^T$
$X = V_{\text{new}}$

$U_{M \times M}$ is an orthonormal basis
$V_{N \times N}$ is an orthonormal basis
$\Sigma_{M \times N}$ is a diagonal matrix

$\Sigma_{M \times N} = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_N \\ 0 & \cdots & 0 \end{bmatrix}$ — $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N \geq 0$

$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N \geq 0$

$\lambda_1 = \sigma_1^2$
$\lambda_2 = \sigma_2^2$
$\lambda_3 = \sigma_3^2$
$\lambda_4 = \sigma_4^2$

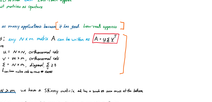
$\Sigma = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \sigma_3 & \\ & & & \sigma_4 \end{bmatrix}$

$\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \sigma_4 \geq 0$

3.2 Clustering data with K-means algorithm

we have m n-dimensional data points
and we want to find centroids $\mu_1, \dots, \mu_k \in \mathbb{R}^n$ such
that each m data points is closer to its closest centroid

example: we pick the number clusters and the location of their centroids



$\mu_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

we split into 2 clusters

$\mu_1 = [1, 1, 1, 1, 1, 1, 1]$, $\mu_2 = [2, 2, 2, 2, 2, 2, 2]$

$\mu_1 = [0, 0, 0, 0, 0, 0, 0]$, $\mu_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \end{bmatrix}$

$T = [\mu_1, \mu_2]$, $U = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 \end{bmatrix}$ $\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}_{2 \times 7}$ $\begin{cases} C = \mu_1 \\ C = \mu_2 \end{cases}$

- here is clusters T: $T(1) = \mu_1$, $T(2) = \mu_2$
- the method is the same as the data points is closer $\frac{\|x - \mu_1\|^2}{\|x - \mu_2\|^2}$

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

- relabeling: $c_i = \underset{1 \leq j \leq k}{\arg\min} \|x_i - \mu_j\|^2$ — read step
- $\mu_j = \frac{1}{|C_j|} \sum_{i \in C_j} x_i$ — update centroids

$T_{\text{new}} = X$

how do convergence

$C = \sum_{j=1}^{k} c_j = \sum_{j=1}^{k} \sum_{i \in C_j} \|x_i - \mu_j\|^2$ $\quad \|x_i - \mu_j\|^2$

How to implement K-means algorithm

1.1 Initialize: choose $\mu_1^0, \dots, \mu_k^0 \in \mathbb{R}^n$ randomly from $x_1, \dots, x_m$

2.1 Assignment: for $i = 1, \dots, m$ c_i is closest to μ_j^0

2.2 update centroids: $\mu_j^{t+1} = \frac{\sum_{i=1}^{m} 1\{c_i = j\} x_i}{\sum_{i=1}^{m} 1\{c_i = j\}}$

repeat steps 1 and 2

3) is convergence: $J(\mu_1, \dots, \mu_k)$ not decreasing

- everything you add a other clusters you have to recalculate all the data

Mixture data for clustering and soft clustering recalculate

This algorithm is greedy algorithm
- but you initialize it a many times and we know optimal could be reached
- the local maxima maximum and the best that the algorithm can finally converge

- the clusters and the centroids are a local max
- you can choose the mean

Finally: SVD 3.3

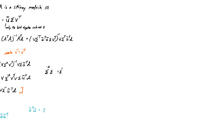
- Definition singular value decomposition (SVD)
- Regression efficiency SVD
- Using SVD with the inverse and matrix products
- Low-rank approximation: best low-rank approx
- Eckhart-Young theorem and Mirsky

SVD are many applications because it has good low-rank approx

Theorem: any $M \times N$ matrix A can be written as $A = U \Sigma V^T$
where
- U is $M \times M$ orthogonal matrix
- V is $N \times N$ orthogonal matrix
- Σ is $M \times N$ diagonal Σ with
  singular value on the first $\min(M, N)$ diagonal

- if $M \geq N$: we have a tall matrix, we have a lot of zero rows and the bottom
- if $M \leq N$: we have a fat matrix, we have a lot zero rows on the right

- U is of dimension $M \times M$ orthogonal
- V is of dimension $N \times N$ orthogonal
- Σ is of dimension $M \times N$ diagonal
- Σ has the same dimension as A
- Σ has the same singular value as A (the singular value of A)



U is of dimension $M \times M$
(has both of right singular vectors
which are M many rows or V^T)

orthogonal V and U have the same dimension

for tall matrix
$A = \sum_{i=1}^{N} \sigma_i u_i v_i^T = \sum_{i=1}^{N} \sigma_i u_i v_i^T$

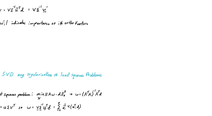
$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_N \geq 0$

$U^T U = I_N$
$V^T V = I_N$
$U^T U = I_{N \times N}$
$V^T V = I_{N \times N}$

$A = \sum_{i=1}^{N} u_i$

the rank you can get low-rank approx
$\text{rank}(A) = \|\sigma\|_0$ (norm)

$A = \sum_{i=1}^{N} u_i v_i^T$
and u_i is eigenvector in A
v_i is A by u_i



multiplication
$y = Ax = U \Sigma V^T x = U \Sigma (V^T x)$

Eigenvector theorem: the max amount you can multiply a vector by $= \sigma_1$

3.3 Properties of SVD

$A = U \Sigma V^T$ where $A \in \mathbb{R}^{M \times N}$
$U \in \mathbb{R}^{M \times M}$
$V \in \mathbb{R}^{N \times N}$

- if $\text{rank}(A) \leq r$ (then A is singular and $\text{null}(A) = \text{null}(A)$)
- $\text{rank}(A) = r$, then $A^+ = V \Sigma^+ U^T$

3.5 SVD and least-squares problems

Review of least squares problem:

we want to design a solution of an
linear system

$\min_x \|Ax - b\|_2^2$

$A = m \times n$ or $m \geq n$

so then $y = (A^T A)^{-1} A^T b$

and the error is $\|r\|_2$

e.g. A is a strong matrix so

$A = \tilde{U} \tilde{\Sigma} V^T$
$\tilde{U}$ is tall square matrix

$y = (V \tilde{\Sigma}^T \tilde{U}^T \tilde{U} \tilde{\Sigma} V^T)^{-1} V \tilde{\Sigma}^T \tilde{U}^T b$

$\tilde{U}^T \tilde{U} = I$

$y = (V \tilde{\Sigma}^T \tilde{\Sigma} V^T)^{-1} V \tilde{\Sigma}^T \tilde{U}^T b$

$= V \tilde{\Sigma}^{-1} \tilde{U}^T b$

$\tilde{\Sigma}^T \tilde{\Sigma} = \tilde{\Sigma}^{-1}$

so $V \tilde{\Sigma}^{-1} \tilde{U}^T b$]

$\tilde{\Sigma}^2 = I$

- $\tilde{\Sigma} = \tilde{\Sigma}^{-1}$

how to identify using SVD orthonormal basis

we start with finding data. so we can get so

$\begin{bmatrix} \tilde{U} \\ \tilde{U} \end{bmatrix} w = \begin{bmatrix} b \\ 0 \end{bmatrix}$ — orthogonal matrix
$\Rightarrow$ $\text{Proj}_{\tilde{U}} b$
$\Downarrow$
get y and use it to predict
the figure $\tilde{y} = \text{diag}(\tilde{\Sigma}^{-1} \tilde{w})$

- $A = \tilde{U} \tilde{\Sigma} V^T$
- $\tilde{U}^T w = b$
- $w = V \tilde{\Sigma}^{-1} \tilde{U}^T b = V \tilde{\Sigma}^{-1} \tilde{y}$

full: identity orthogonal or $\tilde{U}$ with zero

4.1 SVD and regularization in least squares problem

least squares problem: $\min_x \|Ax - b\|_2^2 = w = (A^T A)^{-1} A^T b$

$A = U \Sigma V^T$ so $w = V \tilde{\Sigma}^{-1} \tilde{U}^T = \sum_{i=1}^{r} \frac{1}{\sigma_i} v_i (u_i^T b)$

if without 1.1 low rank or high norm data
this is best because it means the same singular amplitude

- if $\text{rank}(A) = r$ then w_i can blow up single σ_i

b)

$f(x) = \frac{1}{2} \log(1 + e^{\frac{x - \mu}{s}}) + \frac{1}{2} \|w\|_2^2$

$\frac{1}{2} \frac{1}{s} \left(\log(1 + e^{\frac{x - \mu}{s}}) \right)$

$\frac{1}{1 + e^{\frac{x - \mu}{s}}} \left(e^{\frac{x - \mu}{s}} \right) \left(\frac{1}{s} \right)$

$\nabla_w \left(\frac{1}{2} \log(1 + e^{w^T x}) \right) + \nabla \|w\|_2^2$

$= \frac{1}{2} \left(\frac{e^{w^T x}}{(1 + e^{w^T x})} \right) + 2\lambda w$

4.2 Principal component analysis PCA

- Σ is part of Λ is eigenvalue of covariance

PCA assume (zero mean)
center data around the mean

Σ: total data $= \Lambda = \frac{1}{m} \sum_{i=1}^{m} x_i x_i^T$ — mean centered data

The first PC is the direction v_1 maximizing the
projection variance when $\|v_1\|_2^2 = 1$

$\text{Var}(v_1^T x) = v_1^T \Sigma v_1$

$\text{Var}(v_1^T x) = \text{Var}(\frac{1}{m} \sum_{i=1}^{m} x_i)$

$\max_{\|v_1\|=1} \left(\frac{1}{m} \sum_{i=1}^{m} x_i v_1^T \right) = \frac{1}{m} \sum_{i=1}^{m} x_i v_1^T \left(\frac{1}{m} \sum_{i=1}^{m} x_i v_1^T \right)$

The line that accounts for most variance is

$x_i = \Sigma^T x_i$

$x_i^T = x_i \Sigma^T$

The line associated with the first PC is $\frac{\Sigma^T}{\|\Sigma^T\|}$

The 2nd PC is $v_2 \perp v_1$ (and all singular vectors)

$\|v_1\|_2^2 = 1$, $v_1 \perp v_2$

v_1, v_2 are orthogonal

$\text{Var}(v_2 x_i) \leq \frac{\Sigma_2}{m}$

4.3 Eckart-Young-Mirsky: the proof

4.4 Spectral clustering

Take social media and form a graph where nodes
$G_1 \to \{1, 2, 4\}$
$G_2 \to \{3, 5\}$
$B = E D E^T = \frac{1}{2} v_2 v_2^T$
$B = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$

$v_2 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}$