# CSCI B609:
# “Foundations of Data Science”

## Lecture 8/9: Faster Power Method and Applications of SVD

Slides at http://grigory.us/data-science-class.html

**Grigory Yaroslavtsev**

**http://grigory.us**

# Faster Power Method

- PM drawback: $A^T A$ is dense even for sparse $A$
- Pick random Gaussian $\boldsymbol{x}$ and compute $B^k \boldsymbol{x}$
- $\boldsymbol{x} = \sum_{i=1}^{\boldsymbol{d}} c_i \boldsymbol{v}_i$(augment $\boldsymbol{v_i}$'s to o.n.b. if $r < \boldsymbol{d}$)
- $B^k \boldsymbol{x} \approx \left(\sigma_1^{2k} \boldsymbol{v}_1 \boldsymbol{v}_1^T\right)\left(\sum_{i=1}^{d} c_i \boldsymbol{v}_i\right) = \ \sigma_1^{2k} c_1 \boldsymbol{v}_1$

$$B^k \boldsymbol{x} = (A^T A)(A^T A) \ldots (A^T A)\boldsymbol{x}$$

- **Theorem:** If $\boldsymbol{x}$ is unit $\mathbb{R}^{\boldsymbol{d}}$-vector, $|\boldsymbol{x}^T \boldsymbol{v}_1| \geq \boldsymbol{\delta}$:
  - $V$ = subspace spanned by $\boldsymbol{v}_i' s$ for $\sigma_j \geq (1 - \boldsymbol{\epsilon})\sigma_1$
  - $\boldsymbol{w}$ = unit vector after $k = \frac{1}{2\boldsymbol{\epsilon}} \ln\left(\frac{1}{\boldsymbol{\epsilon\delta}}\right)$ iterations of PM

$\Rightarrow$ $\boldsymbol{w}$ has a component at most $\boldsymbol{\epsilon}$ orthogonal to $V$

# Faster Power Method: Analysis

- $A = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$ and $\boldsymbol{x} = \sum_{i=1}^{\boldsymbol{d}} c_i \boldsymbol{v}_i$
- $B^k \boldsymbol{x} = \sum_{i=1}^{\boldsymbol{d}} \sigma_i^{2k} \boldsymbol{v}_i \boldsymbol{v}_i^T \sum_{j=1}^{\boldsymbol{d}} c_j \boldsymbol{v}_j = \sum_{i=1}^{\boldsymbol{d}} \sigma_i^{2k} c_i \boldsymbol{v}_i$

$$\left|\left|B^k \boldsymbol{x}\right|\right|_2^2 = \left\|\sum_{i=1}^{\boldsymbol{d}} \sigma_i^{2k} c_i \boldsymbol{v}_i\right\|_2^2 = \sum_{i=1}^{\boldsymbol{d}} \sigma_i^{4k} c_i^2 \geq \sigma_1^{4k} c_1^2 \geq \sigma_i^{4k} \delta^2$$

- (Squared ) component orthogonal to $V$ is

$$\sum_{i=m+1}^{\boldsymbol{d}} \sigma_i^{4k} c_i^2 \leq (1-\boldsymbol{\epsilon})^{4k} \sigma_1^{4k} \sum_{i=m+1}^{\boldsymbol{d}} c_i^2 \leq \ (1-\boldsymbol{\epsilon})^{4k} \sigma_1^{4k}$$

- Component of $\boldsymbol{w} \perp V \leq (1-\boldsymbol{\epsilon})^{2k} / \boldsymbol{\delta} \leq \boldsymbol{\epsilon}$

# Choice of $\boldsymbol{x}$

- $\boldsymbol{y}$ random spherical Gaussian with unit variance
- $\boldsymbol{x} = \frac{\boldsymbol{y}}{||\boldsymbol{y}||_2}$:

$$Pr\left[\left|\boldsymbol{x}^T\boldsymbol{v}\right| \leq \frac{1}{20\sqrt{\boldsymbol{d}}}\right] \leq \frac{1}{10} + 3e^{-\boldsymbol{d}/64}$$

- $Pr\left[||\boldsymbol{y}||_2 \geq \boldsymbol{2}\sqrt{\boldsymbol{d}}\right] \leq 3e^{-\boldsymbol{d}/64}$ (Gaussian Annulus)
- $\boldsymbol{y}^T\boldsymbol{v} \sim N(0,1) \Rightarrow \Pr\left[\left|\left|\boldsymbol{y}^T\boldsymbol{v}\right|\right|_2 \leq \frac{1}{10}\right] \leq \frac{1}{10}$
- Can set $\boldsymbol{\delta} = \frac{1}{20\sqrt{\boldsymbol{d}}}$ in the "faster power method"

# Singular Vectors and Eigenvectors

- Right singular vectors are eigenvectors of $A^T A$
- $\sigma_i^2$ are eigenvalues of $A^T A$
- Left singular vectors are eigenvectors of $AA^T$
- $A^T A$ satisfies $\forall \boldsymbol{x}$: $\boldsymbol{x}^T B \boldsymbol{x} \geq 0$
  - $B = \sum_i \sigma_i^2 \boldsymbol{v}_i \boldsymbol{v}_i^T$
  - $\forall \boldsymbol{x}$: $\boldsymbol{x}^T \boldsymbol{v}_i \boldsymbol{v}_i^T \boldsymbol{x} = (\boldsymbol{x}^T \boldsymbol{v}_i)^2 \geq 0$
  - Such matrices are called positive semi-definite
- Any p.s.d matrix can be decomposed as $A^T A$

# Application of SVD: Centering Data

- Minimize sum of squared distances from $\boldsymbol{A_i}$ to $S_k$
- **SVD**: best fitting $S_k$ if data is centered
- What if not?
- **Thm.** $S_k$ that minimizes squared distance goes through centroid of the point set:
$$\frac{1}{n}\sum \boldsymbol{A_i}$$
- Will only prove for $k = 1$, analogous proof for arbitrary $k$ (see textbook)

# Application of SVD: Centering Data

- **Thm.** Line that minimizes squared distance goes through the centroid
- Line: $\ell = \boldsymbol{a} + \lambda \boldsymbol{v}$; distance $dist(\boldsymbol{A_i}, \ell)$
- $||\boldsymbol{A_i} - \boldsymbol{a}||_2^2 = dist(\boldsymbol{A_i}, \ell)^2 + \langle \boldsymbol{v}, \boldsymbol{A_i} \rangle^2$
- Center so that $\sum_{i=1}^{n} \boldsymbol{A_i} = \boldsymbol{0}$ by subtracting the centroid
- $\sum_i^n dist(\boldsymbol{A_i}, \ell)^2 = \sum_{i=1}^{n} (||\boldsymbol{A_i} - \boldsymbol{a}||_2^2 - \langle \boldsymbol{v}, \boldsymbol{A_i} \rangle^2)$

$$= \sum_{i=1}^{n} (||\boldsymbol{A_i}||_2^2 + ||\boldsymbol{a}||_2^2 - 2\langle \boldsymbol{A_i}, \boldsymbol{a} \rangle - \langle \boldsymbol{v}, \boldsymbol{A_i} \rangle^2)$$

$$= \sum_{i=1}^{n} ||\boldsymbol{A_i}||_2^2 + n||\boldsymbol{a}||_2^2 - 2\langle \sum_{i=1}^{n} \boldsymbol{A_i}, \boldsymbol{a} \rangle - \sum_{i=1}^{n} \langle \boldsymbol{v}, \boldsymbol{A_i} \rangle^2$$

$$= \sum_{i=1}^{n} ||\boldsymbol{A_i}||_2^2 + n||\boldsymbol{a}||_2^2 - \sum_{i=1}^{n} \langle \boldsymbol{v}, \boldsymbol{A_i} \rangle^2$$

- Minimized when $\boldsymbol{a} = \boldsymbol{0}$

# Principal Component Analysis

- $\boldsymbol{n} \times \boldsymbol{d}$ matrix: customers×movies preference
- $\boldsymbol{n}$ = #customers, $\boldsymbol{d}$ = #movies
- $A_{ij} =$ how much customer $i$ likes movie $j$
- Assumption: $A_{ij}$ can be described with $k$ factors
  - Customers and movies: vectors in $\boldsymbol{u_i}$ and $\boldsymbol{v_i} \in \mathbb{R}^k$
  - $A_{ij} = \langle \boldsymbol{u_i}, \boldsymbol{v_j} \rangle$
- Solution: $A_k$

$$\text{customers} \left(\begin{array}{c} \\ \\ A \\ \\ \\ \end{array}\right) = \begin{array}{c} \text{factors} \\ \left(\begin{array}{c} \\ \\ U \\ \\ \\ \end{array}\right) \end{array} \begin{array}{c} \text{movies} \\ \left(\begin{array}{c} V \end{array}\right) \end{array}$$

# Class Project

- **Survey of 3-5 research papers**
  - Closely related to the topics of the class
    - Algorithms for high-dimensional data
    - Fast algorithms for numerical linear algebra
    - Algorithms for machine learning and/or clustering
    - Algorithms for streaming and massive data
  - Office hours if you need suggestions
  - Individual (not a group) project
  - **1-page Proposal Due: October 31, 2016 at 23:59 EST**
  - **Final Deadline**: **December 09, 2016 at 23:59 EST**
- Submission by e-mail to **Lisul Islam (IU id: islammdl)**
  - Submission Email Title: Project + Space + “Your Name”
  - Submission format: **PDF from LaTeX**

# Separating mixture of $k$ Gaussians

- **Sample origin problem**:
  - Given samples from $\boldsymbol{k}$ **well-separated** spherical Gaussians
  - **Q**: Did they come from the same Gaussian?
- $\delta$ = distance between centers
- For two Gaussians naïve separation requires

$$\delta > \omega\left(\boldsymbol{d}^{1/4}\right)$$

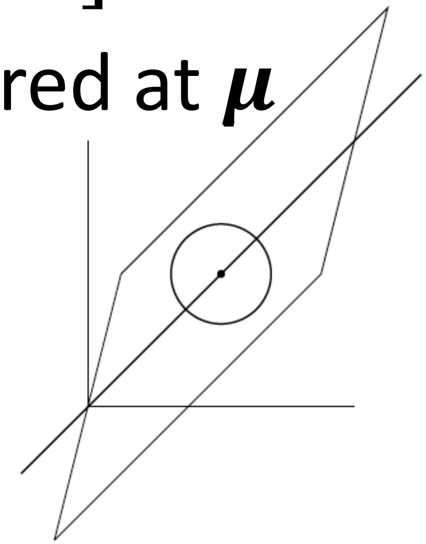
- **Thm.** $\delta = \Omega(\boldsymbol{k}^{\frac{1}{4}})$ suffices
- **Idea:**
  - Project on a $\boldsymbol{k}$-dimensional subspace through centers
  - **Key fact:** This subspace can be found via SVD
  - Apply naïve algorithm

# Separating mixture of $k$ Gaussians

- **Easy fact:** Projection preserves the property of being a unit-variance spherical Gaussian
- **Def.** If $p$ is a probability distribution, **best fit line** $\{c\boldsymbol{v}, c \in \mathbb{R}\}$ is:

$$\boldsymbol{v} = argmax_{|\boldsymbol{v}|=1} \mathbb{E}_{\boldsymbol{x}\sim p}\left[\left(\boldsymbol{v}^{\boldsymbol{T}}\boldsymbol{x}\right)^{\boldsymbol{2}}\right]$$

- **Thm:** Best fit line for a Gaussian centered at $\boldsymbol{\mu}$ passes through $\boldsymbol{\mu}$ and the origin



# Best fit line for a Gaussian

- **Thm:** Best fit line for a Gaussian centered at $\boldsymbol{\mu}$ passes through $\boldsymbol{\mu}$ and the origin

$$\mathbb{E}_{\boldsymbol{x}\sim p}\left[\left(\boldsymbol{v}^T\boldsymbol{x}\right)^{\mathbf{2}}\right] = \mathbb{E}_{\boldsymbol{x}\sim p}\left[\left(\boldsymbol{v}^T(\boldsymbol{x}-\boldsymbol{\mu}) + \boldsymbol{v}^T\boldsymbol{\mu}\right)^{\mathbf{2}}\right]$$

$$= \mathbb{E}_{\boldsymbol{x}\sim p}\left[\boldsymbol{v}^T(\boldsymbol{x}-\boldsymbol{\mu})^{\mathbf{2}} + 2(\boldsymbol{v}^T\boldsymbol{\mu})\boldsymbol{v}^T(\boldsymbol{x}-\boldsymbol{\mu}) + (\boldsymbol{v}^T\boldsymbol{\mu})^2\right]$$

$$= \mathbb{E}_{\boldsymbol{x}\sim p}\left[\boldsymbol{v}^T(\boldsymbol{x}-\boldsymbol{\mu})^{\mathbf{2}}\right] + 2(\boldsymbol{v}^T\boldsymbol{\mu})\mathbb{E}_{\boldsymbol{x}\sim p}[\boldsymbol{v}^T(\boldsymbol{x}-\boldsymbol{\mu})] + (\boldsymbol{v}^T\boldsymbol{\mu})^2$$

$$= \mathbb{E}_{\boldsymbol{x}\sim p}\left[\boldsymbol{v}^T(\boldsymbol{x}-\boldsymbol{\mu})^{\mathbf{2}}\right] + (\boldsymbol{v}^T\boldsymbol{\mu})^2$$

$$= \sigma^2 + (\boldsymbol{v}^T\boldsymbol{\mu})^2$$

- Where we used:
  - $\mathbb{E}_{\boldsymbol{x}\sim p}[\boldsymbol{v}^T(\boldsymbol{x}-\boldsymbol{\mu})] = \mathbf{0}$
  - $\mathbb{E}_{\boldsymbol{x}\sim p}[\boldsymbol{v}^T(\boldsymbol{x}-\boldsymbol{\mu})^{\mathbf{2}}] = \sigma^2$
- Best fit line maximizes $(\boldsymbol{v}^T\boldsymbol{\mu})^2$

# Best fit subspace for one Gaussian

- Best fit $k$-dimensional subspace $\boldsymbol{V}_k$:

$$\boldsymbol{V}_k = \underset{\boldsymbol{V}: dim(\boldsymbol{V}) = k}{argmax} \; \mathbb{E}_{\boldsymbol{x} \sim p} \left[ ||proj(\boldsymbol{x}, \boldsymbol{V})||_2^2 \right]$$

- For a spherical Gaussian $\boldsymbol{V}$ is a best-fit $k$-dimensional subspace **iff** it contains $\boldsymbol{\mu}$
- If $\boldsymbol{\mu} = 0$ then any $k$-dim. subspace is best fit
- If $\boldsymbol{\mu} \neq 0$ then best fit line $\boldsymbol{v}$ goes through $\boldsymbol{\mu}$
  - Same greedy process as SVD projects on $\boldsymbol{v}$
  - After projection we have Gaussian with $\boldsymbol{\mu} = 0$
  - Any $(k-1)$-dimensional subspace would do

# Best fit subspace for $k$ Gaussians

- **Thm.** $\boldsymbol{p}$ is a mixture of $k$ spherical Gaussians $\Rightarrow$ best fit $k$-dim. subspace contains their centers
- $p = w_1\boldsymbol{p}_1 + w_2\boldsymbol{p}_2 + \cdots + w_k\boldsymbol{p}_k$
- Let $\boldsymbol{V}$ be a subspace of dimension $\leq k$

$$\mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{p}}\left[\left|\left|proj(\boldsymbol{x},\boldsymbol{V})\right|\right|_2^2\right] = \sum_{i=1}^{k} w_i\, \mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{p}_i}\left[\left|\left|proj(\boldsymbol{x},\boldsymbol{V})\right|\right|_2^2\right]$$

- Each term is maximized if $\boldsymbol{V}$ contains all $\boldsymbol{\mu}_{\boldsymbol{i}}'s$
- If we only have a finite number of samples then accuracy has to be analyzed carefully

# HITS Algorithm for Hubs and Authorities

- Document ranking: project on 1st singular vector
- WWW: directed graph with links = edges
- $\boldsymbol{n}$ Authorities: pages containing original info
- $\boldsymbol{d}$ Hubs: collections of links to authorities
  - Authority depends on importance of pointing hubs
  - Hub quality depends on how authoritative links are
- Authority vector: $\boldsymbol{v_j}, j = 1, \ldots, \boldsymbol{n}$: $\boldsymbol{v_j} \sim \sum_{i=1}^{\boldsymbol{d}} \boldsymbol{u}_i \boldsymbol{A}_{ij}$
- Hub vector: $\boldsymbol{u_i}, i = 1, \ldots, \boldsymbol{d}$: $\boldsymbol{u_i} \sim \sum_{j=1}^{\boldsymbol{n}} \boldsymbol{v}_j \boldsymbol{A}_{ij}$
- Use power method: $\boldsymbol{u} = \boldsymbol{A}\boldsymbol{v}, \boldsymbol{v} = \boldsymbol{A}^T \boldsymbol{u}$
- Converges to first left/right singular vectors

# Exercises

- Ex. 1: $A$ is $n \times n$ matrix with orthonormal rows
  - Show that it has orthonormal columns
- Ex. 2: Interpret the left and right singular vectors of the document x term matrix
- Ex. 3. Use power method to compute singular values of the matrix:

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$