Last time

LSH

- important technique for solving approx nearest neighbor queries
- idea: construct hash fns that are likely to map similar items to same bucket

Today

PCA
principal components analysis

data dependent dimensionality reduction

Credit for figures:
Roughgarden & Valiant
John Benedetto
Novembre et al
Alex Williams
Sandipan Dey
Leskovec, Rajaraman, Ullman slides

data set $x_1, \dots, x_m$
$x_i \in \mathbb{R}^n$

$n = 4$

	kale	taco bell	sashimi	pop tarts
x_1 Alice	10	1	2	7
x_2 Bob	7	2	1	10
x_3 Carolyn	2	9	7	3
x_4 Dave	3	6	10	2

$m = 4$

a_{i1}	a_{i2}
1	1
1	-1
-1	1
-1	-1

$v_1 = (3, -3, -3, 3)$

$v_2 = (1, -1, 1, -1)$

Table 1: Your friends' ratings of four different foods.

$\bar{x} = (5.5, 4.5, 5, 5.5)$

$$x_i \approx \bar{x} + a_{i1} v_1 + a_{i2} v_2$$

$$\bar{x} + 1 \cdot v_1 + 1 \cdot v_2 = (9.5, 0.5, 3, 7.5)$$

	kale	taco bell	sashimi	pop tarts
$x_1 - \bar{x}$ Alice	4.5	-3.5	-3	1.5
$x_2 - \bar{x}$ Bob	1.5	-3	-4	4.5
Carolyn	-3.5	4.5	2	-2.5
$x_4 - \bar{x}$ Dave	-2.5	1.5	5	-3.5

$$\sum (x_i - \bar{x}) = 0$$

	kale	taco bell	sashimi	pop tarts
Alice	1.5	-0.5	0	-2.5
Bob	-2.5	0	-1	1.5
Carolyn	-0.5	1.5	-1	1.5
Dave	0.5	-1.5	2	-0.5

$x_1, \ldots, x_m$

represent them approx. $\forall i$

$$x_i \approx \sum_{j=1}^{k} a_{ij} v_j$$

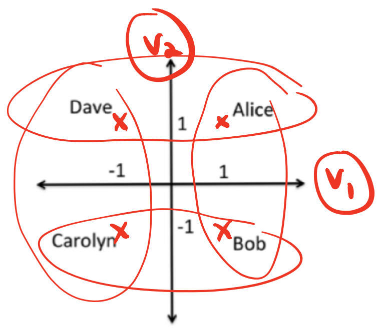
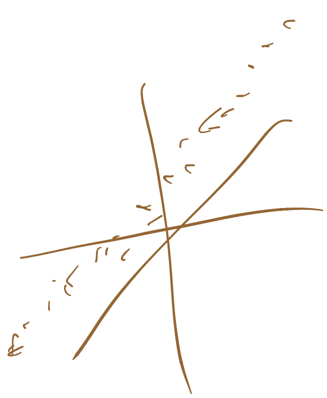
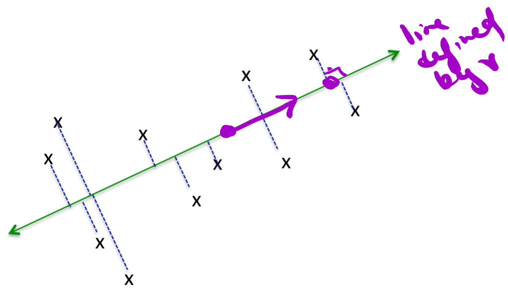


Figure 1: Visualizing 4-dimensional data in the plane.

JL dim reduction	vs	PCA
- cares about preserving distances		doesn't
- coordinates had no meaning		goal to find meaning in vectors $v_1, \ldots, v_k$
- $\frac{c \log n}{\varepsilon^2}$ dimensions to preserve distances $1 \pm \varepsilon$		useful even if $k = 1, 2$



goal: to find "intrinsic" dimensionality



PCA: preprocessing

1) subtract out mean

2) from a practical perspective scale coordinates.

$$x_{ij} = \frac{x_{ij}}{\sqrt{\sum_{k=1}^{m} x_{kj}^2}}$$

$\boxed{k=1:}$

Objective: choose $\vec{v}$ $\|\vec{v}\|=1$

so as to minimize

$$\text{minimize } \frac{1}{m}\sum_{i=1}^{m}\left[\text{dist}(x_i \text{ to line defined by } v)\right]^2 \equiv$$

Claim: this is the same thing as choosing v of length 1 to maximize $\frac{1}{m}\sum_{i=1}^{m}(x_i, v)^2$

maximize

$$\sum_{i=1}^{m} \quad (x_i, v)^2 + \text{dist}^2 = \|x_i\|^2$$

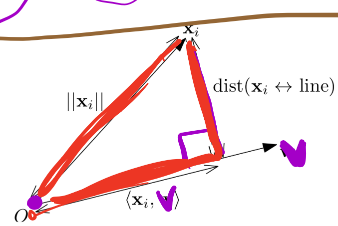
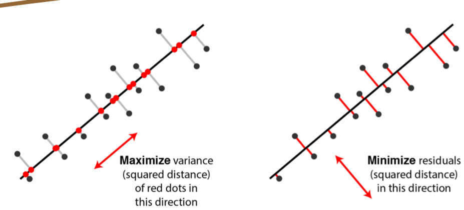


Figure 4: The geometry of the inner product.



Two equivalent views of principal component analysis.

$$X = (x_i, v) \text{ w/ prob } \frac{1}{m}$$

$E(X) = 0 \qquad \frac{1}{m}\sum (x_i, v) = \frac{1}{m}\left(\sum x_i, v\right) = 0$

$Var(X) = E((x-\mu)^2) = E(x^2)$

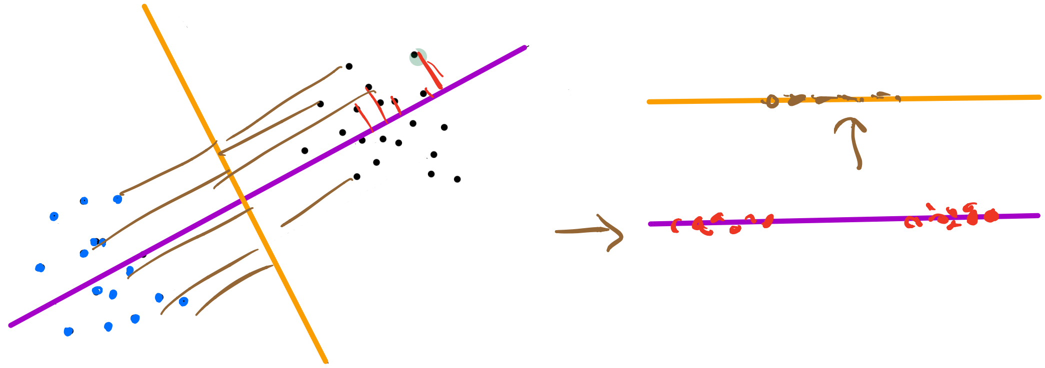


Figure 5: For the good line, the projection of the points onto the line keeps the two clusters separated, while the projection onto the bad line merges the two clusters.

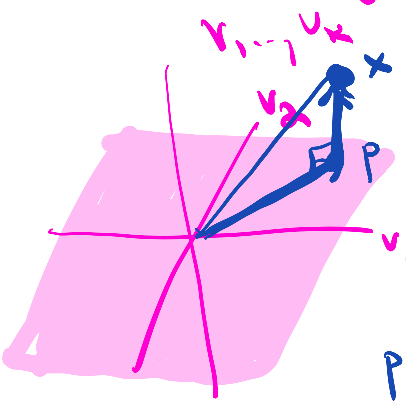
larger k objective

find k-dim subspace S so as to max $\frac{1}{m}\sum_{i=1}^{m}\left(\text{length of } x_i\text{'s projection to } S\right)^2$

Find set of k orthonormal vectors.

$v_1, \dots, v_k \qquad \|v_i\|^2 = 1 \qquad (v_i, v_j) = 0 \ \forall i \neq j$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$



subspace set of all vectors that can be written as comb $\sum_{j=1}^{k} c_j v_j$

$p = a_1 v_1 + a_2 v_2$

$(x - p, v_1) = 0$

$u \cdot v \quad (u, v) \quad u^T v$

$(x, v_1) = (p, v_1) = (a_1 v_1 + a_2 v_2, v_1)$
$= a_1 \underbrace{(v_1, v_1)}_{1} + a_2 \underbrace{\cancel{(v_2, v_1)}}_{0}$

$a_1 = (x, v_1)$

$\|p\|^2 = (a_1 v_1 + a_2 v_2, a_1 v_1 + a_2 v_2) = a_1^2 + a_2^2$
$= (x, v_1)^2 + (x, v_2)^2$

objective: find $v_1, \ldots, v_k$ orthonormal

to maximize $\underbrace{\frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{k} (x_i, v_j)^2}$

avg. sum of squared projected lengths.

Application: Visualization.

(1) Perform PCA $\longrightarrow$ $\underline{v_1, \ldots, v_k}$ "principal components" (top k)

(2) $\forall x_i$ define "v_1 coord" (x_i, v_1)

"v_2 coord" (x_i, v_2)

$\vdots$

"v_k coord" (x_i, v_k)

(3) Plot your pts

$x_i = \underline{((x_i, v_1), (x_i, v_2), (x_i, v_3))}$

- look for clusters
- look what are pts particularly large along v_1

genetic data of Europeans $\boxed{\text{SNPs}}$ A, T, C, G

250,000 positions

people

$k = 2$

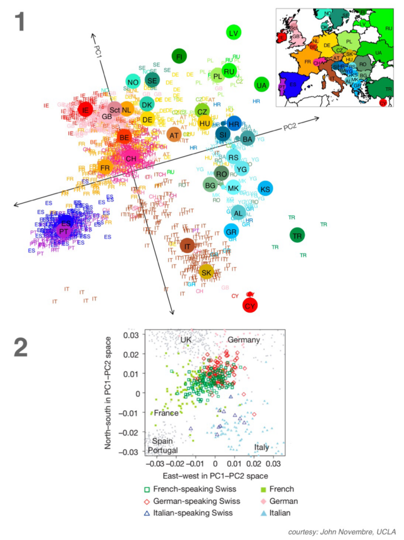
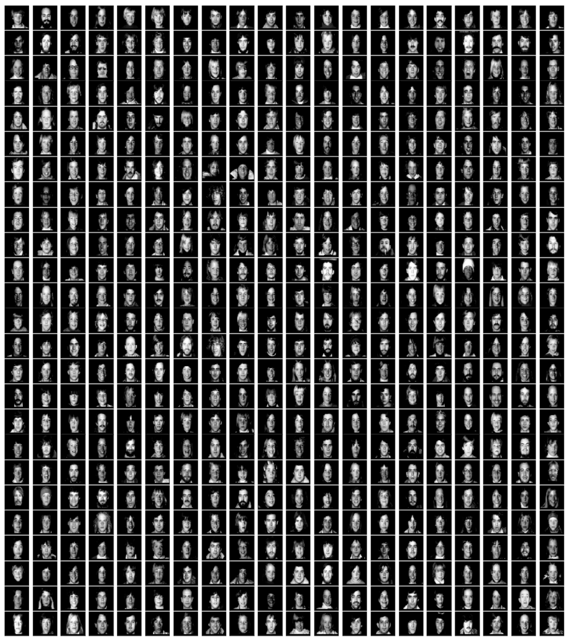


Figure 1: The genetic map of Europe using PCA, with the geographic map of Europe for reference. Figure 2: The same map, but zoomed in on Switzerland. Swiss individuals tend to cluster with countries that speak the same language. *(Courtesy: John Novembre, UCLA)*

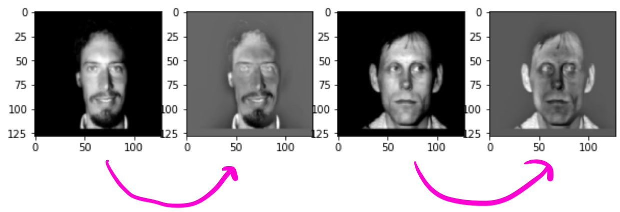
Application 2: Compression

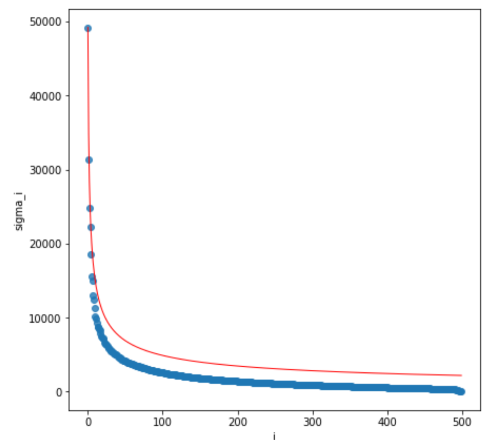
Eigenfaces.

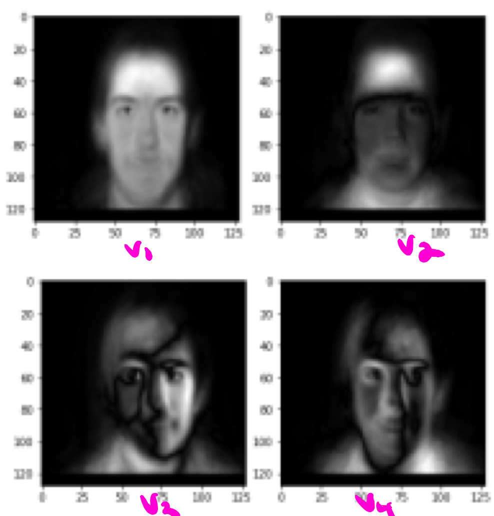
500 faces



2 examples of images after subtracting "mean" face.



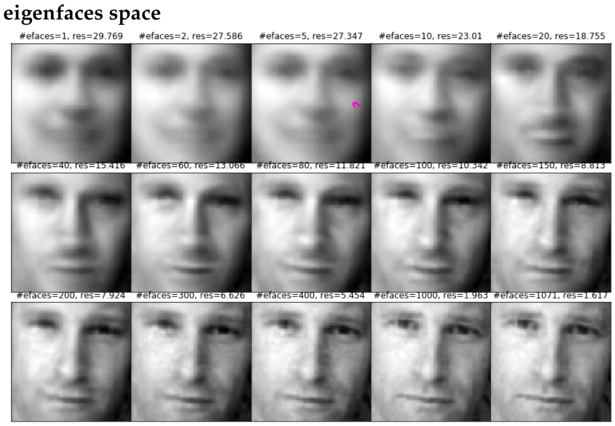


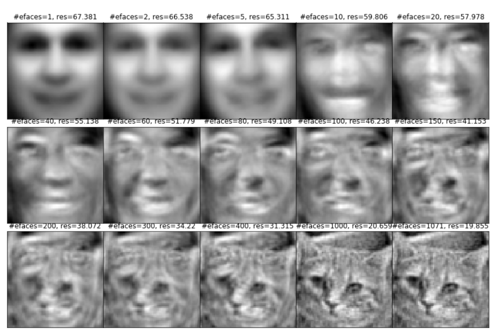


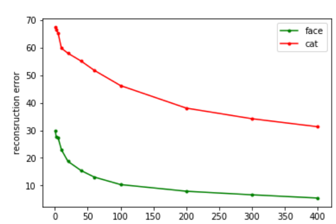
first 4 principal components

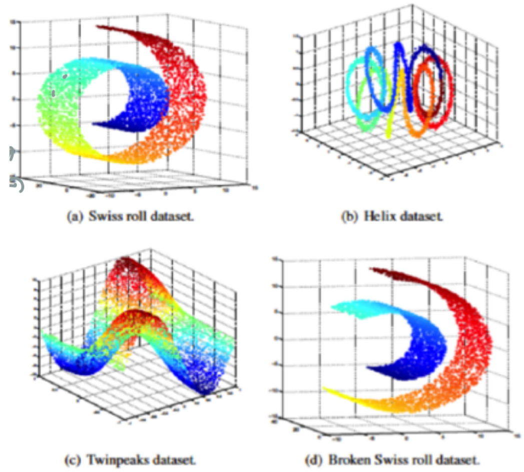
Original Face Image

eigenfaces space









(a) Swiss roll dataset. (b) Helix dataset.

(c) Twinpeaks dataset. (d) Broken Swiss roll dataset.

How PCA works

$\underline{k=1}$ find v s.t. $\|v\|=1$

to maximize $\frac{1}{m}\boxed{\sum_{i=1}^{m}(x_i, v)^2}$

$$X = \begin{pmatrix} \text{---} & x_1 & \text{---} \\ & \vdots & \\ \text{---} & x_m & \text{---} \end{pmatrix} \qquad Xv = \begin{pmatrix} (x_1, v) \\ \vdots \\ (x_m, v) \end{pmatrix}$$

$$(Xv)^T \qquad \boxed{((x_1,v) \;\cdots\; (x_m,v))} \cdot Xv$$

$$\boxed{(Xv)^T}\, Xv$$

$$(Xv)^T = v^T X^T$$

$$\underbrace{v^T X^T X}_{A} \cdot v = \sum_{i=1}^{m}(x_i, v)^2$$

$$((x_1,v) \;\cdots\; (x_m,v)) \begin{pmatrix} (x_1,v) \\ \vdots \\ (x_m,v) \end{pmatrix}$$

find v to

max $v^T A v$

$$\boxed{A = X^T X} \to \text{correlation covariance}$$

$$A_{k\ell} = \sum_{i=1}^{m} x_{ik}\, x_{i\ell}$$

$$k \begin{pmatrix} | & & | \\ x_1 & \cdots & x_m \\ | & & | \end{pmatrix} \overset{\ell^{th}}{\begin{pmatrix} \text{---} & x_1 & \text{---} \\ & \vdots & \\ \text{---} & x_m & \text{---} \end{pmatrix}}$$

A is symmetric.

Suppose rows of X are documents, cols words

$$\max_{\|v\|=1} v^T A v$$

Suppose $A = \begin{pmatrix} 2 & & 0 \\ & 1 & \\ 0 & & \frac{1}{2} \end{pmatrix}$

$v = (v_1, v_2, v_3)$

$v^T A v$

$\rightarrow 2 \cdot v_1^2 + 1 \cdot v_2^2 + \frac{1}{2} \cdot v_3^2$

$$\boxed{v_1^2 + v_2^2 + v_3^2 = 1}$$

Every symmetric

$v = (1, 0, 0)$

$v_1 = 1$

$$\begin{pmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_n \end{pmatrix}$$

$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$

$\sum \lambda_i v_i^2$

$\sum v_i^2 = 1$

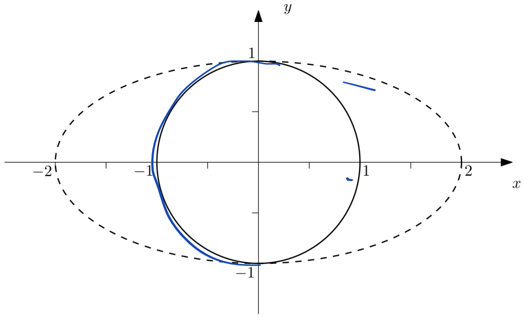


Figure 1: The point (x, y) on the unit circle is mapped to $(2x, y)$.

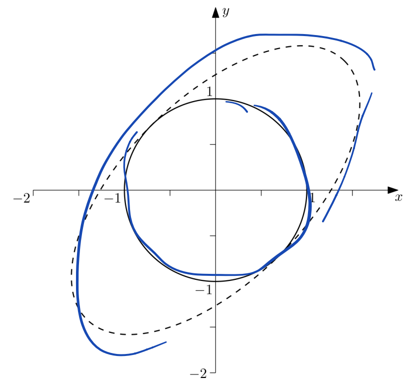


Figure 2: The same scaling as Figure 1, but now rotated 45 degrees.

$$\begin{pmatrix}2 & 0\\ 0 & 1\end{pmatrix}$$

$$\begin{pmatrix}\frac32 & \frac12\\ \frac12 & \frac32\end{pmatrix} = \underbrace{\begin{pmatrix}\frac{1}{\sqrt2} & -\frac{1}{\sqrt2}\\ \frac{1}{\sqrt2} & \frac{1}{\sqrt2}\end{pmatrix}}_{\text{rotate back }45^\circ}\cdot\underbrace{\begin{pmatrix}2 & 0\\ 0 & 1\end{pmatrix}}_{\text{stretch}}\cdot\underbrace{\begin{pmatrix}\frac{1}{\sqrt2} & \frac{1}{\sqrt2}\\ -\frac{1}{\sqrt2} & \frac{1}{\sqrt2}\end{pmatrix}}_{\text{rotate clockwise }45^\circ}\cdot\begin{pmatrix}y_1\\ y_2\end{pmatrix}$$

$=$

$$\boxed{\underset{n\times n}{A} = Q\,D\,Q^T}$$

diagonal

$$Q = \begin{array}{c} q_1 \; q_2 \; \cdots \; q_n \\ \begin{pmatrix} | & | & & | \\ | & | & \cdots & | \\ | & | & & | \end{pmatrix}\end{array} \quad e_1 \qquad Q^T = \begin{pmatrix} — q_1 — \\ — q_2 — \\ \vdots \\ — q_n — \end{pmatrix}$$

orthogonal $\equiv$ cols are orthonormal

$\|q_i\|^2 = 1 \qquad (q_i, q_j) = 0 \quad \forall\, i \neq j$

Special

$$(Qv)^T Qv$$

$$= v^T \underbrace{Q^T Q}_{I} v = v^T v$$

$$QQ^T = I$$

$$Q^TQ = I$$

orthogonal matrices preserve length.

max $\quad v^T A v = v^T \underbrace{Q D \underbrace{Q^T v}}$

$y = Q^T v$

if $\|v\|=1 \Rightarrow \|y\|=1 \Leftarrow \sum y_j^2 = 1$

$$y^T D y = \sum_{j=1}^{n} \lambda_j y_j^2$$

$y_1 = 1 \quad y_j = 0 \quad j > 1$

$y = e_1 \qquad e_1 = Q^T v \qquad \lambda_1 \geq \lambda_2 \geq \cdots$

$$Q e_1 = \underbrace{Q Q^T}_{I} v$$

$$v = Q e_1 = q_1 \text{ (first col of } Q\text{)}$$

$M \cdot z \qquad z$ is eigenvector of M w/ eigenvalue λ

$$Mz = \lambda z$$

Observation: $Q e_1$ is eigenvector of matrix A corresponding to eigenvalue λ_1

$\Longrightarrow Q e_j$ " " λ_j

$$\underline{A Q e_j} = Q D \underbrace{Q^T Q}_{I} e_j = Q D e_j$$

$$= Q \lambda_j e_j$$

$$= \lambda_j \underbrace{Q e_j}_{j^{th}\text{ col of } Q}$$

$v^T A v$

soln for $k=1$: largest eigenvector of $A = X^T X$

$A = Q D Q^T$

to do PCA w/ $k=1$

find unit vector v that maximizes $v^T \underbrace{X^T X}_{A} v$

principal eigenvector of A

$A = Q D Q^T$

set of all eigenvectors

v $\boxed{Qe_1} = q_1$ $\quad \lambda_1$

$\vdots \qquad \vdots$

$Qe_n = q_n$ $\quad \lambda_n$

diag of D.
all nonnegative.

u_0 random vector. eigenvectors $q_1, \dots, q_n$ are a basis

$$u_0 = \sum_{j=1}^{n} c_j q_j$$

$$A u_0 = \sum_j c_j A q_j = \sum_j c_j \lambda_j q_j$$

$$A \sum_j c_j \lambda_j q_j = \sum_j c_j \lambda_j A q_j = \sum_j c_j \lambda_j^2 q_j$$

$$A \sum_j c_j \lambda_j^2 q_j = \sum_j c_j \lambda_j^3 q_j$$

$$A^k u_0 = \sum_j c_j \lambda_j^k q_j$$

$$= \lambda_1^k \left[c_1 q_1 + c_2 \frac{\lambda_2^k}{\lambda_1^k} q_2 + c_3 \frac{\lambda_3^k}{\lambda_1^k} q_3 + \dots \right]$$

$\leq \lambda_3 \leq \lambda_2 < \lambda_1$

$$= \boxed{c_1 \lambda_1^k} q_1$$

rescale $\Rightarrow q_1$

Algorithm 1
POWER ITERATION

Given matrix $\mathbf{A} = \mathbf{X}^\top\mathbf{X}$:

- Select random unit vector $\mathbf{u}_0$
- For $i = 1, 2, \ldots$, set $\mathbf{u}_i = \mathbf{A}^i\mathbf{u}_0$. If $\mathbf{u}_i/||\mathbf{u}_i|| \approx \mathbf{u}_{i-1}/||\mathbf{u}_{i-1}||$, then return $\mathbf{u}_i/||\mathbf{u}_i||$.

vector not changing.

$O\left(\frac{\log n}{\log(\lambda_1/\lambda_2)}\right)$ iterations.

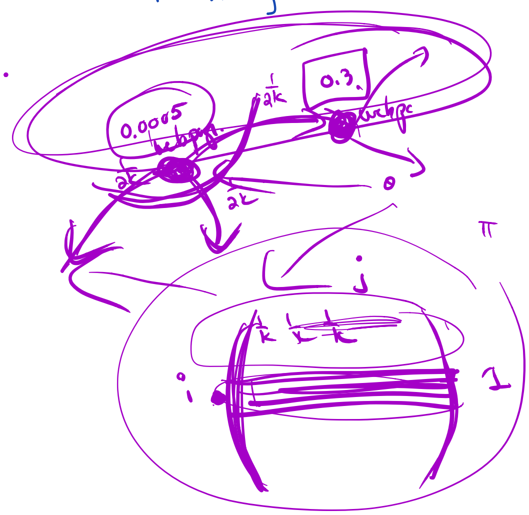
$A^2 u_0$

$A \quad A^2 \quad A^4 \quad A^8$

$u_1 = A u_0$
$u_2 = A u_1$
$u_3 = A u_2$
$\vdots$
$u_i = A^i u_0$

$\lambda_1 > \lambda_2$

Pagerank.



1. Find the top component, $\mathbf{v}_1$, using power iteration.
2. Project the data matrix orthogonally to $\mathbf{v}_1$:

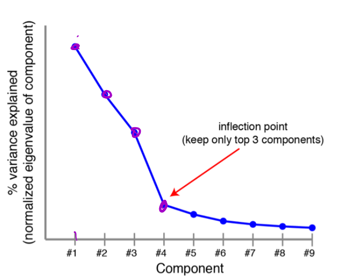
$$\begin{bmatrix} — & \mathbf{x}_1 & — \\ — & \mathbf{x}_2 & — \\ & \vdots & \\ — & \mathbf{x}_m & — \end{bmatrix} \mapsto \begin{bmatrix} — & (\mathbf{x}_1 - \langle \mathbf{x}_1, \mathbf{v}_1 \rangle \mathbf{v}_1) & — \\ — & (\mathbf{x}_2 - \langle \mathbf{x}_2, \mathbf{v}_1 \rangle \mathbf{v}_1) & — \\ & \vdots & \\ — & (\mathbf{x}_m - \langle \mathbf{x}_m, \mathbf{v}_1 \rangle \mathbf{v}_1) & — \end{bmatrix}.$$

This corresponds to subtracting out the variance of the data that is already explained by the first principal component $\mathbf{v}_1$.

3. Recurse by finding the top $k - 1$ principal components of the new data matrix.

Singular value decomposition.

$O(n^2 m)$
$O(m^2 n)$



Scree plot. Principal components are ranked by the amount of variance they capture in the original dataset, a scree plot can provide some sense of how many components are needed.