COMS 4771 Lecture 17

1. Principal component analysis
2. Singular value decomposition

Representation learning

Useful representations of data

Representation learning:

- **Given**: raw feature vectors $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(n)} \in \mathbb{R}^d$.
- **Goal**: learn a "useful" feature transformation $\boldsymbol{\phi} \colon \mathbb{R}^d \to \mathbb{R}^k$.
  (Often $k \ll d$—i.e., *dimensionality reduction*—but not always.)

Useful representations of data

Representation learning:

- **Given**: raw feature vectors $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(n)} \in \mathbb{R}^d$.
- **Goal**: learn a "useful" feature transformation $\boldsymbol{\phi} \colon \mathbb{R}^d \to \mathbb{R}^k$.
  (Often $k \ll d$—i.e., *dimensionality reduction*—but not always.)

  Can then use $\boldsymbol{\phi}$ as a feature map for supervised learning.

Useful representations of data

Representation learning:

- **Given**: raw feature vectors $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(n)} \in \mathbb{R}^d$.
- **Goal**: learn a "useful" feature transformation $\boldsymbol{\phi} \colon \mathbb{R}^d \to \mathbb{R}^k$.
  (Often $k \ll d$—i.e., *dimensionality reduction*—but not always.)

  Can then use $\boldsymbol{\phi}$ as a feature map for supervised learning.

Some previously encontered examples:

- Feature maps corresponding to kernels (+approximations).
  (This is usually *data-oblivious*—feature map doesn't depend on the data.)
- Centering; standardization.

tricky thing: the new base vector(sub-space)'s dimensions are in the same length of original vector(original space)

Representation learning:

- **Given**: raw feature vectors $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(n)} \in \mathbb{R}^d$.
- **Goal**: learn a "useful" feature transformation $\boldsymbol{\phi} \colon \mathbb{R}^d \to \mathbb{R}^k$.
  (Often $k \ll d$—i.e., *dimensionality reduction*—but not always.)

  Can then use $\boldsymbol{\phi}$ as a feature map for supervised learning.

Some previously encontered examples:

- Feature maps corresponding to kernels (+approximations).
  (This is usually *data-oblivious*—feature map doesn't depend on the data.)
- Centering; standardization.

What are other desirable properties of a feature representation?

Principal component analysis

Projections

training example, no labels

- **Input**: $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(n)} \in \mathbb{R}^d$, target dimensionality $k \in \mathbb{N}$.
- **Output**: a k-dimensional subspace, represented by an orthonormal basis $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_k \in \boxed{\mathbb{R}^d.}$
- **Projection**: Formally, projection of $\boldsymbol{x} \in \mathbb{R}^d$ to $\boxed{\operatorname{span}(\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_k)}$ is

still in original basis, sub-space

$$\underbrace{\left(\sum_{i=1}^{k} \boldsymbol{v}_i \boldsymbol{v}_i^\top\right)}_{\boldsymbol{\Pi}} \boldsymbol{x} = \sum_{i=1}^{k} \langle \boldsymbol{v}_i, \boldsymbol{x} \rangle \boldsymbol{v}_i \in \mathbb{R}^d.$$

the product is used to compute move along that direction

But, we can simply represent the projection in terms of its coefficients w.r.t. the orthonormal basis $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_k$:

note the transformation

$$\boldsymbol{\phi}(\boldsymbol{x}) := \begin{bmatrix} \boxed{\langle \boldsymbol{v}_1, \boldsymbol{x} \rangle} \\ \langle \boldsymbol{v}_2, \boldsymbol{x} \rangle \\ \vdots \\ \langle \boldsymbol{v}_k, \boldsymbol{x} \rangle \end{bmatrix} \in \mathbb{R}^k.$$

note: the base vector's dimension does not change

Minimize residual squared error

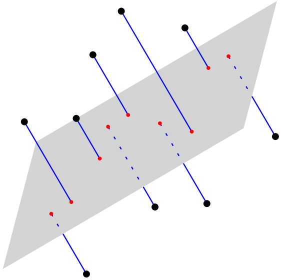
Goal: find k-dimensional $\boxed{\text{projector } \boldsymbol{\Pi} : \mathbb{R}^d \to \mathbb{R}^d}$ such that the total residual squared error

$$\sum_{i=1}^{n} \left\| \boldsymbol{x}^{(i)} - \boldsymbol{\Pi} \boldsymbol{x}^{(i)} \right\|_2^2$$

is as small as possible.

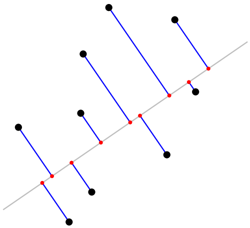
transformation in the same dimensional space

maximum variance in that direction



Projection of minimum residual squared error

$k = 1$ **case** $(\boldsymbol{\Pi} = \boldsymbol{v}\boldsymbol{v}^\top)$

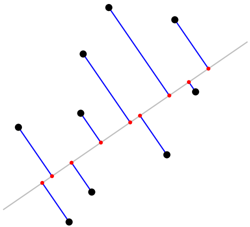


Goal: find unit vector $\boldsymbol{v} \in \mathbb{R}^d$ to minimize

$$\sum_{i=1}^{n} \left\| \boldsymbol{x}^{(i)} - \boldsymbol{v}\boldsymbol{v}^\top \boldsymbol{x}^{(i)} \right\|_2^2$$

PROJECTION OF MINIMUM RESIDUAL SQUARED ERROR

$k = 1$ **case** $(\boldsymbol{\Pi} = \boldsymbol{v}\boldsymbol{v}^\top)$



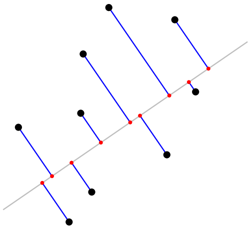
Goal: find unit vector $\boldsymbol{v} \in \mathbb{R}^d$ to minimize

$$\sum_{i=1}^{n} \left\| \boldsymbol{x}^{(i)} - \boldsymbol{v}\boldsymbol{v}^\top \boldsymbol{x}^{(i)} \right\|_2^2$$

$$= \quad \sum_{i=1}^{n} \|\boldsymbol{x}^{(i)}\|_2^2 - \boldsymbol{v}^\top \left(\sum_{i=1}^{n} \boldsymbol{x}^{(i)} \boldsymbol{x}^{(i)\top} \right) \boldsymbol{v}$$

Projection of minimum residual squared error

$k = 1$ **case** $(\boldsymbol{\Pi} = \boldsymbol{v}\boldsymbol{v}^\top)$



Goal: find unit vector $\boldsymbol{v} \in \mathbb{R}^d$ to minimize

$$\sum_{i=1}^{n} \left\| \boldsymbol{x}^{(i)} - \boldsymbol{v}\boldsymbol{v}^\top \boldsymbol{x}^{(i)} \right\|_2^2$$

$$= \sum_{i=1}^{n} \|\boldsymbol{x}^{(i)}\|_2^2 - \boldsymbol{v}^\top \left(\sum_{i=1}^{n} \boldsymbol{x}^{(i)} \boldsymbol{x}^{(i)\top} \right) \boldsymbol{v}$$

$$= \sum_{i=1}^{n} \|\boldsymbol{x}^{(i)}\|_2^2 - \boldsymbol{v}^\top \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}.$$

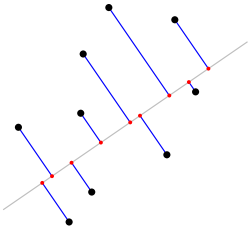
($\boldsymbol{x}^{(i)\top}$ is i-th row of $\boldsymbol{X} \in \mathbb{R}^{n \times d}$.)

PROJECTION OF MINIMUM RESIDUAL SQUARED ERROR

not covex

just a single vector

$k = 1$ **case** $(\boldsymbol{\Pi} = \boldsymbol{v}\boldsymbol{v}^\top)$



Goal: find unit vector $\boldsymbol{v} \in \mathbb{R}^d$ to minimize

$$\sum_{i=1}^{n} \left\| \boldsymbol{x}^{(i)} - \boldsymbol{v}\boldsymbol{v}^\top \boldsymbol{x}^{(i)} \right\|_2^2$$

$$= \sum_{i=1}^{n} \|\boldsymbol{x}^{(i)}\|_2^2 - \boldsymbol{v}^\top \left(\sum_{i=1}^{n} \boldsymbol{x}^{(i)} \boldsymbol{x}^{(i)\top} \right) \boldsymbol{v}$$

$$= \sum_{i=1}^{n} \|\boldsymbol{x}^{(i)}\|_2^2 - \boldsymbol{v}^\top \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}.$$

($\boldsymbol{x}^{(i)\top}$ is i-th row of $\boldsymbol{X} \in \mathbb{R}^{n \times d}$.)

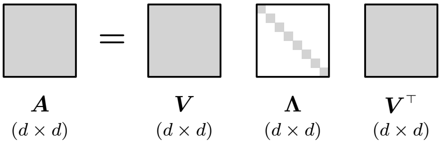
to be convex, the X must be postive definte

$$\operatorname*{arg\,min}_{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\|_2 = 1} \sum_{i=1}^{n} \left\| \boldsymbol{x}^{(i)} - \boldsymbol{v}\boldsymbol{v}^\top \boldsymbol{x}^{(i)} \right\|_2^2 \equiv \boxed{\operatorname*{arg\,max}_{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\|_2 = 1} \boldsymbol{v}^\top \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}.}$$

just see it in the original space. the new coordination is the dot on the line

Eigendecompositions

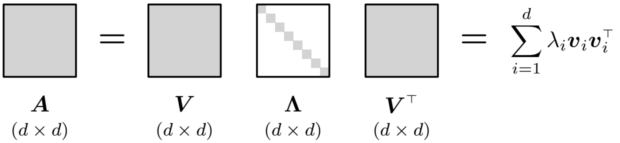
Every symmetric matrix $\boldsymbol{A} \in \mathbb{R}^{d \times d}$ guaranteed to have eigendecomposition with real eigenvalues:



$$= \sum_{i=1}^{d} \lambda_i \boldsymbol{v}_i \boldsymbol{v}_i^\top$$

real **eigenvalues**: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \quad (\boldsymbol{\Lambda} = \mathrm{Diag}(\lambda_1, \lambda_2, \ldots, \lambda_d))$;
corresponding orthonormal **eigenvectors**: $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_d \quad (\boldsymbol{V} = [\boldsymbol{v}_1 | \boldsymbol{v}_2 | \cdots | \boldsymbol{v}_d])$.

Every symmetric matrix $\boldsymbol{A} \in \mathbb{R}^{d \times d}$ guaranteed to have eigendecomposition with real eigenvalues:



real **eigenvalues**: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \quad (\boldsymbol{\Lambda} = \mathrm{Diag}(\lambda_1, \lambda_2, \ldots, \lambda_d))$;
corresponding orthonormal **eigenvectors**: $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_d \quad (\boldsymbol{V} = [\boldsymbol{v}_1 | \boldsymbol{v}_2 | \cdots | \boldsymbol{v}_d])$.

Variational characeterization of eigenvectors:

$$\max_{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\|_2 = 1} \boldsymbol{v}^\top \boldsymbol{A} \boldsymbol{v} \;=\; \lambda_1, \qquad \boldsymbol{v}_1 = \arg\max(\cdots).$$

Every symmetric matrix $\boldsymbol{A} \in \mathbb{R}^{d \times d}$ guaranteed to have eigendecomposition with real eigenvalues:

$$\underset{(d \times d)}{\boldsymbol{A}} = \underset{(d \times d)}{\boldsymbol{V}} \; \underset{(d \times d)}{\boldsymbol{\Lambda}} \; \underset{(d \times d)}{\boldsymbol{V}^\top} = \sum_{i=1}^{d} \lambda_i \boxed{\boldsymbol{v}_i \boldsymbol{v}_i^\top}$$

matrix

real **eigenvalues**: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \quad (\boldsymbol{\Lambda} = \mathrm{Diag}(\lambda_1, \lambda_2, \ldots, \lambda_d))$;
corresponding orthonormal **eigenvectors**: $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_d \quad (\boldsymbol{V} = [\boldsymbol{v}_1 | \boldsymbol{v}_2 | \cdots | \boldsymbol{v}_d])$.

very important,
represent eigenvalue

Variational characteterization of eigenvectors:

just equal to this

Text

$$\max_{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\|_2 = 1} \boldsymbol{v}^\top \boldsymbol{A} \boldsymbol{v} = \lambda_1, \qquad \boldsymbol{v}_1 = \arg\max(\cdots).$$

$$\boxed{\max_{\substack{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\|_2 = 1 \\ \langle \boldsymbol{v}, \boldsymbol{v}_i \rangle = 0 \ \forall i < k}}} \boldsymbol{v}^\top \boldsymbol{A} \boldsymbol{v} = \lambda_k, \qquad \boldsymbol{v}_k = \arg\max(\cdots).$$

just realted
\lambda

the k vector is orthaonomal with all
previous vectors!

Principal component analysis ($k = 1$)

$k = 1$ **case ($\boldsymbol{\Pi} = \boldsymbol{v}\boldsymbol{v}^\top$)**

$$\operatorname*{arg\,min}_{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\|_2 = 1} \sum_{i=1}^{n} \left\| \boldsymbol{x}^{(i)} - \boldsymbol{v}\boldsymbol{v}^\top \boldsymbol{x}^{(i)} \right\|_2^2 \quad \equiv \quad \operatorname*{arg\,max}_{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\|_2 = 1} \boldsymbol{v}^\top \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}.$$

Solution: eigenvector of $\boldsymbol{X}^\top \boldsymbol{X}$ corresponding to largest eigenvalue ("top eigenvector").

$k = 1$ **case** $(\boldsymbol{\Pi} = \boldsymbol{v}\boldsymbol{v}^\top)$

$$\operatorname*{arg\,min}_{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\|_2 = 1} \sum_{i=1}^{n} \left\| \boldsymbol{x}^{(i)} - \boldsymbol{v}\boldsymbol{v}^\top \boldsymbol{x}^{(i)} \right\|_2^2 \quad \equiv \quad \operatorname*{arg\,max}_{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\|_2 = 1} \boldsymbol{v}^\top \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}.$$

inherentence

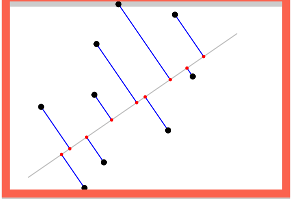
Solution: eigenvector of $\boldsymbol{X}^\top \boldsymbol{X}$ corresponding to largest eigenvalue ("top eigenvector").

variance along the vector

$$\frac{1}{n} \boldsymbol{v}^\top \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v} \; = \; \frac{1}{n} \sum_{i=1}^{n} (\boldsymbol{v}^\top \boldsymbol{x}^{(i)})^2 \; = \; \text{empirical variance of } \boldsymbol{v}^\top \boldsymbol{x}$$

(assuming $\frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}^{(i)} = \mathbf{0}$—i.e., "centering" already applied).

top eigenvector $\equiv$ direction of maximum variance



General k case ($\boldsymbol{\Pi} = \boldsymbol{V}\boldsymbol{V}^\top$)

$$\underset{\boldsymbol{V}\in\mathbb{R}^{d\times k}:\boldsymbol{V}^\top\boldsymbol{V}=\boldsymbol{I}}{\arg\min} \sum_{i=1}^{n} \left\| \boldsymbol{x}^{(i)} - \boldsymbol{V}\boldsymbol{V}^\top \boldsymbol{x}^{(i)} \right\|_2^2 \quad \equiv \quad \underset{\boldsymbol{V}\in\mathbb{R}^{d\times k}:\boldsymbol{V}^\top\boldsymbol{V}=\boldsymbol{I}}{\arg\max} \operatorname{tr}\big(\boldsymbol{V}^\top\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{V}\big).$$

Solution: k eigenvectors of $\boldsymbol{X}^\top\boldsymbol{X}$ corresponding to k largest eigenvalue

Principal component analysis (general k)

General k case ($\boldsymbol{\Pi} = \boldsymbol{V}\boldsymbol{V}^\top$) sum of elements on the main diagonal

$$\operatorname*{arg\,min}_{\boldsymbol{V}\in\mathbb{R}^{d\times k}:\boldsymbol{V}^\top\boldsymbol{V}=\boldsymbol{I}} \sum_{i=1}^{n} \left\| \boldsymbol{x}^{(i)} - \boldsymbol{V}\boldsymbol{V}^\top \boldsymbol{x}^{(i)} \right\|_2^2 \equiv \operatorname*{arg\,max}_{\boxed{\boldsymbol{V}\in\mathbb{R}^{d\times k}:\boldsymbol{V}^\top\boldsymbol{V}=\boldsymbol{I}}} \operatorname{tr}\left(\boldsymbol{V}^\top\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{V}\right).$$

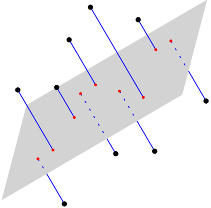
Solution: k eigenvectors of $\boxed{\boldsymbol{X}^\top\boldsymbol{X}\text{ corresponding}}$ to k largest eigenvalue

$$\boxed{\frac{1}{n}}\operatorname{tr}\left(\boldsymbol{V}^\top\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{V}\right) = \sum_{i=1}^{k} \text{empirical variance of } \boldsymbol{v}_i^\top\boldsymbol{x}$$

direction!

(assuming $\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}^{(i)} = \boldsymbol{0}$—i.e., $\boxed{\text{“centering” already applied).}}$

top k eigenvectors $\equiv$ k-dim. subspace of maximum variance



PRINCIPAL COMPONENT ANALYSIS (PCA)

Data matrix $\boldsymbol{X} \in \mathbb{R}^{n \times d}$

Rank k PCA (k dimensional linear subspace)

- Get top k eigenvectors $\boldsymbol{V} = [\boldsymbol{v}_1 | \boldsymbol{v}_2 | \dots | \boldsymbol{v}_k]$ of

$$\boxed{\frac{1}{n}\boldsymbol{X}^\top \boldsymbol{X}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}^{(i)} \boldsymbol{x}^{(i)\top}.$$

- *Feature map*: $\boldsymbol{\phi}(\boldsymbol{x}) := (\langle \boldsymbol{v}_1, \boldsymbol{x} \rangle, \langle \boldsymbol{v}_2, \boldsymbol{x} \rangle, \dots, \langle \boldsymbol{v}_k, \boldsymbol{x} \rangle) \in \mathbb{R}^k$

- *Uncorrelating property*:

$$\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\phi}(\boldsymbol{x}^{(i)}) \boldsymbol{\phi}(\boldsymbol{x}^{(i)})^\top = \mathrm{Diag}(\lambda_1, \lambda_2, \dots, \lambda_k)$$

- *Reconstruction*: $\boldsymbol{x} \mapsto \boldsymbol{V} \boldsymbol{\phi}(\boldsymbol{x})$

transformation.
still in its original dimesnsion

Principal component analysis (PCA)

Data matrix $\boldsymbol{X} \in \mathbb{R}^{n \times d}$

Rank k PCA with centering (k dimensional affine subspace)

- Get top k eigenvectors $\boldsymbol{V} = [\boldsymbol{v}_1 | \boldsymbol{v}_2 | \dots | \boldsymbol{v}_k]$ of

$$\frac{1}{n} \sum_{i=1}^{n} (\boldsymbol{x}^{(i)} - \boldsymbol{\mu})(\boldsymbol{x}^{(i)} - \boldsymbol{\mu})^\top$$

  where $\boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}^{(i)}$.

- *Feature map*: $\boldsymbol{\phi}(\boldsymbol{x}) := (\langle \boldsymbol{v}_1, \boldsymbol{x} - \boldsymbol{\mu} \rangle, \langle \boldsymbol{v}_2, \boldsymbol{x} - \boldsymbol{\mu} \rangle, \dots, \langle \boldsymbol{v}_k, \boldsymbol{x} - \boldsymbol{\mu} \rangle) \in \mathbb{R}^k$

- *Uncorrelating property*:

$$\frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\phi}(\boldsymbol{x}^{(i)}) = \mathbf{0}$$

$$\frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\phi}(\boldsymbol{x}^{(i)}) \boldsymbol{\phi}(\boldsymbol{x}^{(i)})^\top = \mathrm{Diag}(\lambda_1, \lambda_2, \dots, \lambda_k)$$

- *Reconstruction*: $\boldsymbol{x} \mapsto \boldsymbol{\mu} + \boldsymbol{V} \boldsymbol{\phi}(\boldsymbol{x})$

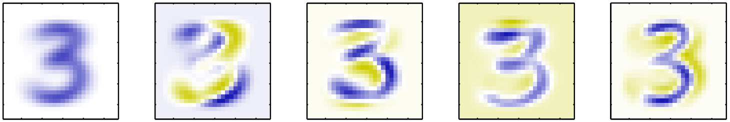
Example: compressing digits images

16×16 pixel images of handwritten 3s $\boxed{\text{as vectors in } \mathbb{R}^{256}}$)

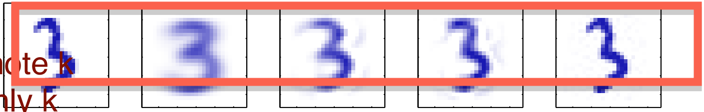
each pixel is a vector

Mean $\boldsymbol{\mu}$ and eigenvectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3, \boldsymbol{v}_4$

Mean | $\lambda_1 = 3.4 \cdot 10^5$ | $\lambda_2 = 2.8 \cdot 10^5$ | $\lambda_3 = 2.4 \cdot 10^5$ | $\lambda_4 = 1.6 \cdot 10^5$



Reconstructions:



$\boldsymbol{x}$ | $k = 1$ | $k = 10$ | $k = 50$ | $k = 200$

magic, note k means. only k number per image.

Only have to store k numbers per image,
along with the mean $\boldsymbol{\mu}$ and k eigenvectors ($256(k+1)$ numbers).

Example: eigenfaces

92×112 pixel images of faces (as vectors in $\mathbb{R}^{10304}$)

100 example images

top $k = 48$ eigenvectors

Computation

Power method

Problem: Given matrix $\boldsymbol{X} \in \mathbb{R}^{n \times d}$, compute the top eigenvector of $\boldsymbol{X}^\top \boldsymbol{X}$.

- **Initialize** with random $\hat{\boldsymbol{v}} \in \mathbb{R}^d$.
- **Repeat**:
  1. $\hat{\boldsymbol{v}} := \boldsymbol{X}^\top \boldsymbol{X} \hat{\boldsymbol{v}}$.
  2. $\hat{\boldsymbol{v}} := \hat{\boldsymbol{v}} / \|\hat{\boldsymbol{v}}\|_2$.

Power method

Problem: Given matrix $\boldsymbol{X} \in \mathbb{R}^{n \times d}$, compute the top eigenvector of $\boldsymbol{X}^\top \boldsymbol{X}$.

- **Initialize** with random $\hat{\boldsymbol{v}} \in \mathbb{R}^d$.
- **Repeat**:
  1. $\hat{\boldsymbol{v}} := \boldsymbol{X}^\top \boldsymbol{X} \hat{\boldsymbol{v}}$.
  2. $\hat{\boldsymbol{v}} := \hat{\boldsymbol{v}} / \|\hat{\boldsymbol{v}}\|_2$.

Theorem: For any $\varepsilon \in (0,1)$, with high probability (over choice of initial $\hat{\boldsymbol{v}}$),

$$\hat{\boldsymbol{v}}^\top \boldsymbol{X}^\top \boldsymbol{X} \hat{\boldsymbol{v}} \;\geq\; (1-\varepsilon) \cdot \text{top eigenvalue of } \boldsymbol{X}^\top \boldsymbol{X}$$

after $O\left(\frac{1}{\varepsilon} \log \frac{d}{\varepsilon}\right)$ iterations.

Power method

Problem: Given matrix $\boldsymbol{X} \in \mathbb{R}^{n \times d}$, compute the top eigenvector of $\boldsymbol{X}^\top \boldsymbol{X}$.

- **Initialize** with random $\hat{\boldsymbol{v}} \in \mathbb{R}^d$.

random is important

- **Repeat**:
  1. $\hat{\boldsymbol{v}} := \boldsymbol{X}^\top \boldsymbol{X} \hat{\boldsymbol{v}}$.
  2. $\hat{\boldsymbol{v}} := \hat{\boldsymbol{v}} / \|\hat{\boldsymbol{v}}\|_2$.

power method for basis vector calculation!

Theorem: For any $\varepsilon \in (0,1)$, with high probability (over choice of initial $\hat{\boldsymbol{v}}$),

$$\hat{\boldsymbol{v}}^\top \boldsymbol{X}^\top \boldsymbol{X} \hat{\boldsymbol{v}} \ \geq \ (1-\varepsilon) \cdot \text{top eigenvalue of } \boldsymbol{X}^\top \boldsymbol{X}$$

after $O\left(\frac{1}{\varepsilon} \log \frac{d}{\varepsilon}\right)$ iterations.

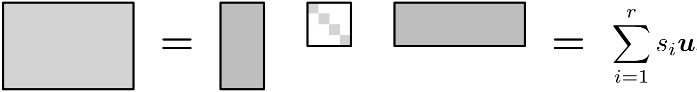
after … iterations, the result eigenvector must meet this condition

Similar algorithm can be used to get top k eigenvectors (for small-ish k).

Singular value decomposition

Singular value decomposition

Every matrix $\boldsymbol{A} \in \mathbb{R}^{n \times d}$ has a **singular value decomposition (SVD)**



$$= \sum_{i=1}^{r} s_i \boldsymbol{u}_i \boldsymbol{v}_i^\top$$

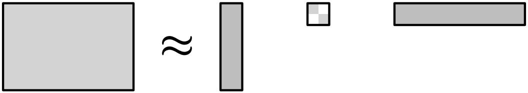
$\boldsymbol{A}$	$\boldsymbol{U}$	$\boldsymbol{S}$	$\boldsymbol{V}^\top$
$(n \times d)$	$(n \times r)$	$(r \times r)$	$(r \times d)$

where

- $r = \operatorname{rank}(\boldsymbol{A}) \quad (r \leq \min\{n, d\})$;
- $\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I} \quad$ (i.e., $\boldsymbol{U} = [\boldsymbol{u}_1 | \boldsymbol{u}_2 | \cdots | \boldsymbol{u}_r]$ has orthonormal columns)
  left singular vectors;
- $\boldsymbol{S} = \operatorname{Diag}(s_1, s_2, \ldots, s_r) \quad$ where $s_1 \geq s_2 \geq \cdots \geq s_r > 0$
  singular values;
- $\boldsymbol{V}^\top \boldsymbol{V} = \boldsymbol{I} \quad$ (i.e., $\boldsymbol{V} = [\boldsymbol{v}_1 | \boldsymbol{v}_2 | \cdots | \boldsymbol{v}_r]$ has orthonormal columns)
  right singular vectors.

Low-rank SVD

For any $k \leq \operatorname{rank}(\boldsymbol{A})$, **rank-$k$ SVD approximation**:

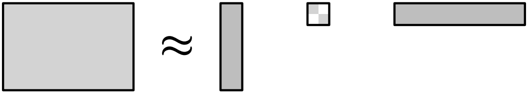


$$\underset{(n \times d)}{\boldsymbol{A}} \approx \underset{(n \times k)}{\widehat{\boldsymbol{U}}_k} \; \underset{(k \times k)}{\widehat{\boldsymbol{S}}_k} \; \underset{(k \times d)}{\widehat{\boldsymbol{V}}_k^\top} = \sum_{i=1}^{k} s_i \boldsymbol{u}_i \boldsymbol{v}_i^\top$$

(Just retain top k left/right singluar vectors and singular values from SVD.)

Low-rank SVD

For any $k \leq \text{rank}(\boldsymbol{A})$, **rank-$k$ SVD approximation**:



$$\approx \qquad = \boxed{\sum_{i=1}^{k} s_i \boldsymbol{u}_i \boldsymbol{v}_i^\top}$$

$\boldsymbol{A}$ $(n \times d)$ $\qquad \widehat{\boldsymbol{U}}_k$ $(n \times k)$ $\qquad \widehat{\boldsymbol{S}}_k$ $(k \times k)$ $\qquad \widehat{\boldsymbol{V}}_k^\top$ $(k \times d)$

(Just retain top k left/right singluar vectors and singular values from SVD.)

Best rank-k approximation:

$$\widehat{\boldsymbol{A}} := \widehat{\boldsymbol{U}}_k \widehat{\boldsymbol{S}}_k \widehat{\boldsymbol{V}}_k^\top = \mathop{\arg\min}_{\substack{\boldsymbol{M} \in \mathbb{R}^{n \times d}: \\ \text{rank}(\boldsymbol{M}) \leq k}} \sum_{i=1}^{n} \sum_{j=1}^{d} (A_{i,j} - M_{i,j})^2 .$$

Minimum value is simply given by

$$\sum_{i=1}^{n} \sum_{j=1}^{d} (A_{i,j} - \widehat{A}_{i,j})^2 = \sum_{t>k} s_t^2 .$$

calculate the residue at each element of the matrix

Example: latent semantic analysis

Represent corpus of documents by counts of words they contain:

	aardvark	abacus	abalone	$\cdots$
document 1	3	0	0	$\cdots$
document 2	7	0	4	$\cdots$
document 3	2	4	0	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	

- One column per vocabulary word in $\boldsymbol{A} \in \mathbb{R}^{n \times d}$
- One row per document in $\boldsymbol{A} \in \mathbb{R}^{n \times d}$
- $A_{i,j} =$ numbers of times word j appears in document i.

Example: latent semantic analysis

Modeling assumption:

- $k \ll \min\{n, d\}$ "topics", each represented by a distributions over vocabulary words:

  $$\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_k \in \mathbb{R}^d$$

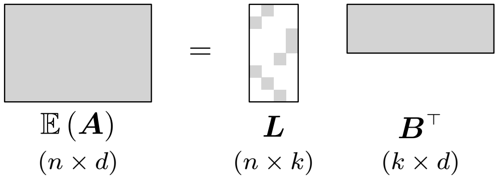
  (Each $\boldsymbol{\beta}_t = (\beta_{t,1}, \beta_{t,2}, \ldots, \beta_{t,d})$ is a probability vector.)

- Each document i is associated with a topic $t_i \in \{1, 2, \ldots, k\}$.

  Document i's count vector (i-th row in $\boldsymbol{A}$) is drawn from a multinomial distribution with probabilities given by $\boldsymbol{\beta}_{t_i}$.

Implication of modeling assumption

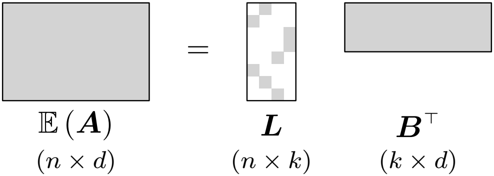
In expectation, $\boldsymbol{A}$ has rank k:



- L_{i,t_i} = length of document i (other entries are zero).
- $\boldsymbol{\beta}_t$ = t-th column of $\boldsymbol{B}$

Implication of modeling assumption

In expectation, $\boldsymbol{A}$ has rank k:



- L_{i,t_i} = length of document i (other entries are zero).
- $\boldsymbol{\beta}_t$ = t-th column of $\boldsymbol{B}$

Observed matrix $\boldsymbol{A}$:

$$\boldsymbol{A} \;=\; \mathbb{E}(\boldsymbol{A}) + \textbf{Zero mean noise}$$

so $\boldsymbol{A}$ is generally of rank $\min\{n, d\} \gg k$.

Example: latent semantic analysis

Using SVD: rank-k SVD $\widehat{\boldsymbol{U}}_k \widehat{\boldsymbol{S}}_k \widehat{\boldsymbol{V}}_k^\top$ of $\boldsymbol{A}$ gives approximation to $\boldsymbol{L}\boldsymbol{B}^\top$:

$$\widehat{\boldsymbol{A}} \;:=\; \widehat{\boldsymbol{U}}_k \widehat{\boldsymbol{S}}_k \widehat{\boldsymbol{V}}_k^\top \;\approx\; \mathbb{E}(\boldsymbol{A}) \;=\; \boldsymbol{L}\boldsymbol{B}^\top.$$

(SVD helps remove some of the effect of the noise.)

Example: latent semantic analysis

Using SVD: rank-k SVD $\widehat{\boldsymbol{U}}_k \widehat{\boldsymbol{S}}_k \widehat{\boldsymbol{V}}_k^\top$ of $\boldsymbol{A}$ gives approximation to $\boldsymbol{L}\boldsymbol{B}^\top$:

$$\widehat{\boldsymbol{A}} := \widehat{\boldsymbol{U}}_k \widehat{\boldsymbol{S}}_k \widehat{\boldsymbol{V}}_k^\top \approx \mathbb{E}(\boldsymbol{A}) = \boldsymbol{L}\boldsymbol{B}^\top.$$

(SVD helps remove some of the effect of the noise.)

- Each of the n documents can be summarized by k numbers:

$$\widehat{\boldsymbol{A}} \widehat{\boldsymbol{V}}_k = \widehat{\boldsymbol{U}}_k \widehat{\boldsymbol{S}}_k \in \mathbb{R}^{n \times k}.$$

Example: latent semantic analysis

Using SVD: rank-k SVD $\widehat{\boldsymbol{U}}_k \widehat{\boldsymbol{S}}_k \widehat{\boldsymbol{V}}_k^\top$ of $\boldsymbol{A}$ gives approximation to $\boldsymbol{L}\boldsymbol{B}^\top$:

$$\widehat{\boldsymbol{A}} := \widehat{\boldsymbol{U}}_k \widehat{\boldsymbol{S}}_k \widehat{\boldsymbol{V}}_k^\top \approx \mathbb{E}(\boldsymbol{A}) = \boldsymbol{L}\boldsymbol{B}^\top.$$

(SVD helps remove some of the effect of the noise.)

- Each of the n documents can be summarized by k numbers:

$$\widehat{\boldsymbol{A}} \widehat{\boldsymbol{V}}_k = \widehat{\boldsymbol{U}}_k \widehat{\boldsymbol{S}}_k \in \mathbb{R}^{n \times k}.$$

- New document representation very useful for information retrieval.

  (Example: cosine similarities between documents become faster to compute and possibly less noisy.)

Example: latent semantic analysis

Using SVD: rank-k SVD $\widehat{\boldsymbol{U}}_k \widehat{\boldsymbol{S}}_k \widehat{\boldsymbol{V}}_k^\top$ of $\boldsymbol{A}$ gives approximation to $\boldsymbol{L}\boldsymbol{B}^\top$:

$$\widehat{\boldsymbol{A}} := \widehat{\boldsymbol{U}}_k \widehat{\boldsymbol{S}}_k \widehat{\boldsymbol{V}}_k^\top \approx \mathbb{E}(\boldsymbol{A}) = \boldsymbol{L}\boldsymbol{B}^\top.$$

(SVD helps remove some of the effect of the noise.)

- Each of the n documents can be summarized by k numbers:

$$\widehat{\boldsymbol{A}} \widehat{\boldsymbol{V}}_k = \widehat{\boldsymbol{U}}_k \widehat{\boldsymbol{S}}_k \in \mathbb{R}^{n \times k}.$$

- New document representation very useful for information retrieval.

  (Example: cosine similarities between documents become faster to compute and possibly less noisy.)

- Actually estimating $\boldsymbol{L}$ and $\boldsymbol{B}$ takes a bit more work.

Recap

- **PCA**: directions of maximum variance in data $\equiv$ subspace that minimizes residual squared error.
- **Computation**: power method
- **SVD**: general decomposition for arbtirary rectangular matrices
  Low-rank SVD: best low-rank approximation of a matrix
- **PCA/SVD**: often useful when low-rank structure is expected (e.g., probabilistic modeling).