Lecture 16 — PCA and SVD.

Alex Schwing and Matus Telgarsky

(Some slide content from Daniel Hsu (Columbia)!)

Announcements.

- Midterm **not yet graded**!
- Homeworks after spring break **pushed back 1 week**!

Schedule for today.

- Overview.
- PCA basics.
- PCA and SVD.
- PCA applications.
- Algorithms.

Reading: Murphy book, parts of chapter 12.

Overview.

Overview.

So far we have focused on **supervised learning**:
constructing a mapping $f : \mathcal{X} \to \mathcal{Y}$ given pairs $((x_i, y_i))_{i=1}^n$.

Overview.

So far we have focused on **supervised learning**:
constructing a mapping $f : \mathcal{X} \to \mathcal{Y}$ given pairs $((x_i, y_i))_{i=1}^{n}$.
Examples:

- k-nn
- Least squares.
- Logistic regression.
- SVM.
- Neural networks.
- Structured prediction.

Unsupervised learning.

Next we will study **unsupervised learning**:
finding structure in **unlabeled** data $(x_i)_{i=1}^n$.

Unsupervised learning.

Next we will study **unsupervised learning**:
finding structure in **unlabeled** data $(x_i)_{i=1}^n$.
Examples:

- PCA.
- k-means.
- Gaussian Mixture Models.
- Hidden Markov Models.
- Generative Adversarian Networks.

Unsupervised learning.

What is the **goal** in unsupervised learning?

- Recover “hidden structure” (e.g., cliques in noisy graphs).
- Data compression / dimension reduction.
- Interpret / explain data and models.
- Features for supervised learning
  (e.g., word embeddings).

Unsupervised learning.

What is the **goal** in unsupervised learning?

- Recover "hidden structure" (e.g., cliques in noisy graphs).
- Data compression / dimension reduction.
- Interpret / explain data and models.
- Features for supervised learning
  (e.g., word embeddings).

The task in unsupervised learning is less clear-cut.

PCA (Principle Component Analysis).

PCA (Principle Component Analysis).

Task (informal):
find best-fitting low-dimensional subspace to $(x_i)_{i=1}^n$.

PCA (Principle Component Analysis).

Task (informal):
find best-fitting low-dimensional subspace to $(x_i)_{i=1}^n$.

Task: Given $(x_i)_{i=1}^n$,
find linear subspace L (with projection operator Π_L)
which minimizes variance:

$$\operatorname*{arg\,min}_{\substack{\text{subspaces } L \subseteq \mathbb{R}^d \\ \dim(L)=k}} \frac{1}{n} \sum_{i=1}^{n} \|x_i - \Pi_L x_i\|^2 .$$

PCA – matrix form (part 1).

Original form:

$$\mathop{\arg\min}_{\substack{\text{subspaces } L \subseteq \mathbb{R}^d \\ \dim(L)=k}} \frac{1}{n} \sum_{i=1}^{n} \left\| x_i - \Pi_L x_i \right\|^2 .$$

PCA – matrix form (part 1).

Original form:

$$\mathop{\arg\min}_{\substack{\text{subspaces } L \subseteq \mathbb{R}^d \\ \dim(L)=k}} \frac{1}{n} \sum_{i=1}^{n} \|x_i - \Pi_L x_i\|^2 .$$

To derive a simpler matrix form:

- Collect $(x_i)_{i=1}^n$ as rows of matrix $X \in \mathbb{R}^{n \times d}$.
- L is k-dimensional $\iff$ has basis $(v_1, \ldots, v_k)$.
  Collect $(v_i)_{i=1}^k$ into $V \in \mathbb{R}^{d \times k}$.
  Note $VV^\top$ denotes orthogonal projection onto columns of V.
- For matrix M, define **Frobenius norm** $\|M\|_{\mathrm{F}}^2 = \sum_{i,j} M_{ij}^2$.

PCA – matrix form (part 1).

Original form:

$$\underset{\substack{\text{subspaces } L\subseteq\mathbb{R}^d \\ \dim(L)=k}}{\arg\min}\ \frac{1}{n}\sum_{i=1}^{n}\|x_i - \Pi_L x_i\|^2 .$$

To derive a simpler matrix form:

- Collect $(x_i)_{i=1}^n$ as rows of matrix $X \in \mathbb{R}^{n\times d}$.
- L is k-dimensional $\iff$ has basis $(v_1, \ldots, v_k)$.
  Collect $(v_i)_{i=1}^k$ into $V \in \mathbb{R}^{d\times k}$.
  Note $VV^\top$ denotes orthogonal projection onto columns of V.
- For matrix M, define **Frobenius norm** $\|M\|_{\mathrm{F}}^2 = \sum_{i,j} M_{ij}^2$.

With this notation, obtain alternate matrix form:

$$\underset{\substack{V\in\mathbb{R}^{d\times k} \\ V^\top V=I}}{\arg\min}\ \frac{1}{n}\left\|X^\top - VV^\top X^\top\right\|_{\mathrm{F}}^2 .$$

PCA – matrix form (part 2).

Given $X \in \mathbb{R}^{n \times d}$ and $V \in \mathbb{R}^{d \times k}$ with $V^\top V = I$,
since $\|M\|_\mathsf{F}^2 = \mathrm{trace}(M^\top M)$,

$$\left\|X^\top - VV^\top X^T\right\|_\mathsf{F}^2 = \left\|X^T\right\|_\mathsf{F}^2 - 2\mathrm{trace}(XVV^\top X^T) + \mathrm{trace}(XVV^\top VV^\top X^T)$$
$$= \|X\|_\mathsf{F}^2 - \mathrm{trace}(V^\top X^\top XV) = \|X\|_\mathsf{F}^2 - \|XV\|_\mathsf{F}^2 .$$

PCA – matrix form (part 2).

Given $X \in \mathbb{R}^{n\times d}$ and $V \in \mathbb{R}^{d\times k}$ with $V^\top V = I$,
since $\|M\|_\mathsf{F}^2 = \mathrm{trace}(M^\top M)$,

$$\begin{aligned}\left\|X^\top - VV^\top X^T\right\|_\mathsf{F}^2 &= \left\|X^T\right\|_\mathsf{F}^2 - 2\mathrm{trace}(XVV^\top X^T) + \mathrm{trace}(XVV^\top VV^\top X^T) \\ &= \|X\|_\mathsf{F}^2 - \mathrm{trace}(V^\top X^\top XV) = \|X\|_\mathsf{F}^2 - \|XV\|_\mathsf{F}^2 .\end{aligned}$$

PCA can thus be rewritten

$$\operatorname*{arg\,min}_{\substack{V\in\mathbb{R}^{d\times k}\\ V^\top V=I}} \frac{1}{n}\left\|X^\top - VV^\top X^\top\right\|_\mathsf{F}^2 = \operatorname*{arg\,max}_{\substack{V\in\mathbb{R}^{d\times k}\\ V^\top V=I}} \|XV\|_\mathsf{F}^2 .$$

Aside: eigendecompositions.

Aside: eigendecompositions.

Recall: given a matrix M, then (Q, Λ) are an **eigendecomposition** when:

- Q is orthonormal ($Q^\top Q = I$).
- Λ is diagonal.
- $M = Q \Lambda Q^\top = \sum_{i=1}^{d} \lambda_i q_i q_i^\top$.

Aside: eigendecompositions.

Recall: given a matrix M, then (Q, Λ) are an **eigendecomposition** when:

- Q is orthonormal $(Q^\top Q = I)$.
- Λ is diagonal.
- $M = Q\Lambda Q^\top = \sum_{i=1}^{d} \lambda_i q_i q_i^\top$.

Moreover:

- $(q_1, \ldots, q_d)$ are **eigenvectors**, $(\lambda_1, \ldots, \lambda_d)$ are **eigenvalues**.
- When M is symmetric, eigendecomposition **exists** and is **real**. Convention: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$.
- Eigendecomposition not in general unique! (E.g., zero matrix. . .)

PCA via eigenvalues.

We've boiled PCA down to

$$\underset{\substack{V \in \mathbb{R}^{d \times k} \\ V^\top V = I}}{\arg\min} \frac{1}{n} \left\| X^\top - VV^\top X^\top \right\|_{\mathsf{F}}^2 = \underset{\substack{V \in \mathbb{R}^{d \times k} \\ V^\top V = I}}{\arg\max} \operatorname{trace}(V^\top X^\top X V).$$

PCA via eigenvalues.

We've boiled PCA down to

$$\mathop{\arg\min}_{\substack{V\in\mathbb{R}^{d\times k}\\ V^\top V=I}} \frac{1}{n}\left\|X^\top - VV^\top X^\top\right\|_{\mathsf{F}}^2 = \mathop{\arg\max}_{\substack{V\in\mathbb{R}^{d\times k}\\ V^\top V=I}} \operatorname{trace}(V^\top X^\top XV).$$

$X^\top X$ is symmetric, with eigendecomposition $X^\top X = Q\Lambda Q^\top$.
We can also rewrite V in the basis Q, thus

$$\begin{aligned}\max_{\substack{V\in\mathbb{R}^{d\times k}\\ V^\top V=I}} \operatorname{trace}(V^\top X^\top XV) &= \max_{\substack{QV\in\mathbb{R}^{d\times k}\\ V^\top V=I}} \operatorname{trace}\left((QV)^\top Q\Lambda Q^\top (QV)\right) \\ &= \max_{\substack{QV\in\mathbb{R}^{d\times k}\\ V^\top V=I}} \operatorname{trace}\left(V^\top \Lambda V\right) = \lambda_1 + \cdots + \lambda_k.\end{aligned}$$

PCA via eigenvalues.

We've boiled PCA down to

$$\operatorname*{arg\,min}_{\substack{V\in\mathbb{R}^{d\times k}\\ V^\top V=I}} \frac{1}{n}\left\|X^\top - VV^\top X^\top\right\|_{\mathsf{F}}^2 = \operatorname*{arg\,max}_{\substack{V\in\mathbb{R}^{d\times k}\\ V^\top V=I}} \operatorname{trace}(V^\top X^\top X V).$$

$X^\top X$ is symmetric, with eigendecomposition $X^\top X = Q\Lambda Q^\top$.
We can also rewrite V in the basis Q, thus

$$\begin{aligned}
\max_{\substack{V\in\mathbb{R}^{d\times k}\\ V^\top V=I}} \operatorname{trace}(V^\top X^\top X V) &= \max_{\substack{QV\in\mathbb{R}^{d\times k}\\ V^\top V=I}} \operatorname{trace}\left((QV)^\top Q\Lambda Q^\top (QV)\right)\\
&= \max_{\substack{QV\in\mathbb{R}^{d\times k}\\ V^\top V=I}} \operatorname{trace}\left(V^\top \Lambda V\right) = \lambda_1 + \cdots + \lambda_k.
\end{aligned}$$

Therefore:

- The solution to PCA is the top k eigenvectors of $X^\top X$.
- The eigenvalues give the maximum *value*.

PCA summary.

We are given data $(x_i)_{i=1}^n$;
We want subspace L, $\dim(L) = k$, minimizing $\sum_{i=1}^n \|x_i - \Pi_L x_i\|^2$.

PCA summary.

We are given data $(x_i)_{i=1}^n$;
We want subspace L, $\dim(L) = k$, minimizing $\sum_{i=1}^n \|x_i - \Pi_L x_i\|^2$.

- Form matrix $X \in \mathbb{R}^{n \times d}$ with x_i as row i.
- Compute top eigenvectors $(v_1, \ldots, v_k)$ of $X^\top X$.
- Collect $(v_1, \ldots, v_l)$ as columns of $V \in \mathbb{R}^{d \times k}$.
- Output V; note $\Pi_L = VV^\top$.

PCA summary.

We are given data $(x_i)_{i=1}^n$;
We want subspace L, $\dim(L) = k$, minimizing $\sum_{i=1}^n \|x_i - \Pi_L x_i\|^2$.

- Form matrix $X \in \mathbb{R}^{n \times d}$ with x_i as row i.
- Compute top eigenvectors $(v_1, \dots, v_k)$ of $X^\top X$.
- Collect $(v_1, \dots, v_l)$ as columns of $V \in \mathbb{R}^{d \times k}$.
- Output V; note $\Pi_L = VV^\top$.

Remark. Often we want **PCA with centering**:
Find the mean $\mu = n^{-1} \sum_{i=1}^n x_i$,
Form $X \in \mathbb{R}^{n \times d}$ where row i has $x_i - \mu$.
Associate x_i with $\mu + \Pi_L(x_i - \mu)$.

PCA and SVD.

PCA and SVD.

Any questions so far?

SVD (Singular Value Decomposition).

Every matrix $M \in \mathbb{R}^{n \times d}$ has an SVD $(U, S, V^\top)$.

- $U \in \mathbb{R}^{n \times r}$ with $U^\top U = I$ and $r := \operatorname{rank}(M)$.
  Columns of U are **left singular vectors** $(u_1, \ldots, u_r)$.
- $S = \operatorname{diag}(s_1, \ldots, s_r)$; these are the **singular values** $s_1 \geq \cdots \geq s_r$.
- $V \in \mathbb{R}^{d \times r}$ with $V^\top V = I$.
  Columns of V are **right singular vectors** $(v_1, \ldots, v_r)$.
- $M = USV^\top = \sum_{i=1}^{r} s_i u_i v_i^\top$.

SVD (Singular Value Decomposition).

Every matrix $M \in \mathbb{R}^{n \times d}$ has an SVD $(U, S, V^\top)$.

- $U \in \mathbb{R}^{n \times r}$ with $U^\top U = I$ and $r := \operatorname{rank}(M)$.
  Columns of U are **left singular vectors** $(u_1, \ldots, u_r)$.
- $S = \operatorname{diag}(s_1, \ldots, s_r)$; these are the **singular values** $s_1 \geq \cdots \geq s_r$.
- $V \in \mathbb{R}^{d \times r}$ with $V^\top V = I$.
  Columns of V are **right singular vectors** $(v_1, \ldots, v_r)$.
- $M = USV^\top = \sum_{i=1}^{r} s_i u_i v_i^\top$.

Remarks.

- Some call this the **thin SVD** or **truncated SVD** (E.g., Murphy book).
- $\sum_i s_i u_i v_i^\top$ is very convenient (consider $r = 0$).
- *Again* not in general unique (consider $s_1 = s_2$).

More on the SVD.

Every matrix $M \in \mathbb{R}^{n \times d}$ has SVD $M = USV^\top$
with $U^\top U = I \in R^{r \times r}$, $V^\top V \in \mathbb{R}^{r \times r}$, $S = \text{diag}(s_1, \ldots, s_r)$.

- $M^\top M$ is symmetric and positive semi-definite
(latter since $x^\top M^\top M x = |Mx|^2 \geq 0$.)
Note $M^T M = VS^2 V^\top$.
- Same with $MM^\top$; also $MM^\top = US^2 U^\top$.
- Eigenvalues of $MM^\top$ and $M^\top M$ coincide;
agree with $(s_1^2, \ldots, s_r^2, 0, \ldots 0)$.
- Eigenvectors of $M^\top M$ are **right singular vectors**;
Eigenvectors of $MM^\top$ are **left singular vectors**.

SVD and PCA.

Given data $(x_i)_{i=1}^n$ collected as rows of $X \in \mathbb{R}^{n \times d}$,
PCA solution was top k eigenvectors of $X^\top X$,
the projected points are $VV^\top X$ where V collects eigenvectors.

SVD and PCA.

Given data $(x_i)_{i=1}^n$ collected as rows of $X \in \mathbb{R}^{n \times d}$,
PCA solution was top k eigenvectors of $X^\top X$,
the projected points are $VV^\top X$ where V collects eigenvectors.

- Eigenvectors of $X^\top X$ are right singular vectors V in $X = USV^\top$.
- PCA solution is V_k (first k columns of V).
- Projected data is $V_k V_k^\top X^\top = V_k V_k^\top VSU^\top = V_k S_k U_k^\top$.
  Reduced dimension description is $S_k U_k^\top$.

PCA summary so far.

- Goal in PCA: find linear subspace L close to data, $\dim(L) = k$.
- Objective function:

$$\operatorname*{arg\,min}_{\substack{\text{subspaces } L \subseteq \mathbb{R}^d \\ \dim(L) = k}} \sum_{i=1}^{n} \|x_i - \Pi_L x_i\|^2 .$$

- Solution 1: top k eigenvectors of $X^\top X$.
- Solution 2: top k right singular vectors of X.

Questions so far?

PCA applications.

(Slides from Daniel Hsu!)

PCA applications.

(Slides from Daniel Hsu!)

Application 1: digit data.

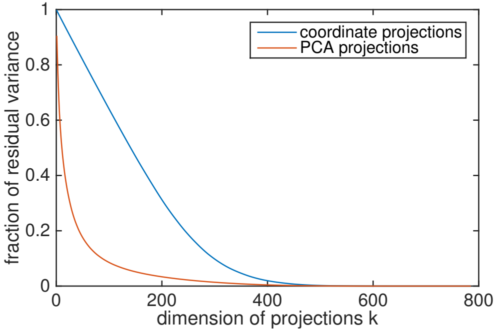
Data $(x_i)_{i=1}^n$ with $x_i \in \mathbb{R}^{784}$.

- Residual variance left by rank-k PCA projection:

$$1 - \frac{\sum_{j=1}^k \text{variance in direction } v_j}{\text{total variance}} .$$

- Residual variance left by best k *coordinate* projections:

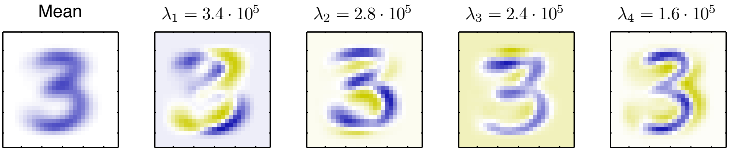
$$1 - \frac{\sum_{j=1}^k \text{variance in direction } \mathbf{e}_j}{\text{total variance}} .$$



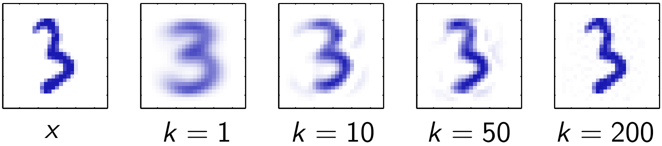
Application 1: digit data.

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

Mean μ and eigenvectors v_1, v_2, v_3, v_4



Reconstructions:



Only have to store k numbers per image,
along with the mean μ and k eigenvectors ($256(k+1)$ numbers).

Application 2: eigenfaces.

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

100 example images

top $k = 48$ eigenvectors

Application 3: topic modeling.

- Let $(x_i)_{i=1}^n$ denote *text documents*:
  each $x_i \in \mathbb{R}^d$ contains normalized word counts
  (d possible words).
- With SVD/PCA, replace x_i with $VV^\top x = Vy$;
  now $y \in \mathbb{R}^k$ (e.g., $k = 100 \ll 30,000 = d$).
- **Problem** (here and before): negative values! (NMF?)
- **Further reading:** look up *LSA (latent semantic analysis)* and *LSI (latent semantic indexing)*.

Algorithms.

Algorithms.

- We reduced PCA to eigenvectors of $X^\top X$.
- An easy solver here is the **power method**.

Algorithms.

- We reduced PCA to eigenvectors of $X^\top X$.
- An easy solver here is the **power method**.
- Basic observation: given $M = Q\Lambda Q^\top$, then

$$M^t = Q\Lambda^t Q^\top = \sum_{i=1}^{d} \lambda_i^t q_i q_i^\top .$$

- I.e., M^t has clearer "eigenvalue structure" then M. How to leverage this algorithmically?

Power method background.

- From $M = Q\Lambda Q^\top$, have $M^t = Q\Lambda^t Q^\top = \sum_i \lambda_i^t q_i q_i^\top$.
- Pick any unit vector x; write it as Qy for unit vector y.
- Therefore $M^t x = \sum_i \lambda_i^t q_i q_i^\top x = \sum_i \lambda_i^t y_i q_i$.
  Seems to "amplify" top eigenvalue!
- Indeed, setting $\Delta := \max_{j \geq 1} \frac{\lambda_j y_j}{\lambda_1 y_1}$,

$$\frac{(q_1^\top M^t x)^2}{\|M^t x\|^2} = \frac{\lambda_1^{2t} y_1^{2t}}{\sum_i \lambda_i^{2t} y_i^{2t}} = \frac{1}{1 + \sum_{i \geq 2} \left(\frac{\lambda_i}{\lambda_1}\right)^{2t} \left(\frac{y_i}{y_1}\right)^{2t}} \geq \frac{1}{1 + k\Delta^{2t}}$$

$$= 1 - \frac{k\Delta^{2t}}{1 + k\Delta^{2t}} \geq 1 - k\Delta^{2t}.$$

- **Thus:** if gap λ_1/λ_2 large and y_1 not too small,
  then ${}^{M^t x}/_{\|M^t x\|} \approx q_1$.

Power method.

Since ${M^t x}/{\|M^t x\|} \approx q_1$, iterate as follows.

- Randomly initialize x_0 with $\|x_0\| = 1$.
- Iterate $x_{t+1} := \frac{Mx_t}{\|Mx_t\|}$.

Power method.

Since ${}^{M^t x}\!/\!_{\|M^t x\|} \approx q_1$, iterate as follows.

- Randomly initialize x_0 with $\|x_0\| = 1$.
- Iterate $x_{t+1} := \frac{Mx_t}{\|Mx_t\|}$.

Remarks.

- Previous slide shows: $\ln(1/\epsilon)$ steps for ϵ-apx solution!
- For left and right singular vectors: replace M with $MM^\top$ and $M^\top M$.

Power method code.

```
norm = numpy.linalg.norm
M = numpy.random.randn(5, 5)
M = M.T @ M
x = numpy.random.randn(5)
x /= norm(x)
xs = []
for i in range(10):
    x = M @ x
    x /= norm(x)
    xs.append(x)
(Lambda, Q) = numpy.linalg.eigh(M)
v = Q[:, -1]
print([ min(norm(v - x), norm(-v - x)) for x in xs ])
```

Output:

```
[0.2879, 0.0825, 0.02375, 0.006839, 0.001969, 0.0005670,
0.0001632, 4.701e-05, 1.353e-05, 3.898e-06]
```

Schedule for today.

- Overview.
- PCA basics.
- PCA and SVD.
- PCA applications.
- Algorithms.

Any questions?