Representation learning:

- **Given**: raw feature vectors $x_1, x_2, \ldots, x_n \in \mathbb{R}^d$.
- **Goal**: learn a “useful” feature transformation $\phi : \mathbb{R}^d \to \mathbb{R}^k$.
  
  (Often $k \ll d$—i.e., *dimensionality reduction*—but not always.)

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

Some previously encountered examples:

- Feature maps corresponding to pos. def. kernels (+approximations).
  
  (Usually *data-oblivious*—feature map doesn’t depend on the data.)

- Centering $x \mapsto x - \mu$
  
  (Effect: resulting features have mean 0.)

- Standardization $x \mapsto \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_d)^{-1}(x - \mu)$.
  
  (Effect: resulting features have mean 0 and unit variance.)

What other properties of a feature representation may be desirable?
**Dimensionality Reduction via Projections**

**Projections**

- **Input:** \(x_1, x_2, \ldots, x_n \in \mathbb{R}^d\), target dimensionality \(k \in \mathbb{N}\).
- **Output:** a \(k\)-dimensional subspace, represented by an orthonormal basis \(q_1, q_2, \ldots, q_k \in \mathbb{R}^d\).
- **Projection:** Formally, projection of \(x \in \mathbb{R}^d\) to \(\text{span}(q_1, q_2, \ldots, q_k)\) is
  \[
  \left(\sum_{i=1}^k q_i q_i^\top\right) x = \sum_{i=1}^k (q_i, x) q_i \in \mathbb{R}^d.
  \]

But, we can simply represent the projection in terms of its coefficients w.r.t. the orthonormal basis \(q_1, q_2, \ldots, q_k\):

\[
\phi(x) := \begin{bmatrix}
\langle q_1, x \rangle \\
\langle q_2, x \rangle \\
\vdots \\
\langle q_k, x \rangle 
\end{bmatrix} \in \mathbb{R}^k.
\]

**Projection of Minimum Residual Squared Error**

**Minimize residual squared error**

**Objective:** find \(k\)-dimensional projector \(\Pi: \mathbb{R}^d \rightarrow \mathbb{R}^d\) such that the average residual squared error

\[
\frac{1}{n} \sum_{i=1}^n \|x_i - \Pi x_i\|_2^2
\]

is as small as possible.

For simplicity, assume \(\frac{1}{n} \sum_{i=1}^n x_i = 0\) (i.e., data already centered).

**Aside: Eigendecompositions**

Every symmetric matrix \(M \in \mathbb{R}^{d \times d}\) guaranteed to have eigendecomposition with real eigenvalues:

\[
M = \begin{bmatrix}
0 & -1 \\
-1 & 0
\end{bmatrix} = \begin{bmatrix}
V & \Lambda \\
V^\top & 0
\end{bmatrix} = \sum_{i=1}^d \lambda_i v_i v_i^\top
\]

real eigenvalues: \(\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d\) \((\Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_d))\); corresponding orthonormal eigenvectors: \(v_1, v_2, \ldots, v_d\) \((V = [v_1 | v_2 | \cdots | v_d])\).

Fixed-point characterization of eigenvectors:

\[
M v_i = \lambda_i v_i.
\]
Eigendecompositions

Variational characterization of eigenvectors:

\[
\max_{q \in \mathbb{R}^d} q^\top M q \\
\text{s.t. } \|q\|_2 = 1
\]

- Maximum value: \(\lambda_1\) (top eigenvalue)
- Maximizer: \(v_1\) (top eigenvector)

For \(i > 1\),

\[
\max_{q \in \mathbb{R}^d} q^\top M q \\
\text{s.t. } \|q\|_2 = 1 \\
\langle q, v_j \rangle = 0 \quad \forall j < i
\]

- Maximum value: \(\lambda_i\) (\(i\)-th largest eigenvalue)
- Maximizer: \(v_i\) (\(i\)-th eigenvector)

Principal component analysis (\(k = 1\))

\[k = 1 \text{ case } (\Pi = qq^\top)\]

\[
\arg \min_{Q \in \mathbb{R}^{d \times k}} \frac{1}{n} \sum_{i=1}^{n} \|x_i - QQ^\top x_i\|_2^2 \\
\equiv \arg \max_{Q \in \mathbb{R}^{d \times k} : Q^\top Q = I} q^\top \left( \frac{1}{n} A^\top A \right) q.
\]

Solution: eigenvector of \(A^\top A\) corresponding to largest eigenvalue (i.e., the top eigenvector \(v_1\)).

\[
q^\top \left( \frac{1}{n} A^\top A \right) q = \frac{1}{n} \sum_{i=1}^{n} (q^\top x_i)^2 = \text{empirical variance in direction } q
\]

\[
\text{top eigenvector } \equiv \text{direction of maximum variance}
\]

Principal component analysis (general \(k\))

\[\text{General } k \text{ case } (\Pi = QQ^\top)\]

\[
\arg \min_{Q \in \mathbb{R}^{d \times k}} \frac{1}{n} \sum_{i=1}^{n} \|x_i - QQ^\top x_i\|_2^2 \\
\equiv \arg \max_{Q \in \mathbb{R}^{d \times k} : Q^\top Q = I} \text{tr} \left( Q^\top \left( \frac{1}{n} A^\top A \right) Q \right).
\]

Solution: \(k\) eigenvectors of \(A^\top A\) corresponding to \(k\) largest eigenvalue

\[
\text{tr} \left( Q^\top \left( \frac{1}{n} A^\top A \right) Q \right) = \sum_{i=1}^{k} \text{empirical variance in direction } q_i
\]

\[
\text{top } k \text{ eigenvectors } \equiv \text{k-dim. subspace of maximum variance}
\]

PCA on OCR digits data

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

- Fraction of residual variance left by rank-\(k\) PCA projection:
  \[
  1 - \frac{\sum_{j=1}^{k} \text{variance in direction } v_j}{\text{total variance}}
  \]

- Fraction of residual variance left by best \(k\) coordinate projections:
  \[
  1 - \frac{\sum_{j=1}^{k} \text{variance in direction } e_j}{\text{total variance}}
  \]

![Graph of PCA projections vs dimension of projections]
It would appear to be a significant computational challenge to find the eigen-decomposition of this matrix. One might be wondering therefore how it is possible to perform PCA on high dimensional data. For example, if $A \in \mathbb{R}^{n \times d}$ is a data matrix,

- The computational complexity of computing the eigen-decomposition of a $k \times k$ matrix is

$$\frac{1}{n} A^\top A = \frac{1}{n} \sum_{i=1}^{n} x_i x_i^\top.$$

- Feature map: $\phi(x) := (\langle x_1, x \rangle, \langle x_2, x \rangle, \ldots, \langle x_k, x \rangle) \in \mathbb{R}^k$.

- Decorrelating property:

$$\frac{1}{n} \sum_{i=1}^{n} \phi(x_i) \phi(x_i)^\top = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_k).$$

- Approx. reconstruction: $x \mapsto \hat{V}_k \phi(x)$.

**Example: compressing digits images**

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

Mean $\mu$ and eigenvectors $v_1, v_2, v_3, v_4$

<table>
<thead>
<tr>
<th>Mean</th>
<th>$\lambda_1 = 3.4 \times 10^6$</th>
<th>$\lambda_2 = 2.8 \times 10^6$</th>
<th>$\lambda_3 = 2.4 \times 10^6$</th>
<th>$\lambda_4 = 1.6 \times 10^5$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image" alt="3" /></td>
<td><img src="image" alt="3" /></td>
<td><img src="image" alt="3" /></td>
<td><img src="image" alt="3" /></td>
</tr>
</tbody>
</table>

Reconstructions:

- $x$  
- $k = 1$  
- $k = 10$  
- $k = 50$  
- $k = 200$

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

**Example: eigenfaces**

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

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

- **Rank $k$ PCA** ($k$ dimensional linear subspace)
  - Get top $k$ eigenvectors $\hat{V}_k := [v_1 | v_2 | \ldots | v_k]$ of
    $$\frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)(x_i - \mu)^\top.$$

- Feature map: $\phi(x) := (\langle v_1, x - \mu \rangle, \langle v_2, x - \mu \rangle, \ldots, \langle v_k, x - \mu \rangle) \in \mathbb{R}^k$.

- Decorrelating property:

$$\frac{1}{n} \sum_{i=1}^{n} \phi(x_i) = 0$$

$$\frac{1}{n} \sum_{i=1}^{n} \phi(x_i) \phi(x_i)^\top = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_k).$$

- Approx. reconstruction: $x \mapsto \mu + \hat{V}_k \phi(x)$.
Other examples

- \( x \in \mathbb{R}^d \): movement of stock prices for \( d \) different stocks in one day.

**Principal component**: combination of stocks that account for the most variation in stock price movement.

- \( x \in \{1, 2, \ldots, 5\}^d \): levels at which various terms describe an individual (e.g., “jolly”, “impulsive”, “outgoing”, “conceited”, “meddlesome”)

**Principal components**: major personality axes in a population (e.g., “extroversion”, “agreeableness”, “conscientiousness”)

Singular value decomposition

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

\[
A = U S V^\top = \sum_{i=1}^r s_i u_i v_i^\top
\]

where

- \( r = \text{rank}(A) \) \((r \leq \min\{n, d\})\);
- \( U^\top U = I \) (i.e., \( U = [u_1 | u_2 | \cdots | u_r] \) has orthonormal columns)
- \( S = \text{diag}(s_1, s_2, \ldots, s_r) \) where \( s_1 \geq s_2 \geq \cdots \geq s_r > 0 \)
- \( V^\top V = I \) (i.e., \( V = [v_1 | v_2 | \cdots | v_r] \) has orthonormal columns)

Low-rank SVD

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

\[
\hat{A} = \hat{U}_k \hat{S}_k \hat{V}_k^\top \approx A \sum_{i=1}^k s_i u_i v_i^\top
\]

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

**Best rank-\( k \) approximation**:

\[
\hat{A} := \hat{U}_k \hat{S}_k \hat{V}_k^\top = \arg \min_{M \in \mathbb{R}^{n \times d}} \sum_{i=1}^n \sum_{j=1}^d (A_{i,j} - M_{i,j})^2.
\]

Minimum value is simply given by

\[
\sum_{i=1}^n \sum_{j=1}^d (A_{i,j} - \hat{A}_{i,j})^2 = \sum_{i>k} s_i^2.
\]
Represent corpus of documents by counts of words they contain:

<table>
<thead>
<tr>
<th></th>
<th>aardvark</th>
<th>abacus</th>
<th>abalone</th>
</tr>
</thead>
<tbody>
<tr>
<td>document 1</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>document 2</td>
<td>7</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>document 3</td>
<td>2</td>
<td>4</td>
<td>0</td>
</tr>
</tbody>
</table>

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

Modeling assumption:

- $k \ll \min\{n, d\}$ “topics”, each represented by a distributions over vocabulary words:
  $$\beta_1, \beta_2, \ldots, \beta_k \in \mathbb{R}^d$$
  (Each $\beta_t = (\beta_{t,1}, \beta_{t,2}, \ldots, \beta_{t,d})$ 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 $A$) is drawn from a multinomial distribution with probabilities given by $\beta_{t_i}$.

Implication of modeling assumption

In expectation, $A$ has rank $k$:

$$\mathbb{E}(A) = \begin{bmatrix} L & B^T \end{bmatrix}$$

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

Using SVD: rank-$k$ SVD $\hat{U}_k \hat{S}_k \hat{V}_k^T$ of $A$ gives approximation to $LB^T$:

$$\hat{A} := \hat{U}_k \hat{S}_k \hat{V}_k^T \approx \mathbb{E}(A) = LB^T.$$ 
(SVD helps remove some of the effect of the noise.)

- Each of the $n$ documents can be summarized by $k$ numbers:
  $$\hat{A}\hat{V}_k = \hat{U}_k \hat{S}_k \in \mathbb{R}^{n \times k}.$$

- New document representation very useful for information retrieval.
  (Example: cosine similarities between documents become faster to compute and possibly less noisy.)
- Actually estimating $L$ and $B$ takes a bit more work.
Recap

- **PCA**: directions of maximum variance in data ≡ subspace that minimizes residual squared error.
- **SVD**: general decomposition for arbitrary 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).