Machine Learning in High Dimension
IA317
Dimension Reduction

Thomas Bonald

2022 – 2023

High dimension

Data $=$ n samples, each with d features

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

High dimension $= d >> 1$ (possibly larger than n)
Typically a **sparse** matrix

Examples

- Textual data (bags of words)
- Medical data
- Marketing data

Dimension reduction

Data $=$ n samples, each with d features

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

Dimension reduction

$$X = \left[\begin{array}{c} \hspace{3cm} \\ \\ \\ \\ \end{array} \right] \quad \to \quad Y = \left[\begin{array}{c} \hspace{0.8cm} \\ \\ \\ \\ \end{array} \right]$$

Main objectives:

- reduce **complexity** (e.g., for nearest neighbors)
- **clustering**
- **visualization**

Feature selection

Select the k most important features for prediction, like

- **correlation** with the labels
- statistical **independence**
- **mutual information**
- **feature importance** → lecture on **ensemble methods**

→ **supervised** learning

Feature selection

$$X = \begin{bmatrix} \qquad\qquad\qquad \end{bmatrix} \quad \rightarrow \quad Y = \begin{bmatrix} \quad \end{bmatrix}$$

Random projection

Data $=$ n samples, each with d features

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

Choose k **random vectors** in this vector space of high dimension:

$$Z \in \mathbb{R}^{k \times d}$$

Random projection

$$X = \left[\qquad\qquad\qquad \right] \quad \rightarrow \quad Y = XZ^T = \left[\quad \right]$$

cf. Locally Sensitive Hashing

Note: The projection vectors can be made **orthogonal** (QR decomposition)

Matrix factorization

Data $=$ n samples, each with d features

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

Matrix factorization

$$X = \left[\phantom{\rule{6em}{6em}} \right] \approx \left[\phantom{\rule{1.5em}{6em}} \right] \left[\phantom{\rule{6em}{2em}} \right] \quad \rightarrow \quad Y = \left[\phantom{\rule{1.5em}{6em}} \right]$$

Inductive bias

Train set $=$ n samples, each with d features

$$X_{\text{train}} \in \mathbb{R}^{n \times d}$$

Matrix factorization

$$X_{\text{train}} \approx \begin{bmatrix} \quad \\ \\ \\ \\ \end{bmatrix} \begin{bmatrix} \qquad\qquad\qquad\qquad \end{bmatrix} \quad \rightarrow \quad Y_{\text{train}} = \begin{bmatrix} \quad \\ \\ \\ \\ \end{bmatrix}$$

How to reduce the dimension of the **test set** X_{test} so that distances between Y_{train} and Y_{test} make sense?

Outline

Focus on 2 **matrix factorization** techniques:

1. Singular Value Decomposition (SVD)
   ↔ Principal Component Analysis (PCA)
2. Non-negative Matrix Factorization (NMF)

Singular values

Let $X \in \mathbb{R}^{n \times d}$

Definition

We say that $\sigma \geq 0$ is a **singular value** of X if there exist unit vectors $u \in \mathbb{R}^n$ and $v \in \mathbb{R}^d$ such that

$$Xv = \sigma u$$
$$X^T u = \sigma v$$

The vectors u and v are left and right **singular vectors** for σ

Property

The vectors u and v are respective **eigenvectors** of XX^T and $X^T X$ for the **eigenvalue** σ^2

Singular value decomposition

Let $X \in \mathbb{R}^{n \times d}$ of rank r

Theorem

There exist $U \in \mathbb{R}^{n \times r}$, $V \in \mathbb{R}^{d \times r}$ and $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_r)$ such that

$$X = \left[\qquad\qquad\qquad \right] = \left[\quad \right] [\quad] [\qquad\qquad\qquad] = U \Sigma V^T$$

with

$$U^T U = I_r \quad V^T V = I_r \quad \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0$$

The matrices U and V are orthogonal bases of left and right **singular vectors** for the singular values $\sigma_1, \ldots, \sigma_r$.

Proof: Spectral theorem applied to either XX^T or $X^T X$.

Matrix factorization by SVD

Data $=$ n samples, each with d features

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

Matrix factorization

$$X = \begin{bmatrix} \quad & \quad \\ \\ \\ \\ \end{bmatrix} \begin{bmatrix} \qquad\qquad\qquad\qquad \end{bmatrix} = U \Sigma V^T \quad \rightarrow \quad Y = U\Sigma = \begin{bmatrix} \quad & \quad \\ \\ \\ \\ \end{bmatrix}$$

Remark

Projection on the **right singular vectors** (orthonormal basis)

$$Y = XV$$

Induction

Train set $=$ n samples, each with d features

$$X_{\text{train}} \in \mathbb{R}^{n \times d}$$

Step 1: Matrix factorization

$$X_{\text{train}} = \begin{bmatrix} \quad \\ \\ \\ \end{bmatrix} \begin{bmatrix} \qquad\qquad\qquad \end{bmatrix} = U \Sigma V^T$$

Step 2: Dimension reduction

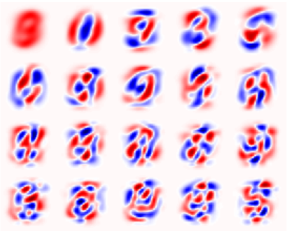
Projection on the **right singular vectors** (of the **train set**)

$$Y_{\text{train}} = X_{\text{train}} V$$
$$Y_{\text{test}} = X_{\text{test}} V$$

Example: MNIST

$X \in \{0, \ldots, 255\}^{n \times d}$
$n = 10,000$ samples
$d = 28 \times 28 = 784$

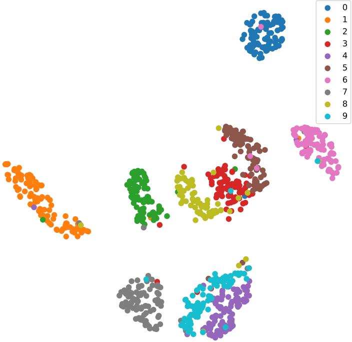
Samples



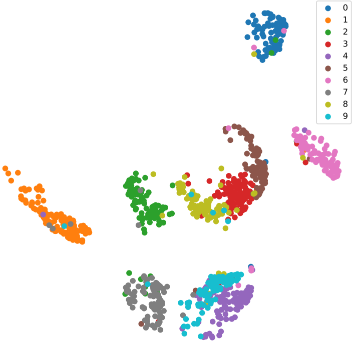
Singular vectors

Example: MNIST

Projection on the first 20 **right singular vectors**
Visualization of 1,000 samples



Train set



Test set

Interpretation of SVD

Let $X \in \mathbb{R}^{n \times d}$

Low-rank approximation

We say that $\hat{X}$ is the **best rank-k approximation** of X if

$$\hat{X} = \arg \min_{M:\mathrm{rank}(M)=k} ||X - M||^2$$

with $|| \cdot ||$ the Frobenius norm (= Euclidean norm for matrices)

Property

For any $k \leq \mathrm{rank}(X)$, the best rank-k approximation of X is

$$\hat{X} = U_{\to k} \Sigma_{\to k} {V_{\to k}}^T$$

with $U_{\to k}, V_{\to k}, \Sigma_{\to k}$ the **restriction** to the first k singular vectors

Approximation error

Let $X \in \mathbb{R}^{n \times d}$

Property

For any $k \leq \text{rank}(X)$, the minimum **error** of a rank-k approximation of X is

$$||X - \hat{X}||^2 = ||X||^2 - \sum_{l \leq k} \sigma_l^2$$

Note: The **relative** error is:

$$\rho = \frac{||X - \hat{X}||^2}{||X||^2} = 1 - \frac{\sum_{l \leq k} \sigma_l^2}{||X||^2}$$

Interpretation of singular vectors

Let $X \in \mathbb{R}^{n \times d}$

Property

The leading singular vector is the direction of **largest inertia**:

$$v_1 = \arg \max_{v:||v||=1} ||Xv||^2$$

Note: If X is centered, in the sense that

$$1^T X = \begin{bmatrix} 1 & \dots & 1 \end{bmatrix} \begin{bmatrix} \quad & & \quad \\ & & \\ & & \end{bmatrix} = 0$$

this is also the direction of **highest variance**

Interpretation of singular vectors

Let $X \in \mathbb{R}^{n \times d}$

Property 1

The k-th singular vector is the direction of highest inertia, **orthogonal** to the previous ones:

$$v_k = \arg \max_{v:||v||=1, {v_1}^T v = \ldots = {v_{k-1}}^T v = 0} ||Xv||^2$$

Property 2

The k-th singular vector is the direction of highest inertia of the **residual** $X - \hat{X}$ with

$$\hat{X} = X \sum_{l<k} v_l {v_l}^T = U_{\rightarrow k-1} \Sigma_{\rightarrow k-1} {V_{\rightarrow k-1}}^T$$

Note: If X is centered, the inertia is the **variance**

Principal Component Analysis

PCA = SVD **after** centering

$$X \quad \rightarrow \quad X - \frac{11^T}{n} X$$

The directions (= principal components) can be interpreted as the directions of **highest variance**

Warning

If X is a **sparse** matrix, its centered version is no longer sparse!

Outline

Focus on 2 matrix factorization techniques:

1. Singular Value Decomposition (SVD)
   $\leftrightarrow$ Principal Component Analysis (PCA)
2. **Non-negative Matrix Factorization** (NMF)

Non-negative matrix factorization

Data $=$ n samples, each with d **non-negative** features

$$X \in \mathbb{R}_+^{n \times d}$$

Non-negative matrix factorization

$$X \approx WH = \left[\begin{matrix} \quad \\ \\ \\ \\ \end{matrix}\right] \left[\begin{matrix} \qquad\qquad\qquad\qquad \end{matrix}\right] \quad \to \quad Y = W = \left[\begin{matrix} \quad \\ \\ \\ \\ \end{matrix}\right]$$

with $W, H \geq 0$

Remark

The dimension reduction is **not** a projection!

Non-negative matrix factorization

Let $X \in \mathbb{R}_+^{n \times d}$ be a non-negative matrix
The optimization problem

$$\min_{W,H \geq 0} \|X - WH\|^2$$

is **convex** in W and H but not in both

Lee-Seung's algorithm (2000)

Alternate updates

$$H \leftarrow H \times \frac{W^T X}{W^T W H} \quad W \leftarrow W \times \frac{X H^T}{W H H^T}$$

with **component-wise** matrix multiplications and divisions

Theorem

The approximation error $\|X - \hat{X}\|$ with $\hat{X} = WH$ is **non-increasing**

Induction

Train set $=$ n samples, each with d non-negative features

$$X_{\text{train}} \in \mathbb{R}_{+}^{n \times d}$$

Step 1: Non-negative matrix factorization

$$X_{\text{train}} = \begin{bmatrix} \quad \\ \\ \\ \end{bmatrix} \begin{bmatrix} \qquad\qquad\qquad\qquad \end{bmatrix} \approx WH \quad \rightarrow \quad Y_{\text{train}} = W = \begin{bmatrix} \quad \\ \\ \\ \end{bmatrix}$$

Step 2: Prediction

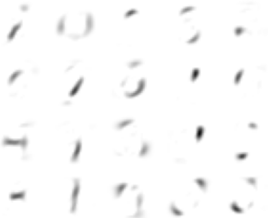
For the **test set**, apply Lee-Seung's algorithm with H fixed:

$$X_{\text{test}} = \begin{bmatrix} \quad \\ \\ \\ \end{bmatrix} \begin{bmatrix} \qquad\qquad\qquad\qquad \end{bmatrix} \approx W'H \quad \rightarrow \quad Y_{\text{test}} = W' = \begin{bmatrix} \quad \\ \\ \\ \end{bmatrix}$$

Example: MNIST

$X \in \{0, \ldots, 255\}^{n \times d}$
$n = 10,000$ samples
$d = 28 \times 28 = 784$

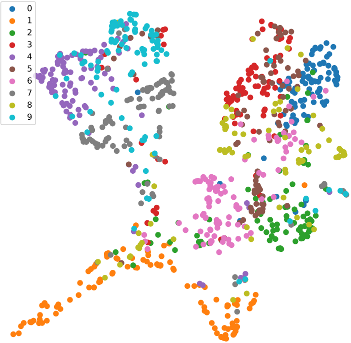
Samples



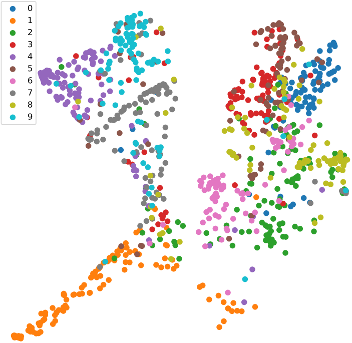
Dictionary (dimension 20)

Example: MNIST

NMF in dimension 20
Visualization of 1,000 samples



Train set



Test set

Summary

Dimension reduction

- 2 **matrix factorization** techniques, SVD and NMF
  Applicable to **sparse** matrices
- Critical choice of the **dimension**
- Different **interpretations**
- Other techniques: auto-encoders, GSVD, ...