

Contents

Class 4	1
How are points distributed in high dimensional space?	1
PCA	1
What do I want to preserve/throw-away when I reduce the dimensions?	2

Class 4

We discuss how to transform raw data into feature vectors today.

100 x 100 image => 10000 feature space (huge dimensional space)

From a learning perspective it is better to have a low dimensional feature space

 Simpler models tend to be better in efficiency and also in their correctness

How are points distributed in high dimensional space?

Directions with more spread tend to contain more info.

 We try to construct the mapping that contains the most information (while dimension reduction)

We'll use covariance to find this mapping.

$$\Sigma = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)(x_i - \mu)^T$$

COVARIANCE MATRIX

Answer: Covariance Matrix & SVD (Singular value decomposition).

We are essentially doing PCA (principal component analysis).

PCA

- compute μ for each feature
- subtract and find data matrix A ($A = X - \mu$).
- $A^T A / N$ and compute covariance matrix
- get eigenvalues for the covariance matrix S or Σ .

Get eigenvalues in (descending order... largest in 1st row and so on). This eigenvalue matrix is denoted by u .

$$z = ux$$

$$\hat{x} = u^T z$$

We can decide how many eigenvalues to keep (for dimension reduction).

These eigenvalues will give us the eigenvectors (that a)

Ratio

$$\frac{\sum_{i=1}^k \lambda_i}{\sum_{i=1}^n \lambda_i}$$

k how many features we want.

PCA does not care about the labels.

PCA is unsupervised algorithm. (Projecting from high dimension to low dimension space).

$A^T A$ has 10000 x 10000 elements.

Instead we find eigenvalues for AA^T (same eigenvalues but less dimensional data 300 x 300).

$$\begin{aligned} A^T A e &= \lambda e \\ \implies AA^T(Ae) &= \lambda(Ae) \end{aligned}$$

What do I want to preserve/throw-away when I reduce the dimensions?

Discussed