San José State University

Math 253: Mathematical Methods for Data Visualization

Principal Component Analysis (PCA)

– A First Dimensionality Reduction Approach

Dr. Guangliang Chen

Introduction

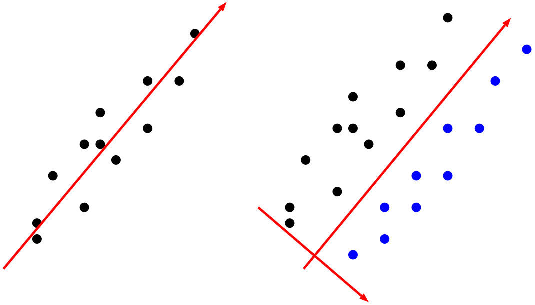
- Many data sets have very high dimensions nowadays, causing significant challenge in storing and processing them.
- We need a way to reduce the dimensionality of the data in order to reduce memory requirement while increasing speed.
- If we discard some dimensions, will that degrade the performance?
- The answer can be no, as long as we do it carefully by **preserving only the information that is needed by the task**. In fact, it may even lead to better results in many cases.

Different dimentionality reduction algorithms preserve different kinds of information (when reducing the dimension):

- **Principal Component Analysis (PCA)**: variance
- **Multidimensional Scaling (MDS)**: distance
- **ISOMap**: geodesic distance
- **Local Linear Embedding (LLE)**: local geometry
- **Laplacian Eigenmaps**: local affinity
- **Linear Discriminant Analysis (LDA)**: separation among classes

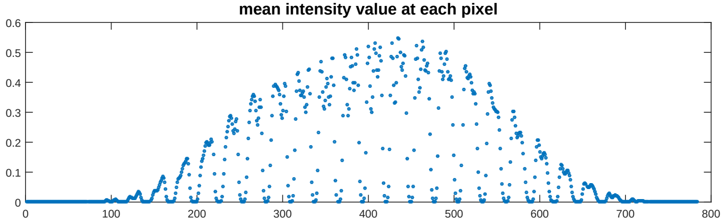
A demonstration

"Useful" information of a data set is often contained in only a small number of dimensions.



Another example

Average intensity value of each pixel of the MNIST handwritten digits:



- Boundary pixels tend to be zero;
- Number of degrees of freedom of each digit is much less than 784.

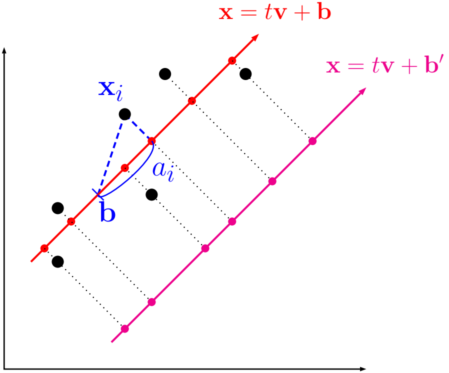
The one-dimensional PCA problem

Problem. Given a set of data points $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$, find a line $\mathcal{S}$ parametrized by $\mathbf{x}(t) = t \cdot \mathbf{v} + \mathbf{b}$ (with $\|\mathbf{v}\| = 1$) such that the orthogonal projections of the data onto the line

$$P_{\mathcal{S}}(\mathbf{x}_i) = \mathbf{v}\, \underbrace{\mathbf{v}^T(\mathbf{x}_i - \mathbf{b})}_{:=a_i} + \mathbf{b}$$

$$= a_i \mathbf{v} + \mathbf{b}, \quad 1 \le i \le n$$

have the largest possible variance.



Mathematical formulation

First observe that for parallel lines, the projections are different, but the amounts of variance are the same! $\longleftarrow$ This implies that the choice of $\mathbf{b}$ is not unique.

To make the problem well defined, we add a constraint by requiring that

$$0 = \bar{a} = \frac{1}{n}\sum a_i = \mathbf{v}^T \cdot \frac{1}{n}\sum(\mathbf{x}_i - \mathbf{b}) = \mathbf{v}^T \cdot (\bar{\mathbf{x}} - \mathbf{b})$$

This yields that $\mathbf{b} = \bar{\mathbf{x}} = \frac{1}{n}\sum \mathbf{x}_i$, i.e., we only consider lines passing through the centroid of the data set.

We have thus eliminated the variable $\mathbf{b}$ from the problem, so that we only need to focus on the unit-vector variable $\mathbf{v}$ (representing the direction of the line).

Since we now have $\bar{a} = 0$, the variance of the projections is simply

$$\frac{1}{n-1}\sum_{i=1}^{n} a_i^2$$

and we can correspondingly reformulate the original problem as follows:

$$\max_{\mathbf{v}:\, \|\mathbf{v}\|=1} \underbrace{\sum a_i^2}_{\text{scatter}}, \qquad \text{where} \quad a_i = \mathbf{v}^T(\mathbf{x}_i - \bar{\mathbf{x}}).$$

Let us further rewrite the objective function:

$$\begin{aligned}
\sum a_i^2 &= \sum \underbrace{\mathbf{v}^T(\mathbf{x}_i - \bar{\mathbf{x}})}_{a_i} \underbrace{(\mathbf{x}_i - \bar{\mathbf{x}})^T \mathbf{v}}_{a_i} \\
&= \sum \mathbf{v}^T \left[(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T \right] \mathbf{v} \\
&= \mathbf{v}^T \underbrace{\left[\sum (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T \right]}_{:= \mathbf{C}\ (d \times d \text{ matrix})} \mathbf{v} \\
&= \mathbf{v}^T \mathbf{C} \mathbf{v}.
\end{aligned}$$

Remark. The matrix $\mathbf{C}$ is called the sample covariance matrix or scatter matrix of the data. It is square, symmetric, and positive semidefinite, because it is a sum of such matrices!

Accordingly, we have obtained the following (Rayleigh quotient) problem

$$\max_{\mathbf{v}:\|\mathbf{v}\|=1} \mathbf{v}^T \mathbf{C} \mathbf{v}$$

By applying the theorem, we can easily obtain the following result.

Theorem 0.1. Given a set of data points $\mathbf{x}_1, \ldots, \mathbf{x}_n$ in $\mathbb{R}^d$ with centroid $\bar{\mathbf{x}} = \frac{1}{n}\sum \mathbf{x}_i$, the optimal direction for projecting the data (in order to have maximum variance) is the largest eigenvector of the sample covariance matrix $\mathbf{C} = \sum(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T$:

$$\max_{\mathbf{v}:\,\|\mathbf{v}\|=1} \mathbf{v}^T \mathbf{C} \mathbf{v} = \underbrace{\lambda_1}_{\text{max scatter}}, \qquad \text{achieved when } \mathbf{v} = \mathbf{v}_1.$$

Remark. It can be shown that

$$\max_{\mathbf{v}:\|\mathbf{v}\|=1,\mathbf{v}_1^T\mathbf{v}=0} \mathbf{v}^T\mathbf{C}\mathbf{v} = \lambda_2, \qquad \text{achieved when} \quad \mathbf{v} = \mathbf{v}_2;$$

$$\max_{\mathbf{v}:\,\|\mathbf{v}\|=1,\mathbf{v}_1^T\mathbf{v}=0,\mathbf{v}_2^T\mathbf{v}=0} \mathbf{v}^T\mathbf{C}\mathbf{v} = \lambda_3, \qquad \text{achieved when} \quad \mathbf{v} = \mathbf{v}_3.$$

This shows that $\mathbf{v}_2, \mathbf{v}_3$ etc. are the next best orthogonal directions.

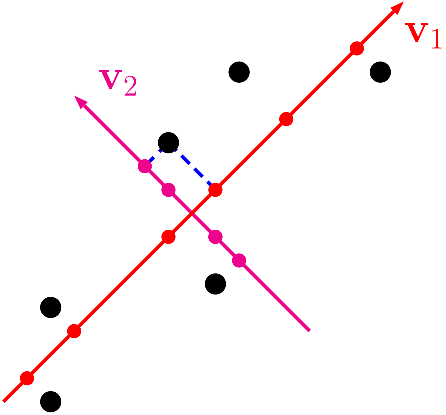
For each $1 \le i \le n$, let

$$a_i = \mathbf{v}_1^T(\mathbf{x}_i - \bar{\mathbf{x}}),$$
$$b_i = \mathbf{v}_2^T(\mathbf{x}_i - \bar{\mathbf{x}}).$$

(so on and so forth for subsequent orthogonal directions).

The scatter of the projections of the data onto each of those directions is

$$\sum a_i^2 = \mathbf{v}_1^T\mathbf{C}\mathbf{v}_1 = \lambda_1$$
$$\sum b_i^2 = \mathbf{v}_2^T\mathbf{C}\mathbf{v}_2 = \lambda_2$$



The total scatter of the k-dimensional PCA projections is equal to the sum of the scatter onto each direction. We prove this for the case of $k = 2$:

$$\sum \|(a_i, b_i) - (0,0)\|^2 = \sum (a_i^2 + b_i^2) = \sum a_i^2 + \sum b_i^2 = \lambda_1 + \lambda_2$$

It is also the maximum possible amount of scatter that can be preserved by all planes of the same dimension.

Furthermore, the orthogonal projections onto different eigenvectors $\mathbf{v}_i$ are uncorrelated: Since $\sum a_i = 0 = \sum b_i$, their covariance is

$$\begin{aligned} \sum a_i b_i &= \sum \mathbf{v}_1^T (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T \mathbf{v}_2 \\ &= \mathbf{v}_1^T \mathbf{C} \mathbf{v}_2 = \mathbf{v}_1^T (\lambda_2 \mathbf{v}_2) = \lambda_2 (\mathbf{v}_1^T \mathbf{v}_2) = 0. \end{aligned}$$

Principal component analysis (PCA)

The previous procedure is called principal component analysis.

- $\mathbf{v}_j$ is called the j**th principal direction**;
- The projection of the data point $\mathbf{x}_i$ onto $\mathbf{v}_j$, i.e., $\mathbf{v}_j^T(\mathbf{x}_i - \bar{\mathbf{x}})$, is called the j**th principal component** of $\mathbf{x}_i$.

In fact, PCA is just a change of coordinate system to use the maximum-variance directions of the data set!

Example 0.1. Perform PCA (by hand) on the following data set (rows are data points):

$$\mathbf{X} = \begin{pmatrix} 1 & -1 \\ -1 & 1 \\ 2 & 2 \\ -2 & -2 \end{pmatrix}.$$

Computing

PCA requires constructing a $d \times d$ matrix from the given data

$$\mathbf{C} = \sum (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T$$

and computing its (top) eigenvectors

$$\mathbf{C} \approx \mathbf{V}_k \mathbf{\Lambda}_k \mathbf{V}_k^T$$

which can be a significant challenge for large data sets in high dimensions.

We show that the eigenvectors of $\mathbf{C}$ can be efficiently computed from the Singular Value Decomposition (SVD) of the centered data matrix.

Let $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_n^T \end{bmatrix} \in \mathbb{R}^{n\times d}$ and $\widetilde{\mathbf{X}} = \begin{bmatrix} \widetilde{\mathbf{x}}_1^T \\ \vdots \\ \widetilde{\mathbf{x}}_n^T \end{bmatrix} \in \mathbb{R}^{n\times d}$ (where $\widetilde{\mathbf{x}}_i = \mathbf{x}_i - \bar{\mathbf{x}}$) be the original and centered data matrices (rows are data points).

Then

$$\mathbf{C} = \sum \widetilde{\mathbf{x}}_i \widetilde{\mathbf{x}}_i^T = [\widetilde{\mathbf{x}}_1 \dots \widetilde{\mathbf{x}}_n] \cdot \begin{bmatrix} \widetilde{\mathbf{x}}_1^T \\ \vdots \\ \widetilde{\mathbf{x}}_n^T \end{bmatrix} = \widetilde{\mathbf{X}}^T \widetilde{\mathbf{X}}.$$

Again, this shows that $\mathbf{C}$ is square, symmetric and positive semidefinite and thus only has nonnegative eigenvalues.

PCA through SVD

Recall that the principal directions of a data set are given by the top eigenvectors of the sample covariance matrix

$$\mathbf{C} = \widetilde{\mathbf{X}}^T \widetilde{\mathbf{X}} \in \mathbb{R}^{d \times d}.$$

Algebraically, they are also the right singular vectors of $\widetilde{\mathbf{X}}$:

$$\widetilde{\mathbf{X}}^T \widetilde{\mathbf{X}} = \mathbf{V} \mathbf{\Sigma}^T \mathbf{U}^T \cdot \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T = \mathbf{V} \underbrace{\left(\mathbf{\Sigma}^T \mathbf{\Sigma} \right)}_{\mathbf{\Lambda}} \mathbf{V}^T$$

Thus, one may just use the SVD of $\widetilde{\mathbf{X}}$ to compute the principal directions (and components), which is much more efficient.

Interpretations:

Let the SVD of a centered data matrix $\widetilde{\mathbf{X}}$ be the following

$$\widetilde{\mathbf{X}} = \mathbf{U} \cdot \boldsymbol{\Sigma} \cdot \mathbf{V}^T = \mathbf{U}\boldsymbol{\Sigma} \cdot \mathbf{V}^T$$

Then

- Columns of $\mathbf{V}$ (right singular vectors $\mathbf{v}_i$) are principal directions;
- Squared singular values ($\lambda_i = \sigma_i^2$) represent amounts of scatter captured by each principal direction;
- Columns of $\mathbf{U}\boldsymbol{\Sigma}$ are different principal components of the data.

To see the last one, consider any principal direction $\mathbf{v}_j$. The corresponding principal component is

$$\widetilde{\mathbf{X}}\mathbf{v}_j = \sigma_j \mathbf{u}_j$$

with scatter $\lambda_j = \sigma_j^2$.

Collectively, for the top k principal directions, the principal components of the entire data set are

$$\begin{aligned} \underbrace{\mathbf{Y}}_{n\times k} &= [\widetilde{\mathbf{X}}\mathbf{v}_1 \ldots \widetilde{\mathbf{X}}\mathbf{v}_k] = \widetilde{\mathbf{X}}[\mathbf{v}_1 \ldots \mathbf{v}_k] & \longleftarrow \widetilde{\mathbf{X}}\mathbf{V}_k \\ &= [\sigma_1\mathbf{u}_1 \ldots \sigma_k\mathbf{u}_k] = [\mathbf{u}_1 \ldots \mathbf{u}_k] \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{bmatrix} & \longleftarrow \mathbf{U}_k\mathbf{\Sigma}_k. \end{aligned}$$

Note also the following:

- The total scatter preserved by the k-dimensional projections is

$$\sum_{1\le j\le k} \lambda_j = \sum_{1\le j\le k} \sigma_j^2.$$

- A parametric equation of the k-dimensional PCA plane is

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{V}_k \boldsymbol{\alpha}$$

- Projections of the data onto this plane are given by the rows of

$$\mathcal{P}_S(\mathbf{X}) = \mathbf{1}\bar{\mathbf{x}}^T + \underbrace{\widetilde{\mathbf{X}}\mathbf{V}_k}_{\mathbf{Y}} \mathbf{V}_k^T$$

An SVD-based algorithm for PCA

Input: Data matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ and integer k (with $0 < k < d$)

Output: Top k principal directions $\mathbf{v}_1, \ldots, \mathbf{v}_k$ and corresponding principal components $\mathbf{Y} \in \mathbb{R}^{n \times k}$.

Steps:

1. Center data: $\widetilde{\mathbf{X}} = [\mathbf{x}_1 - \bar{\mathbf{x}}, \ldots, \mathbf{x}_n - \bar{\mathbf{x}}]^T$ where $\bar{\mathbf{x}} = \frac{1}{n} \sum \mathbf{x}_i$
2. Perform rank-k SVD: $\widetilde{\mathbf{X}} \approx \mathbf{U}_k \mathbf{\Sigma}_k \mathbf{V}_k^T$
3. Return: $\mathbf{Y} = \widetilde{\mathbf{X}} \mathbf{V}_k = \mathbf{U}_k \mathbf{\Sigma}_k$

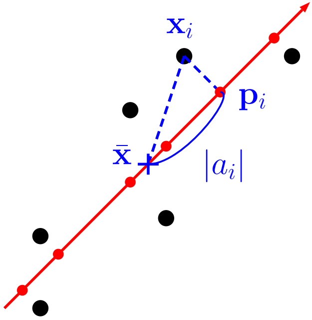
Connection to orthogonal least-squares fitting

We have seen that the following two planes coincide:

(1) **PCA plane**: which maximizes the projection variance,

(2) **Orthogonal best-fit plane**: which minimizes the orthogonal least-squares fitting error.

Mathematical justification:



$$\underbrace{\sum \|\mathbf{x}_i - \bar{\mathbf{x}}\|^2}_{\text{total scatter}} = \underbrace{\sum a_i^2}_{\text{proj. var.}} + \underbrace{\sum \|\mathbf{x}_i - \mathbf{p}_i\|^2}_{\text{ortho. fitting error}} \qquad \underbrace{\mathbf{p}_i}_{\text{proj}} = \mathbf{v} \cdot \underbrace{\mathbf{v}^T(\mathbf{x}_i - \bar{\mathbf{x}})}_{\text{p.c.}} + \bar{\mathbf{x}}$$

Other interpretations of PCA

The PCA plane also tries to preserve, as much as possible, the Euclidean distances between the given data points:

$$\|\mathbf{y}_i - \mathbf{y}_j\|_2 \approx \|\mathbf{x}_i - \mathbf{x}_j\|_2 \quad \text{for “most” pairs } i \neq j$$

More on this when we get to the MDS part.

PCA can also be regarded as a **feature extraction** method:

$$\mathbf{v}_j = \frac{1}{\lambda_j}\mathbf{C}\mathbf{v}_j = \frac{1}{\lambda_j}\widetilde{\mathbf{X}}^T(\widetilde{\mathbf{X}}\mathbf{v}_j) \in \mathrm{Col}(\widetilde{\mathbf{X}}^T), \quad \text{for all } j < \mathrm{rank}(\widetilde{\mathbf{X}})$$

This shows that each $\mathbf{v}_j$ is a linear combination of the centered data points (and also a linear combination of the original data points).

MATLAB implementation of PCA

MATLAB built-in: **[V, US] = pca(X);** % *Rows of X are observations*

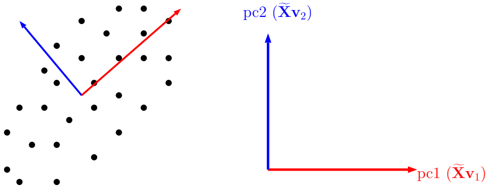
Alternatively, you may want to code it yourself:

```
Xtilde = X - mean(X,1);
[U,S,V] = svds(Xtilde, k); % k is the reduced dimension
Y = Xtilde*V;
```

Application to data visualization

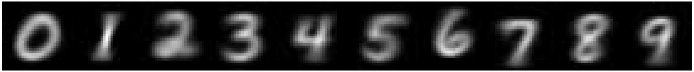
Given a high dimensional data set $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$, one can visualize the data by

- projecting the data onto a 2 or 3 dimensional PCA plane and
- plotting the principal components as new coordinates



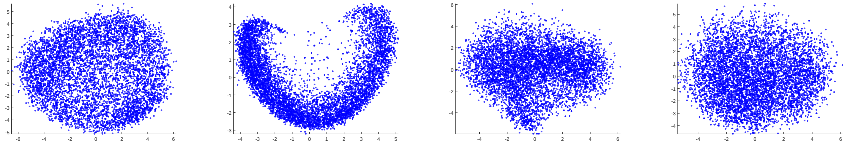
2D visualization of MNIST handwritten digits

1. The "average" writer



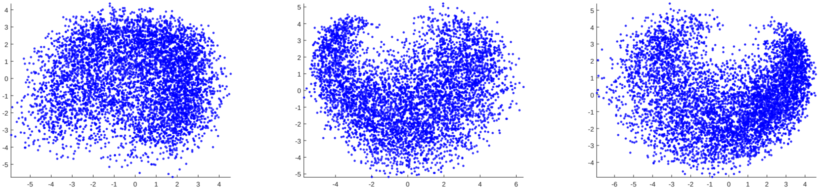
2. The full appearance of each digit class

0 - 3

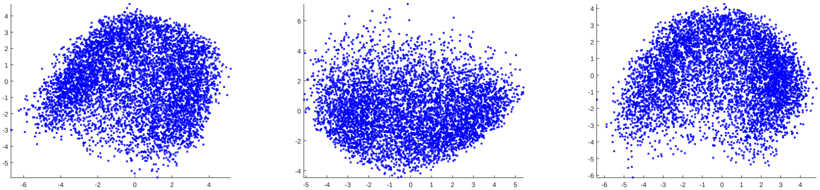


(cont'd on next page)

4-6


4 3 2 1 0 -1 -2 -3 -4 -5
-5 -4 -3 -2 -1 0 1 2 3 4
5 4 3 2 1 0 -1 -2 -3 -4 -5
-4 -2 0 2 4 6
5 4 3 2 1 0 -1 -2 -3 -4
-6 -5 -4 -3 -2 -1 0 1 2 3 4

7-9


4 3 2 1 0 -1 -2 -3 -4 -5
-6 -4 -2 0 2 4
6 4 2 0 -2 -4
-5 -4 -3 -2 -1 0 1 2 3 4 5
4 3 2 1 0 -1 -2 -3 -4 -5 -6
-6 -5 -4 -3 -2 -1 0 1 2 3 4

How to set the parameter k in other settings?

Generally, there are two ways to choose the reduced dimension k:

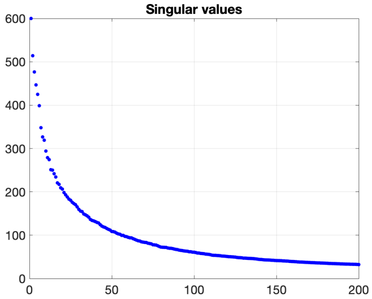
- Set $k = \#$"dominant" singular values
- Choose k such that the top k principal components explain a certain fraction of the total scatter of the data:

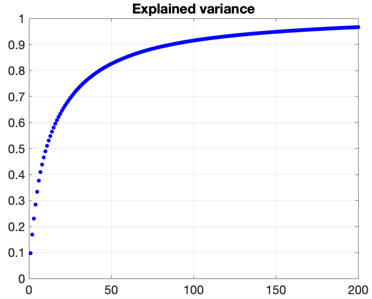
$$\underbrace{\sum_{i=1}^{k} \sigma_i^2}_{\text{explained variance}} \quad / \quad \underbrace{\sum_{i=1}^{r} \sigma_i^2}_{\text{total variance}} \quad > \quad p.$$

  Common values of p are $.95$ (the most commonly used), or $.99$ (more conservative, less reduction), or $.90, .80$ (more aggressive).

However, in practical contexts, it is possible to get much lower than this threshold while maintaining or even improving the accuracy.

Example: MNIST handwritten digits





Concluding remarks on PCA

PCA projects the (centered) data onto a k-dim plane that

- maximize the amount of variance in the projection domain,
- minimizes the orthogonal least-squares fitting error

As a dimension reduction and feature extraction method, it is

- unsupervised (blind to labels),
- nonparameteric (model-free), and
- very popular!

PCA is a deterministic procedure, assuming no measurement errors in the data:

$$\widetilde{\mathbf{X}} = \mathbf{Y} \cdot \mathbf{V}^T$$

To extend it to deal with measurement errors, we can assume a statistical model

$$\widetilde{X}_{ij} = \sum_{r=1}^{k} F_{ir} w_{rj} + \epsilon_{ij}, \quad \text{for all } i, j$$

which in matrix form is

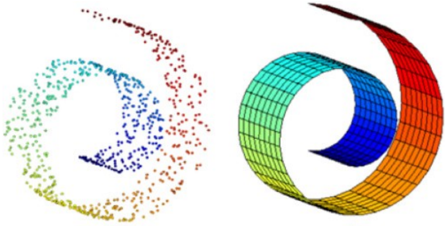
$$\widetilde{\mathbf{X}} = \underbrace{\mathbf{F}}_{\text{factor scores}} \cdot \underbrace{\mathbf{W}}_{\text{factor loadings}} + \underbrace{\mathbf{E}}_{\text{errors}}$$

This method is called **factor analysis** and its solution can be derived by using the MLE approach.

Lastly, PCA is a linear projection method:

$$\mathbf{y} = \mathbf{V}^T(\mathbf{x} - \bar{\mathbf{x}})$$

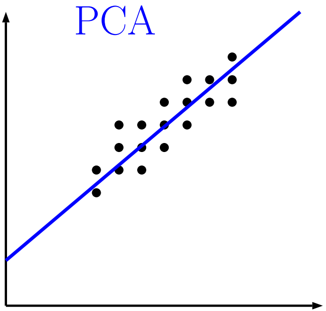
For nonlinear data, PCA will need to use a dimension higher than the manifold dimension (in order to preserve most of the variance).

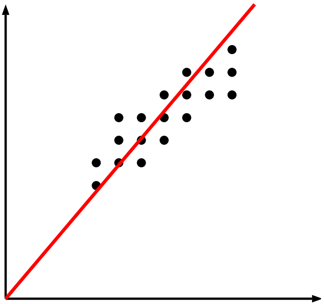


(a) (b)

On the matter of centering

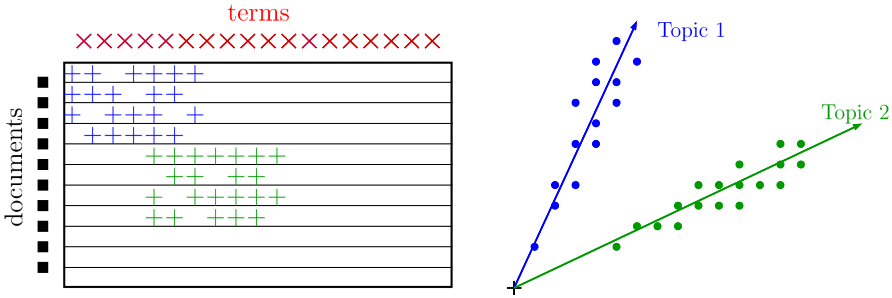
PCA requires data centering (equivalent to fitting a plane through the centroid). What is the best plane through the origin (linear subspace)?





Why using linear subspaces?

They are very useful for modeling document collections:



How to fit a plane through the origin?

Theorem 0.2. Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ be the given data set and $k \geq 1$ an integer. The best k-dimensional plane through the origin for fitting the data is spanned by the top k right singular vectors of $\mathbf{X}$:

$$\underbrace{\mathbf{X}}_{\text{given data}} \approx \underbrace{\mathbf{X}_k}_{\text{projections}} = \underbrace{\mathbf{U}_k \mathbf{\Sigma}_k}_{\text{coefficients}} \underbrace{\mathbf{V}_k}_{\text{basis}}$$

Proof. It suffices to solve

$$\min_{\mathbf{V} \in \mathbb{R}^{d \times k}:\, \mathbf{V}^T \mathbf{V} = \mathbf{I}_k} \|\mathbf{X} - \mathbf{X}\mathbf{V}\mathbf{V}^T\|^2$$

The optimal $\mathbf{V}$ is such that $\mathbf{X}\mathbf{V}\mathbf{V}^T = \mathbf{X}_k$, and can be chosen to be $\mathbf{V}_k$.

Example 0.2. Consider a data set of 3 points in $\mathbb{R}^2$:

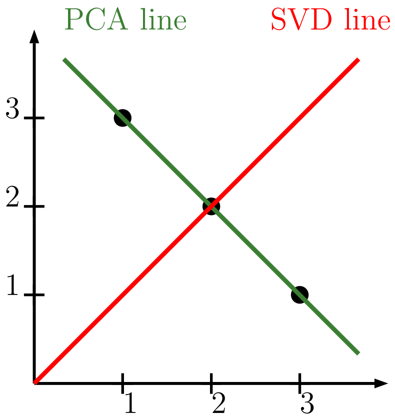
$$(1, 3), (2, 2), (3, 1).$$

The PCA line is

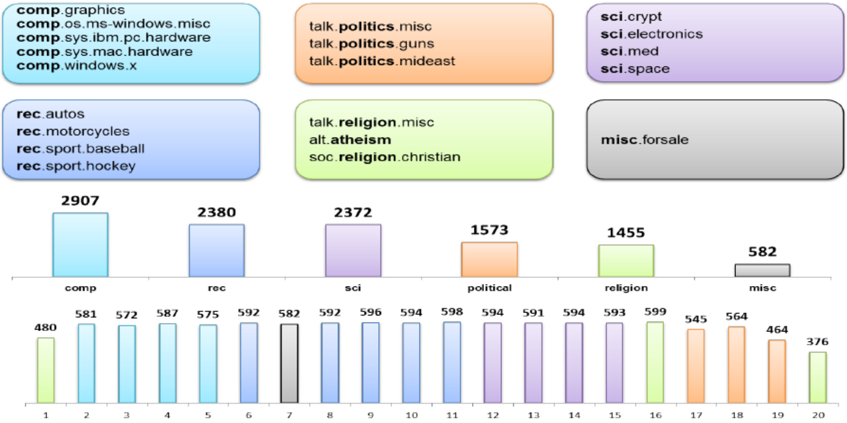
$$\mathbf{x}(t) = (2, 2) + t \cdot \left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right),$$

while the SVD line (best-fit line through the origin) is

$$\mathbf{x}(t) = t \cdot \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right).$$



Application: Visualization of 20 newsgroups data



Summary information:

- 18,774 documents partitioned nearly evenly across 20 different newsgroups.
- A total of 61,118 unique words (including stopwords) present in the corpus.

A significant challenge:

- The stopwords dominate in most documents in terms of frequency and make the newsgroups very hard to be .

A fake document-term matrix:

	the	an	zzzz	math	design	car	cars
doc 1	8	12	1	4	2		
doc 2	7	10		3	4		
doc 3	9	15		5	2		
doc 4	5	9			2	2	2
doc 5	9	7			3	3	1
doc 6	1	1				2	

We will not use any text processing software to perform stopword removal (or other kinds of language processing such as stemming), but rather rely on the following statistical operations (in the shown order) on the document-term frequency matrix $\mathbf{X}$ to deal with stopwords:

1. Convert all the frequency counts into binary (0/1) form

	the	an	zzzz	matrix	design	car	cars
doc 1	1	1	1	1	1		
doc 2	1	1		1	1		
doc 3	1	1		1	1		
doc 4	1	1			1	1	1
doc 5	1	1			1	1	1
doc 6	1	1				1	

2. Remove words that occur either in exactly one document (rare words or typos) or in “too many“ documents (stopwords or common words)

	math	design	car	cars
doc 1	1	1		
doc 2	1	1		
doc 3	1	1		
doc 4		1	1	1
doc 5		1	1	1
doc 6			1	
6	3	5	3	1

3. Apply the inverse document frequency (IDF) weighting to the remaining columns of $\mathbf{X}$:

$$\mathbf{X}(:,j) \leftarrow w_j \cdot \mathbf{X}(:,j), \qquad w_j = \log(n/n_j),$$

where n_j is the number of documents that contain the j-th word

	math	design	car	cars
doc 1	0.6931	0.1823		
doc 2	0.6931	0.1823		
doc 3	0.6931	0.1823		
doc 4		0.1823	0.6931	1.0986
doc 5		0.1823	0.6931	1.0986
doc 6			0.6931	

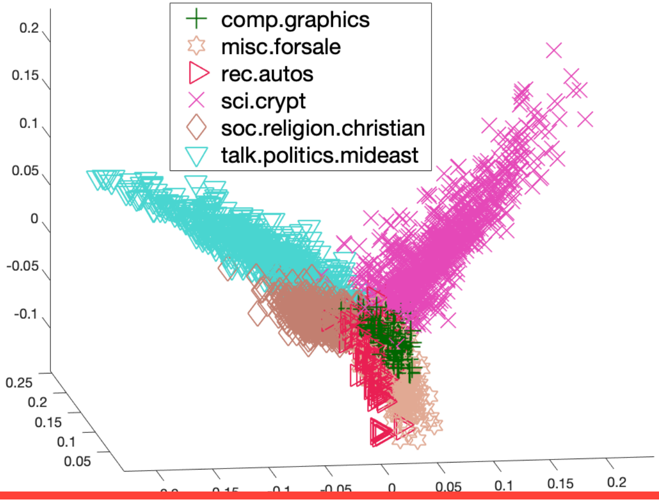
4. Rescale the rows of $\mathbf{X}$ to have unit norm in order to remove the documents' length information

	math	design	car	cars
doc 1	0.9671	0.2544		
doc 2	0.9671	0.2544		
doc 3	0.9671	0.2544		
doc 4		0.1390	0.5284	0.8375
doc 5		0.1390	0.5284	0.8375
doc 6			1	

By applying the above procedure (a particular TF-IDF weighting scheme[1]) to the 20newsgroups data and keeping only the words with frequencies between 2 and 939 (average cluster size), we obtain a matrix of 18,768 nonempty documents and 55,570 unique words, with average row sparsity 73.4.

For ease of demonstration, we focus on six newsgroups in the processed data set (one from each category) and project them by SVD into a 3-dimensional plane through the origin for visualization.

[1]Full name: term frequency inverse document frequency.
See https://en.wikipedia.org/wiki/Tf-idf


comp.graphics
misc.forsale
rec.autos
sci.crypt
soc.religion.christian
talk.politics.mideast
0.2
0.15
0.1
0.05
0
-0.05
-0.1
0.25
0.2
0.15
0.1
0.05
0
0.05
0.1
0.15
0.2

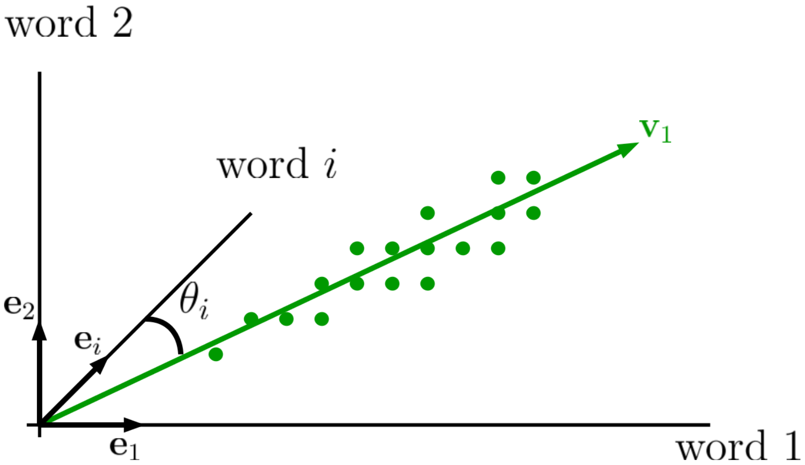
We also display the top 20 words that are the most "relevant" to the underlying topic of each class.

To rank the words based on relevance to each newsgroup, we first compute the top right singular vector $\mathbf{v}_1$ of a fixed newsgroup (without centering), which represents the dominant direction of the cluster.

Each keyword i corresponds to a distinct dimension of the data and is represented by $\mathbf{e}_i$.

The following score can then be used to measure and compare the relevance of each keyword:

$$\mathrm{score}(i) = \cos\theta_i = \langle \mathbf{v}_1, \mathbf{e}_i \rangle = v_1(i), \quad i = 1, \ldots, 55570$$


word 2
$\mathbf{v}_1$
word i
θ_i
$\mathbf{e}_2$
$\mathbf{e}_i$
$\mathbf{e}_1$
word 1

Principal Component Analysis (PCA)