S&DS 365 / 565

Data Mining and Machine Learning

Unsupervised Learning: Principal Component Analysis (PCA)

Yale

Principal component analysis (PCA) is one of the most used unsupervised learning methods.

We will explore five different interpretations of PCA

Principal component analysis (PCA) is one of the most used unsupervised learning methods.

We will explore five different interpretations of PCA

The first is a data variance/dimensionality reduction approach

Principal component analysis (PCA) is one of the most used unsupervised learning methods.

We will explore five different interpretations of PCA

The first is a data variance/dimensionality reduction approach

The second is a data representation approach

Principal component analysis (PCA) is one of the most used unsupervised learning methods.

We will explore five different interpretations of PCA

The first is a data variance/dimensionality reduction approach

The second is a data representation approach

The third is a data denoising approach

Principal component analysis (PCA) is one of the most used unsupervised learning methods.

We will explore five different interpretations of PCA

The first is a data variance/dimensionality reduction approach

The second is a data representation approach

The third is a data denoising approach

The fourth is a low-rank approximation approach

Principal component analysis (PCA) is one of the most used unsupervised learning methods.

We will explore five different interpretations of PCA

The first is a data variance/dimensionality reduction approach

The second is a data representation approach

The third is a data denoising approach

The fourth is a low-rank approximation approach

The fifth is a Gaussian covariance based approach

Before we begin we need to understand what a “direction” of data means

We have said before that PCA finds “interesting directions” of data

PCA finds components that are principle

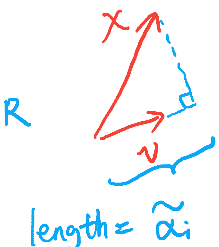
We have data x_i with $i \in [n]$ $x_i \in \mathbb{R}^p$

A **direction** is simply a unit vector $v \in \mathbb{R}^p$ i.e. $\|v\|_2 = 1$.

单位向量

We then project the data on that direction and have projected data.

$\widetilde{\alpha}_i = \langle x_i, v \rangle$ $\in \mathbb{R}$



In this way we have **dimensionality reduction**

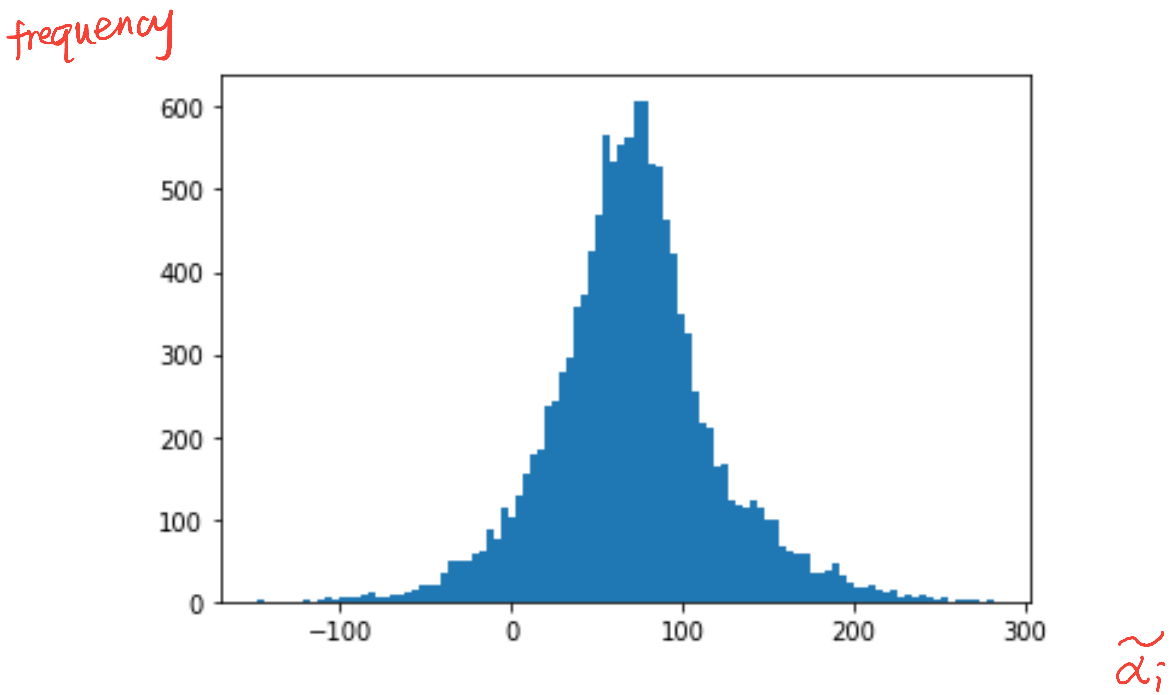
Our original data was in p dimensions, now 1 dimension.

only need $\widetilde{\alpha}_i$ to represent x

What does this projection mean? Let's take a look.

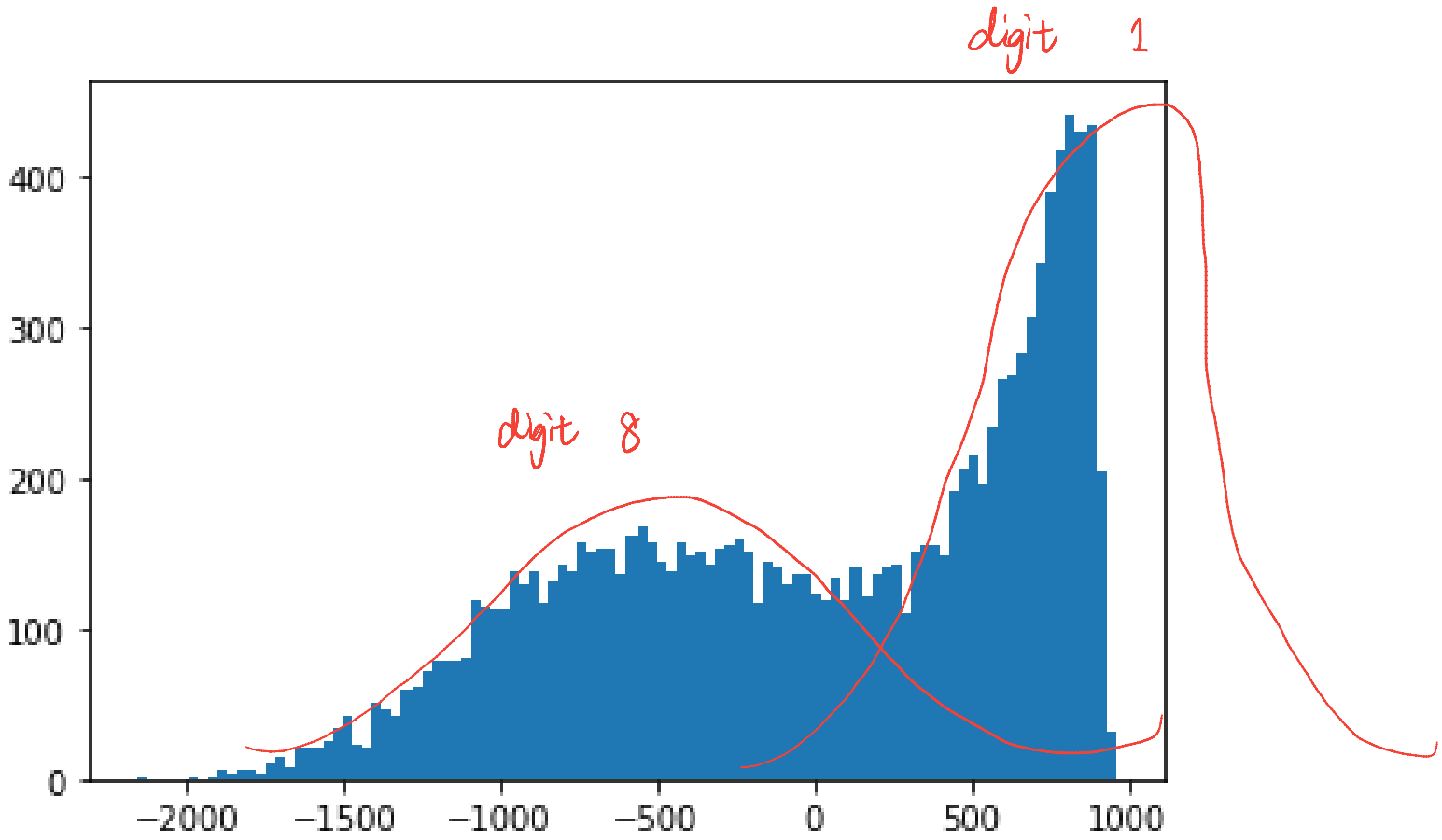
Let's remember the digits example

Let's take a random unit vector v and plot a histogram of $\widetilde{\alpha}_i$.



Not very informative. Just looks pretty Gaussian.

Let’s pick a direction provided by PCA.



Way more interesting.

PCA assume unit vectors are orthogonal make computation easier

For other no linear methods, don't need orthogonality no meaning for that

We can do this in higher dimensions. Suppose we have **two directions** now u_1 and u_2. We assume that the two unit vectors are orthogonal $\langle u_1, u_2 \rangle = 0$ and unit norm (orthonormal).

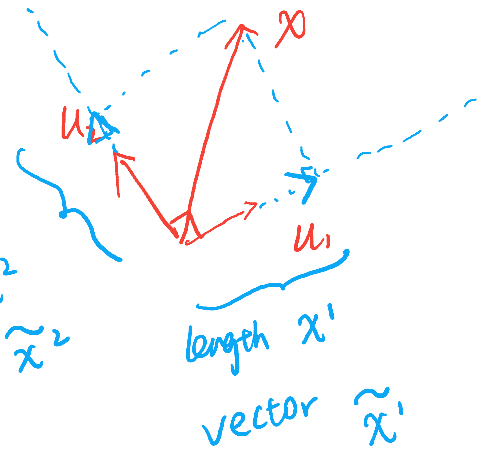
$u_1 \perp u_2$

Now take

(x_i^1, x_i^2) is coordinate of x if base is u_1, u_2

a linear function

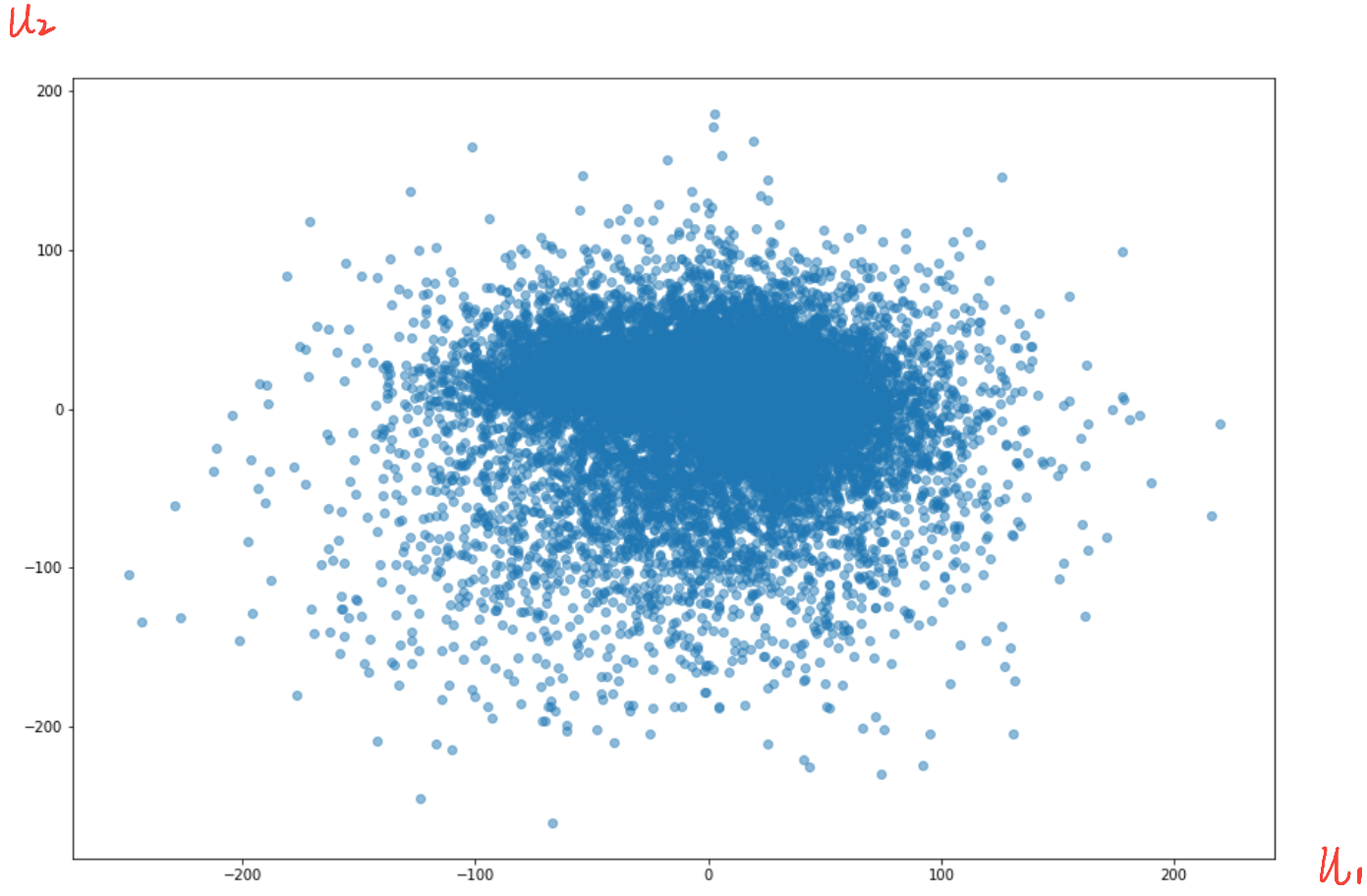
$$x_i^1 = \langle x_i, u_1 \rangle$$
$$x_i^2 = \langle x_i, u_2 \rangle$$



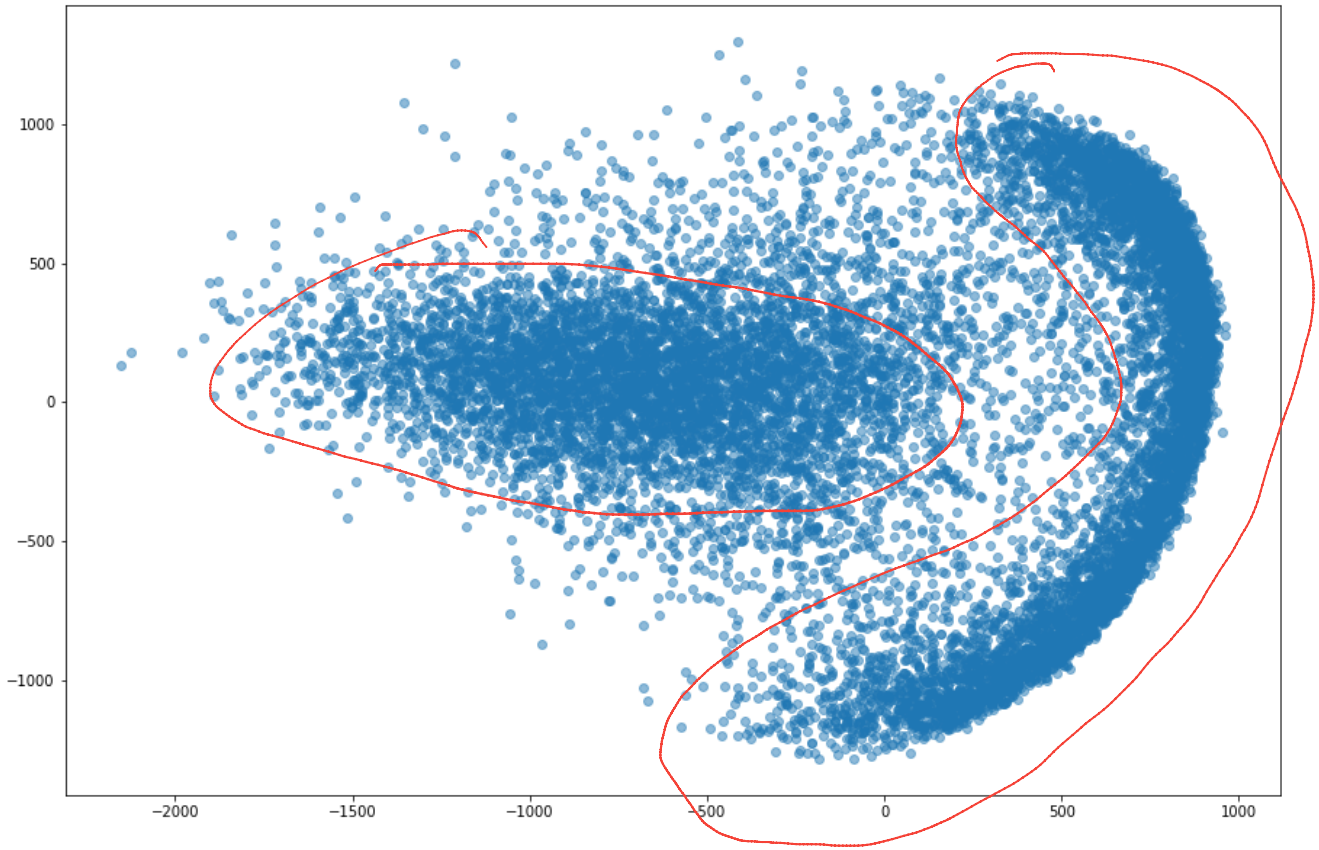
$x \in R^p \quad \tilde{x}^1, \tilde{x}^2 \in R^p$

$u_1, u_2 \in R^p \quad \tilde{x}^1 + \tilde{x}^2 = x$

We can plot those coordinates now as well for two random orthonormal vectors.



Use the PCA components instead



Definitely see more structure.

Dimensionality reduction for visualization

Example: Senate voting

$x_i \in \{-1, 0, +1\}^p$, voting record for Senator i

↙ disaprove

↑ no vote

↑ approve bill

p bills

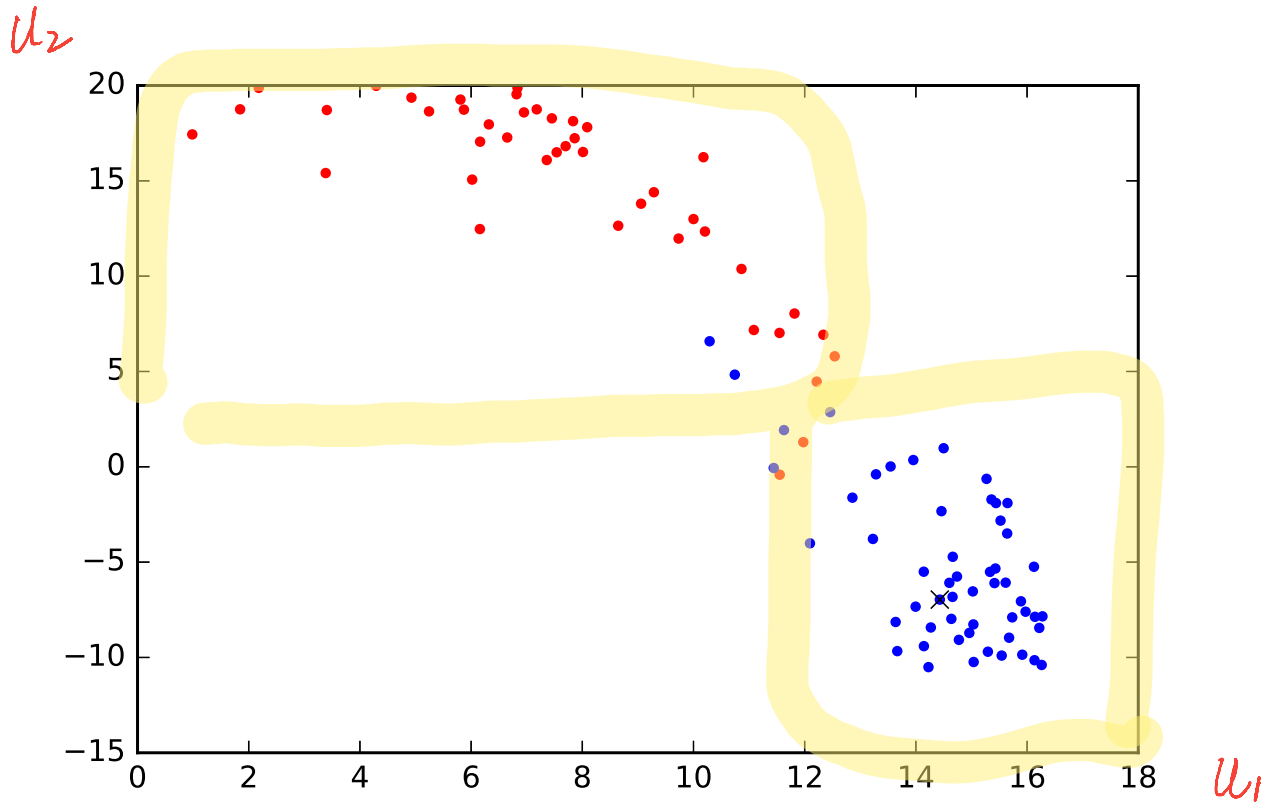
$x \in R^p$

Dimensionality reduction for visualization

Example: Senate voting

see Notebook

2 principle components

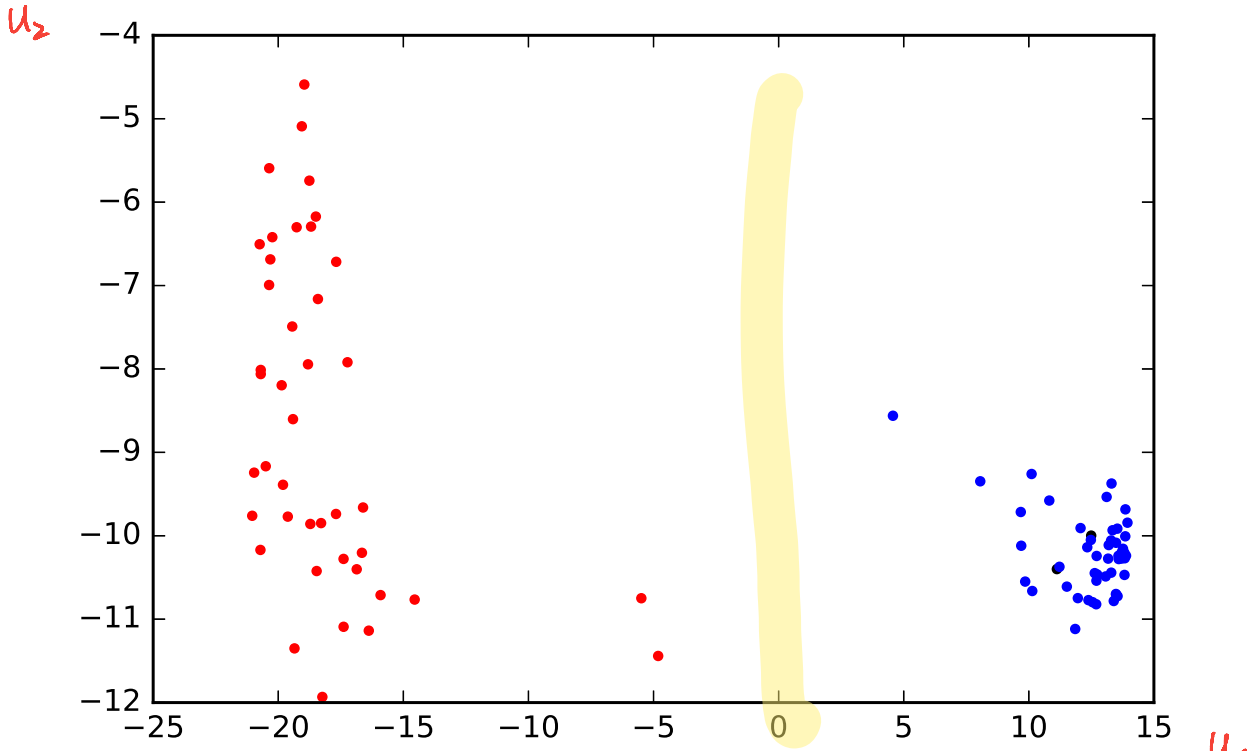


1989

Dimensionality reduction for visualization

Example: Senate voting

(a_1, a_2) 2-D coordinate of data points



partisan

2015

For general base matrix A (not orthogonal) $A \in R^{p \times k}$

$\text{rank}(A) = k \quad p \geq k$

$\tilde{x} = \arg\min_{\alpha} \|A\alpha - x\|_2^2$

least square solution

For dictionary learning

let $k \gg p \Rightarrow \alpha$ to be sparse

α best representative of x

When A is orthogonal

solution is $\tilde{x} = A^T x$

So far $1 - D$ or $2 - D$ projections

K-dimensional projection:

$U = \left[u_1 \; u_2 \; \cdots \; u_k \right] \} p$

U is orthoganal matrix

That is let $U \in \mathbb{R}^{p \times k}$ be a matrix such that $U^T U = I$. Then

k

$$\mathbb{R}^k \ni \tilde{x}_i = U^T x_i$$

$k \times 1 \quad = \quad k \times p \quad p \times 1$

What motivates this?

First Interpretation

preserve variance (size) of data

energy of x_i

Norm-preserving projections

$X = [x_1 \; x_2 \; \cdots \; x_p]$ (p columns, n rows)

Let $X \in \mathbb{R}^{p \times n}$ with columns x_i. (Note this is different than regression where this would be X^T)

Find an orthoganal matrix U — maximize — (what's the best space K that we should focus on coz x_i has largest energy on that space

$$U = \arg \max_{A | A^T A = I} \sum_{i=1}^{n} \|A^T x_i\|_2^2$$

The part we loose when we do this is the part I live in orthogonal complement of K

rewrite as

$$\sum_{i=1}^{n} \|A^T x_i\|_2^2 = \|A^T X\|_F^2$$

(Matrix Norm) — L_2-norm square of matrix

Frobenius norm: $\|B\|_F^2 = \sum_{ij} B_{(ij)}^2 = \text{trace}(BB^T) = \langle B, B \rangle$ is the Euclidean norm for matrices.

see before

The optimization is non-convex, but has a closed form solution involving the **singular value decomposition (SVD)**.

Basic facts about a orthonormal collections of vectors $U \in \mathbb{R}^{p \times k}$ and orthonormal projections

orthonormal matrix

列空间

$K = \text{colspace}(U)$ is a k-dimensional subspace

$U = \begin{bmatrix} | & | & & | \\ u_1 & u_2 & \cdots & u_k \\ | & | & & | \end{bmatrix}$

k个列向量 $u_i \in R^p$

Columns u_i of U form basis vectors for the space

Take $v \in K$ vector $v \in R^k$ can be written as a linear combination of u_i

$$v = \sum_{i=1}^{k} \alpha_i u_i = U\alpha$$

For any vector $v \in k$, can find unique vector $\alpha \in R^k$

Nice property: $\alpha = U^T v$

$\alpha \in \mathbb{R}^k \Rightarrow$ coefficients of the basis vectors $\Rightarrow$ coordinates in K

e_1, e_2 are standard base vectors

$e_i = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ 0 \end{bmatrix}$ ← ith coordinate

Example: $u_1 = e_1 \in \mathbb{R}^p$, $u_2 = e_2 \in \mathbb{R}^p$

For any $v \in K$, v can be written as

$v = (\alpha_1, \alpha_2, 0, 0, 0, \ldots, 0)^T$

$U = \begin{bmatrix} | & | \\ e_1 & e_2 \\ | & | \end{bmatrix}$

$v = \alpha_1 e_1 + \alpha_2 e_2 \in K$

$\begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}_{2\times 1} = \underset{2\times p}{\begin{bmatrix} - e_1 - \\ - e_2 - \end{bmatrix}} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{p\times 1}$ (U^T)

$$\alpha = U^T v$$

$\alpha \Rightarrow$ coordinates (α_1, α_2)

$\begin{cases} \alpha_1 = e_1^T v \\ \alpha_2 = e_2^T v \end{cases}$

Orthonormal basis need not be unique

take G is rotation matrix or any orthonormal transformation

$G \in \mathbb{R}^{k \times k}$ with $G^T G = I$ $= GG^T$

G is orthonormal matrix

Let $\widetilde{U} = UG$, $\widetilde{U}_i$ is orthonormal (why?) — column vectors

In general

$\widetilde{U} \neq U$, but $\text{colspace}(U) = \text{colspace}(\widetilde{U})$

recall previous eg

$u_1 = e_1 \quad u_2 = e_2$

$\tilde{u}_1 = \frac{e_1+e_2}{\sqrt{2}} \quad \tilde{u}_2 = \frac{e_1-e_2}{\sqrt{2}}$

$\text{Span}\{e_1, e_2\} = \text{Span}\{\tilde{u}_1, \tilde{u}_2\}$

but $e_1 \neq \tilde{u}_1$, $e_2 \neq \tilde{u}_2$

Show U is orthonormal $\Leftrightarrow$ Show column of U is orthonormal ($\tilde{u}$)

$\tilde{u}^T \tilde{u} = I_{k \times k}$

Because we are projecting data $\in \mathbb{R}^p$ to $\mathbb{R}^k$ $p > k$
so we loose some info, But we want to know these info.
so we have ↓
正交互补空间

Orthogonal complement of K

Take $w \in \mathbb{R}^p$ with $U^T w = 0_k$ $\in \mathbb{R}^k$ mean w is orthoganal to every column of matrix U

then w is orthoganal to every linear combination of column of U ↓ vector v

Then for all $v \in K$, $\langle w, v \rangle = 0$ ←

w lives in orthogonal complement of K denoted $K^\perp$

i.e. w lives in Null Space (kernel) of U^T for $v \in K$

$UU^T(w + v) = v$ (why?)

Another nice property: Energy can be decomposed between K and $K^\perp$

For $v \in K$ $\|v\|_2^2 = \|U\alpha\|_2^2 = \alpha^T \underbrace{U^T U}_{I} \alpha = \|\alpha\|_2^2 = \|U^T v\|_2^2$.

energy of v

General vector $v \in \mathbb{R}^p \Rightarrow v = UU^T v + w$ where $w = v - UU^T v \in K^\perp$

Show $w \in K^\perp$ ↘

$U^T w = U^T(v - UU^T v) = 0$ $= U^T v - \underbrace{U^T U}_{I} U^T v$

$$\|v\|_2^2 = \underbrace{\|UU^T v\|_2^2}_{\text{part of } v \text{ that lives in } K} + \underbrace{\|(I - UU^T)v\|_2^2}_{\text{part of } v \text{ that lives in } K^\perp}$$

Orthogonal projections

$K = \text{colspace}\, U \subset \mathbb{R}^p$ is k-dimensional

$$P_K(v) = \arg\min_{g \in K} \|v - g\|_2^2$$

$\exists \alpha \in \mathbb{R}^k$ with $g = U\alpha$

$$P_K(v) = U(\arg\min_{\alpha \in \mathbb{R}^k} \|v - U\alpha\|_2^2)$$

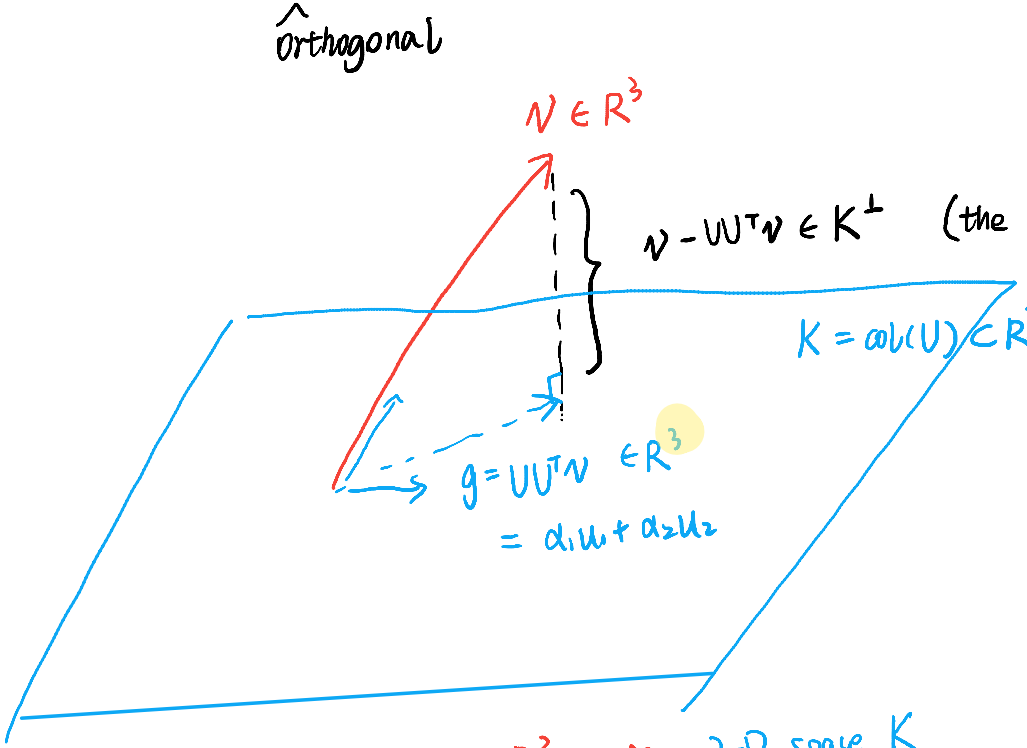
Linear regression again $\widehat{\alpha} = (U^T U)^{-1} U^T v$

$$U^T U = I \implies \widehat{\alpha} = U^T v \implies P_K(v) = UU^T v.$$

UU^T is an **orthogonal projection matrix** onto the space K.

Note that $v \neq UU^T v$ in general unless $v \in K$.

Illustration of projections



$U \in R^{3\times 2}$ $u_1, u_2 \in R^3$

$U = \begin{bmatrix} | & | \\ u_1 & u_2 \\ | & | \end{bmatrix}$

$a = (u_1, u_2) \in R^2$

orthogonal project $v \in R^3$ onto 2-D space K

is try to minimize l_2-norm of error vector $(v - UU^T v) \in K^\perp$ $\Rightarrow$ projection matrix is $UU^T = P_K$

we can show $I - UU^T = P_{K^\perp}$

The orthogonal projection matrix of projection onto orthogonal complement of K is $I - UU^T$

it minimize the distance between v and projected v

$\uparrow$

$(I - UU^T)v$

Column vector of orthonormal matrix A

The orthonormal basis vectors are each directions. Now we return to PCA and the SVD.

$$U = \arg \max_{A | A^T A = I} \|A^T X\|_F^2$$

⇒ effectively find those coefficients that represent the projection of X onto space that spanned by A

How to solve it ?

Recall $X \in \mathbb{R}^{p \times n}$. Assume $n \geq p$.

For any matrix $X \in R^{p \times n}$

Singular value decomposition (SVD): $X = USV^T$ where $U \in \mathbb{R}^{p \times p}$ is an orthonormal matrix, $S \in \mathbb{R}^{p \times p}$ is diagonal with positive entries, and $V^T \in \mathbb{R}^{p \times n}$ is orthonormal.

the rows of V^T is orthonormal

the columns of V ~

Equivalently: $U \in \mathbb{R}^{p \times p}$ and $V \in \mathbb{R}^{n \times n}$ with $S \in \mathbb{R}^{p \times n}$ where $S_{(ij)} \geq 0$ and $S_{(ij)} = 0$ for $i \neq j$. diagonal matrix with positive entries

$X = USV^T = [\ ]_{p \times p} [\ ]_{p \times n} [\ ]_{n \times n}$

$\text{rank}(X) = \text{nnz}(S)$ (**n**umber of **n**on-**z**eros)

scalar, column vectors

$$X = \sum_{i=1}^{\text{rank}(X)} S_{(ii)} u_i v_i^T$$

Convention: $S_{(ii)} \geq S_{(jj)}$ for $i \leq j$. $S = \begin{bmatrix} 9 & & & & \\ & 8 & & & \\ & & 7 & & 0 \\ & & & 6 & \end{bmatrix}$

Diagonal entry of S descending order

u_i are the **left singular vectors**

sample correlation matrix of X

if $E(X) = 0$ $\hat{\Sigma} = \frac{1}{n} XX^T$

v_i are the **right singular vectors**

Diagonal Matrix property: ① eigen value decompo ② eigen value $\in R$

eigen value of $\hat{\Sigma} \geq 0 \Rightarrow \hat{\Sigma} = UDV^T$

means The variation of $\hat{\Sigma}$ is captured by singular vectors

Diagonal entry of S

$\sigma_i = S_{(ii)}$ are the **singular values**

$S_{(11)} \sim S_{(kk)} \rightarrow$ { left: $u_1 \sim u_k$; right: $v_1 \sim v_k$ }

We say the top-k singular vectors are those corresponding to the k largest singular values.

How to solve $U = \arg\max_{A|A^TA=I} \|A^TX\|_F^2$

How to solve $U = \arg\max_{A|A^TA=I} \|A^T X\|_F^2$

Plug in $X = USV^T$
$U \in R^{p\times p}$ $V \in R^{n\times n}$ $S \in R^{p\times n}$

$$\|A^T USV^T\|_F^2 = \|A^T US\|_F^2$$

Multiply a matrix V by right or left whose rows are orthonormal
The Febenius-norm square remain the same ①

Solution ✓

Guess $A = U_k$

ith column of U_k

$U_k \in \mathbb{R}^{p\times k}$, $[U_k]_{:,i} = u_i$ (matrix whose columns are the top k left singular vectors of the matrix X)

Then, Plug in $A = U_k$

This means the energy of X is captured by singular value of X : σ_i

$$\|U_k^T US\|_F^2 = \sum_{i=1}^{k} \sigma_i^2$$

Proof in graph Notes

Also turns out:
$\|X\|_F^2 = \|USV^T\|_F^2 = \|S\|_F^2 = \sum_{i=1}^{\text{rank}(X)} \sigma_i^2$
both are orthonormal matrix according to ①

The leftover energy $= \sum_{i=k+1}^{\text{rank}(X)} \sigma_i^2$

sum of the least (rank(X) - k) squared singular values

Show our guess is right There isn't any other measure that maximize $\|A^T U S\|_F^2$

On the other hand

$$\|A^T U S\|_F^2 = \|\widetilde{U} S\|_F^2 \qquad \text{Define } A^T U = \widetilde{U}$$
$$= \sum_i^k \sigma_i^2 \|\widetilde{u}_i\|_2^2$$

where we take $\widetilde{u}_i = A^T u_i$ where u_i is the i^{th} column of U. Clearly, $0 \leq \|\widetilde{u}_i\|_2^2 \leq 1$. Furthermore, $\sum_i^k \|\widetilde{u}_i\|_2^2 = k$. Therefore,

cos A is an orthonormal matrix whose column vector is orthonormal
multiply by an unit vector will not expand any energy
Thus l_2-norm square of $\widetilde{u}_i$ is up-bounded by 1

proof: $\sum_i^k \|\widetilde{u}_i\|_2^2 = \|\widetilde{U}\|_F^2 = \|A^T U\|_F^2 = \|A^T\|_F^2 = \|A\|_F^2 = \sum_{i=1}^k \|A_{:i}\|_2^2 = k$

column of A are all orthonormal

U is a square matrix with orthonormal columns

l_2-norm of column vector of A

$$\|A^T U S\|_F^2 = \sum_{i=1}^k \sigma_i^2 \|\widetilde{u}_i\|_2^2$$

$p > k$ ≤ 1

$$\leq \sum_{i=1}^k \sigma_i^2$$

Thus, the optimization is upper-bounded by the same quantity achieved by $A = U_k$, so that is the solution. Note that if for any $i, j \leq k$ $(i \neq j)$ then the solution need not be unique.

To recap. Any matrix $X \in \mathbb{R}^{p \times n}$ can be written as $X = USV^T$

$$U_k = \arg \max_{A \in \mathbb{R}^{p \times k}} \|A^T X\|_F^2$$

where U_k is the matrix whose columns are the respective top k left singular vectors of X.

$$\widetilde{\alpha}_i = U_k^T x_i \qquad x_i \in \mathbb{R}^p \qquad \widetilde{\alpha}_i \in \mathbb{R}^k$$

is the reduced dimension representation of x

Put another way, $\widetilde{\alpha}_i$ make up the coordinates to represent x in the space spanned by U_k.

Second Interpretation

representative data of
we want to find data live in P-D that live in k-D (k<p)

Represent the data in a low-rank space

This interpretation is directly connected to the previous one.

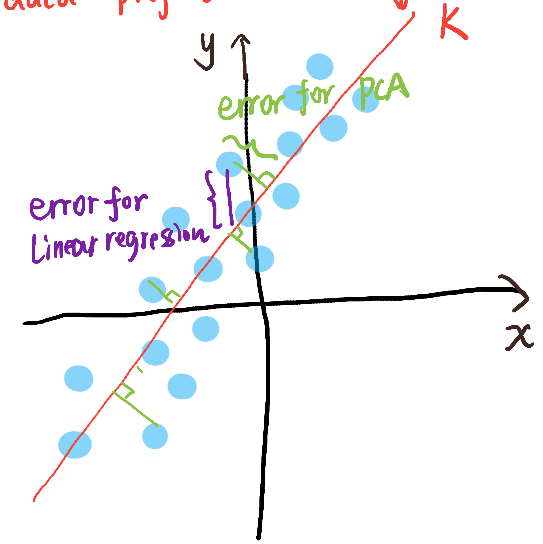
We denote $\widetilde{x}_i = U_k \widetilde{\alpha}_i = U_k U_k^T x_i$. Thus, $\widetilde{x}_i$ is just the projection of x onto the space spanned by U_k.

Illustration

We believe true signal live in low dimensional space

Noise are just added none-sense

PCA: 2-D data project to 1-D



minimizing prediction error

VS. linear regression: minimize $\frac{1}{n}\sum_{i=1}^{n}(y_i-\hat{y}_i)^2$

PCA: $\max_{A^TA=I}\sum_{i=1}^{n}\|AA^Tx_i\|_2^2 = \min_{K,\tilde{x}\in K}\sum_{i=1}^{n}\|\tilde{x}_i-x_i\|^2$

minimizing reconstruction error

We can also set this interpretation as an optimization

Find A that minimize reconstruction error

$k = \arg\min_{K,\, \tilde{x}_i \in K} \sum_{i=1}^{n} \|\tilde{x}_i - x_i\|_2^2$ equiv $U_k = \arg \min_{\substack{A \in \mathbb{R}^{p \times k} \\ A^T A = I}} \sum_{i=1}^{n} \underbrace{\|AA^T x_i - x_i\|_2^2}_{\text{reconstruction error}}$

So we wish to find the orthogonal k dimensional projection that minimizes the error between $\widetilde{x}_i$ and x_i.

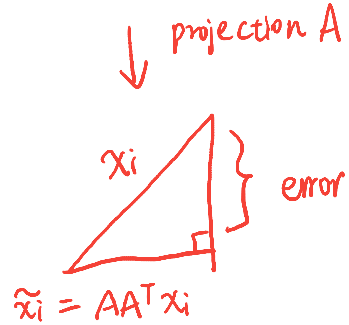
$\widetilde{x}_i = AA^T x_i$

Can verify that

we have seen $\|A\alpha\|_2^2 = \|\alpha\|_2^2$

勾股定理

$$\arg\min \|AA^T x_i - x_i\|_2^2 = \|x_i\|_2^2 - \|AA^T x_i\|_2^2 \quad (\alpha = A^T x_i)$$
$$= \|x_i\|_2^2 - \|A^T x_i\|_2^2$$



Thus, the minimization above is exactly equivalent to the maximization we looked at in the previous interpretation.

$\arg\max_A \|A^T x_i\|_2^2$

Third Interpretation

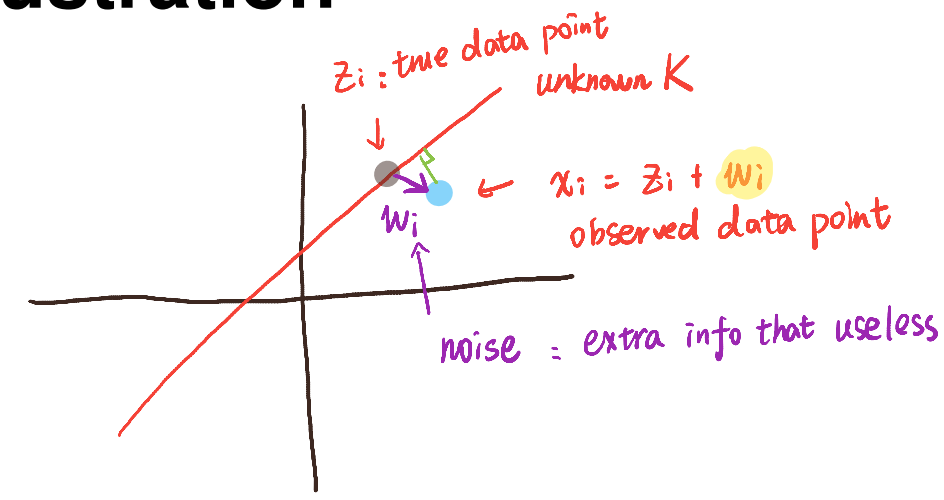
This interpretation is a data-denoising. It is also very connected to the previous interpretation

A Probalistic Model

Gaussian noise

Suppose that our data $x_i = z_i + w_i$ where $z_i \in K$ for some unknown $k-$dimensional space K and $w_i \sim N(0, \sigma^2 I)$.

Illustration



We can show the maximum likelihood of estimate z_i is just projection x_i onto K

But we need find K

We wish to reduce the noise and uncover the "latent" space K. We can frame this as an optimization.

The maximum likelihood estimate of $z_1, \dots, z_n$

$$K, z_i = \arg \min_{K, z_1, z_2, \dots, z_n \in K} \sum_{i=1}^{n} \|z_i - x_i\|_2^2$$

this optimization is equivalent to $\arg\min \sum_{i=1}^{n} \|AA^T x_i - x_i\|_2^2$ $\arg\min_{K, \tilde{x}_i \in K} \sum_{i=1}^{n} \|\tilde{x}_i - x_i\|^2$

Note that we are optimizing over K and z_i. Let $A \in \mathbb{R}^{p \times k}$ be a matrix whose columns are orthogonal and span the space K. Let $z_i = A\alpha_i$.

$$\alpha_i = \arg \min_{A \in \mathbb{R}^{p \times k}, \alpha_i \in \mathbb{R}^k} \sum_{i=1}^{n} \|A\alpha_i - x_i\|_2^2$$

One can verify that

$$\begin{aligned} \|A\alpha_i - x_i\|_2^2 &= \|A\alpha_i - AA^T x_i - (I - AA^T)x_i\|_2^2 \\ &= \|A\alpha_i - AA^T x_i\|_2^2 + \|(I - AA^T)x_i\|_2^2 \end{aligned}$$

$A\alpha_i - AA^T x_i = 0$

$\|(I - AA^T)x_i\|_2^2$ independent of α

The optimization then becomes

$$A, \alpha_i = \arg \min_{A \in \mathbb{R}^{p \times k}, \alpha_i \in \mathbb{R}^k} \sum_{i=1}^{n} \|A\alpha_i - AA^T x_i\|_2^2 + \|(I - AA^T)x_i\|_2^2$$

But the second term is independent of α. For a fixed A, the optimal choice of $\alpha_i = A^T x_i$ can be found in closed form using linear regression/orthogonality/etc... $\|A\alpha_i - AA^T x_i\|_2^2 = \|\alpha_i - A^T x_i\|_2^2 = \|\alpha_i - \alpha_i\|_2^2 = 0$

Thus, the first term is zero, so the optimization becomes

$$U_k = \arg \min_{A \in \mathbb{R}^{p \times k}} \sum_{i=1}^{n} \|(I - AA^T)x_i\|_2^2$$

This optimization yields exactly the same optimization as the second interpretation.

$$U_k = \arg \min_{A \in \mathbb{R}^{p \times k} \mid A^T A = I} \sum_{i=1}^{n} \|AA^T x_i - x_i\|_2^2$$

Recap second and third interpretations: If we wish to denoise data by projecting onto a low-dimensional space ~~K~~ then we optimize over projection P_K to minimize reconstruction error

$$P_K = \arg\min_{P_K} \sum_{i=1}^{n} \|P_K x_i - x_i\|_2^2$$

reconstruction error (l_2-norm square)

The optimal choice is $P_K = U_k U_k^T$.

Another thing need to be optimized is the size of k

Method ①

Plot singular value of data X, 肉眼看 eyebowl where the point to pick

Note: we can't use cross-validation to pick k with smallest reconstruction error
coz as k ↑ construction error will always ↓

② we have intuition on what level noise is on, then any singular value ≤ threshold might attribute to noise, so just set those singular value to 0

Fourth interpretation

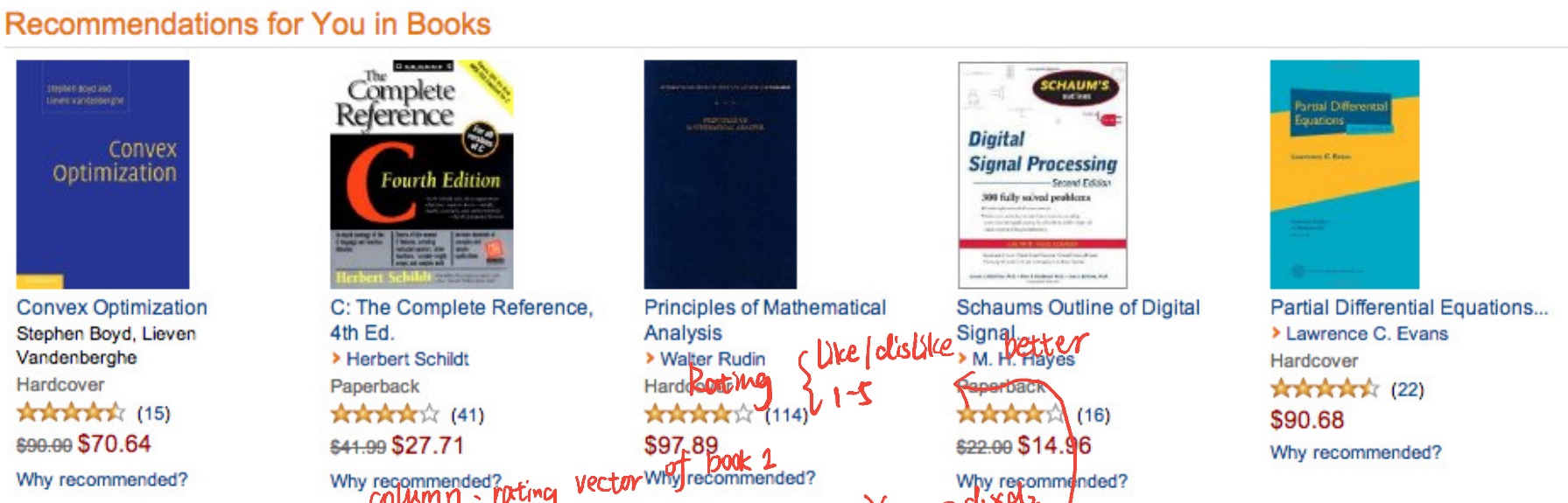
We think true data are low-rank matrix
other things are just noise

Low-rank matrix interpretation.

This is maybe best motivated by an example: recommendation systems.

2014 Netflix release a dataset for public competition
The winner uses low-rank matrix interpretation as workhorse

Recommendations



association

column = rating vector of book 1

Rating Matrix $X \in R^{d_1 \times d_2}$

Objects: Movies, Users, Books, ...

Users	book 1	book 2								
	*	*	*	*	*	*	4	*	*	...
	3	2	*	1	*	3	*	*	3	...
	*	*	4	*	*	3	*	3	4	...
	*	3	*	3	5	3	*	*	1	...
										⋱

row = rating vector of user i

Rating { Like / dislike 1-5 better

missing info but might redundant

Predict them

① less robust method. clustering movie average similar movies to impute

We might expect that certain books are similar as are certain users. Using that information we can make recommendations.

We can let our recommendation matrix X be such that $X_{(ij)}$ is user i's (probably un-observed) rating for item j.

How can we model this?

One option for modeling is using low-rank latent spaces.

Imagine for item j we have features $m_j \in \mathbb{R}^k$.

For instance, for books features could be: `fiction`, `fantasy`, `math`, `physics`, etc...

Then for user i we might have coefficient vector a_i such that $a_i^T m_j \approx X_{(ij)}$. This is exactly the **linear model** for linear regression!

Each entry in a_i represents how much a user likes a certain feature of the object. For example with books the features listed above. So, a_i is like the user's feature vector.

So, if the m_j were known, then we could just use linear regression to estimate a_i for each user based on the movies they have observed.

Recommendation is just prediction $\hat{x}_{(i,\ell)} = m_\ell^T \hat{a}_i$

Then we can just recommend which movie they might like by estimating the user i's rating for an unobserved movie ℓ by computing $m_\ell^T a_i$, and finding the largest predicted ratings.

What if we do not have features available?

We can just learn them. That is we can learn the latent k dimensional subspace of features.

More precisely, suppose we have d_1 users and d_2 movies. we can let $A \in \mathbb{R}^{d_1 \times k}$ be the matrix such that the rows are the user coefficients and $M \in \mathbb{R}^{d_2 \times k}$ be the matrix such that the rows are the object (books, movies) features.

Then, the rating matrix $X \approx AM^T$. Of course, in reality there is noise and things do not perfectly fit the model. Also, for recommendations we have missing entries.

Thus, suppose we have a model $X = AM^T + W$ where W is some noise term with bounded Frobenius norm. often take W as Gaussian

Assume we observed all the entries (we can assume some entries are missing, in this case, W is Gaussian Matrix with deletions. people have shown this also works)

Then we can consider the optimization

$$\arg \min_{A \in \mathbb{R}^{d_1 \times k}, M \in \mathbb{R}^{d_1 \times k}} \underbrace{\|AM^T - X\|_F^2}$$

Frobenius Norm of least square

Ambiguity: $(\frac{A}{2})(2M)^T$ for identibility

Note that AM^T is always a rank k matrix. Therefore, we know we can write it as AM^T where $A \in \mathbb{R}^{d_1 \times k}$ and $A^T A = I$.

If A has orthogonal columns

Thus, the solution is exactly as the previous interpretation!

$\arg \min_{A^TA=I \mid A \in R^{p \times k}; \alpha_i \in R^k} \sum_{i=1}^{n} \|A\alpha_i - x_i\|_2^2$

3rd
- $A\alpha_i$: column of original data X
- x_i: rating column for each movie i

4th
- m_i: feature vector of movie i
- $A m_i$: feature vector of all users

Set $A = U_k$ and $M = V_k S_k$. The role of α_i from the previous interpretation is played by m_i in this interpretation.

In General

Suppose we have a matrix $X \in \mathbb{R}^{d_1 \times d_2}$. For example user movie ratings.

We wish to find a low-rank matrix approximation of X

$$\widehat{X}_k = \arg \min_{Y | \text{rank}(Y) \leq k} \| Y - X \|_F^2$$

$Y = AM^T$

$\begin{cases} A = U_k \\ M = V_k S_k \end{cases}$

Then, $\widehat{X}_k = U_k S_k V_k^T$

Netflix: This optimization x scaling (70%) to get 10% Recommendation Improvement

Fifth interpretation

Notice that we have not really talked about probability or statistics in any of the interpretations.

vs. Numerical Interpretation (1~4)

Analogous to Factor Analysis

We now introduced the **Gaussian random vector** interpretation, which looks at the covariance matrix of that data.

Often before PCA: remove mean of column

To that end, suppose we have $x \sim N(0, \Sigma)$. Suppose I wish to find the unit vector v that maximizes

this direction of variance

$$\arg \max_{v \| \|v\|=1} \mathbb{E}(v^T x)^2$$

That is, v is the direction such that the variance along its direction is maximized.

X is return v: How much I elated to every stock Elication can be negative

Risk Analysis in Economics: find direction that maximize variance (vulnetility) related to Risk in Stocks)

Prior-Fualial

Illustration

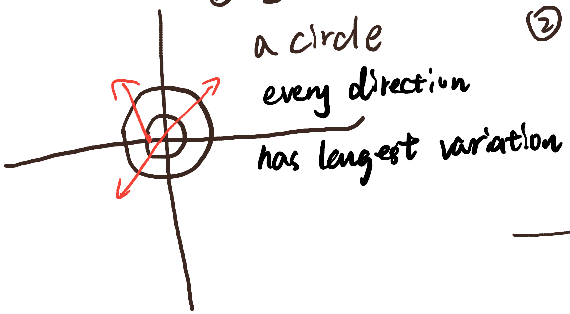
Gaussian Matrix with i.i.d

pdf $\frac{1}{2\pi \det(\Sigma)^{\frac{1}{2}}} \exp\left(-\frac{x^T \Sigma^{-1} x}{2}\right)$

① $\Sigma = I$

a circle

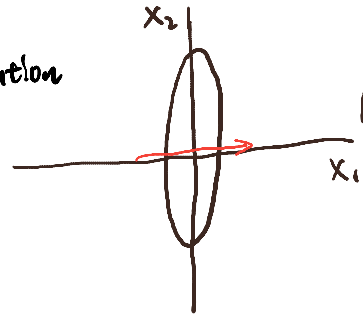
every direction has longest variation



② general e.g. $\Sigma = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$ Ellipse

Now variance change fatest in direction x_1

Ellipse is align with axis

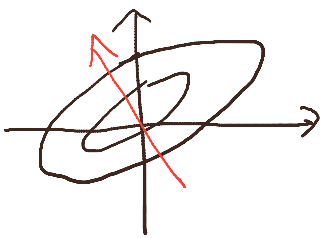


③ Ellipse don't align with axis

Rotate data $\tilde{X} = UX$ $\Sigma = \tilde{X}\tilde{X}^T$

$E(\tilde{X}\tilde{X}^T) = U\Sigma' U^T$ → a diagonal matrix

Eigenty decomposition of Σ is $U\Sigma' U^T$



$E(v^T x)^2 = E[(v^T x)(x^T v)] = v^T E(x x^T v)$

scalar

$= v^T E[x x^T] v$

v is a constant, a linear combination (summation $\sum_{i,j}$)

$$\arg \max_{v \mid \|v\|=1} \mathbb{E}(v^T x)^2 = \arg \max_{v \mid \|v\|_2=1} v^T \Sigma v$$

We recall that Σ is a covariance matrix. Therefore, it is symmetric and positive semi-definite. $\Rightarrow$ all eigen values of $\Sigma \geq 0$

always

条件：Σ is symmetric

Thus, it has an eigen-decomposition $\Sigma = UDU^T$ where D is diagonal with positive entries, and U is a square orthogonal matrix, so $U^T U = UU^T = I$.

For a symmetric and 半正定 matrix SVD = ED

Note that this is exactly the SVD of Σ as well. We will show that the optimal choice of v is u_1 column of U

Case ②

$$\arg\max_v v^T \Sigma v = v^T \Big(\sum_{i=1}^{p} D_{(ii)} u_i u_i^T\Big) v$$

$$= \sum_{i=1}^{p} D_{(ii)} (v^T u_i)^2$$

Both, v_i and u_i are unit vectors, so $0 \leq (v^T u_i)^2 \leq 1$. Also $\sum_i (v^T u_i)^2 = 1$

The optimal choice of v is therefore $v = u_1$, the first eigenvector of Σ!

Put more weight onto component with longest $D_{(ii)}$ (eigen value)

P_k is projection to k-D space with highest Gaussian variance

As before we can generalize to k-dimensional projections.

$$K = \arg\max_K \mathbb{E}\|P_K x\|_2^2 = \arg\max_K \operatorname{tr}(P_K \Sigma P_K^T) \quad \downarrow \; P_k = AA^T$$

$$\|P_k x\|^2 = x^T P_k^T P_k x = \operatorname{tr}(P_k^T P_k x x^T)$$

$$= \arg\max_{A \in \mathbb{R}^{p\times k} | A^T A = I} \operatorname{tr}(AA^T \Sigma A^T A)$$

$$= \operatorname{tr}(P_k x x^T P_k^T)$$

$$= \arg\max_{A \in \mathbb{R}^{p\times k} | A^T A = I} \operatorname{tr}(AA^T U D^{1/2} D^{1/2} U^T A^T A)$$

$$E[\operatorname{tr}(P_k x x^T P_k^T)]$$

linear

$$= \arg\max_{A \in \mathbb{R}^{p\times k} | A^T A = I} \operatorname{tr}(\|AA^T U D^{1/2}\|_F^2)$$

$\|AB\|_F^2 = \|B\|_F^2$ if A is orthogonal matrix

$$= \arg\max_{A \in \mathbb{R}^{p\times k} | A^T A = I} \operatorname{tr}(\|A^T U D^{1/2}\|_F^2)$$

This is exactly the same optimization as in Interpretation 1! So, $A = U_k$, the top k eigenvectors of Σ!

Suppose we do not know what Σ is but we still wish to estimate v?
Suppose we have samples $x_i \sim N(0, \Sigma)$. Just use empirical averages instead! Rather than $\mathbb{E}(v^T x_i)^2$

$$\arg\max_v \underbrace{\frac{1}{n}\sum_{i=1}^{n}(v^T x_i)^2}_{\text{empirical variance}}$$

But this is exactly the first interpretation of PCA!

Actually, we can rewrite it as

$$\frac{1}{n}\sum_{i=1}^{n} v^T x_i x_i^T v = v^T \widehat{\Sigma} v$$

where $\widehat{\Sigma}$ is the empirical covariance matrix of the data!

often big data. if we just want to know left singular vectors, we only need to compute $\sum$ covariance matrix

More generally can consider K-dimensional orthogonal projections

$$K = \arg\max_{K} \frac{1}{n}\sum_{i=1}^{n} \|P_K x_i\|_2^2$$

Recap: PCA for data generated according to a Gaussian model simply finds the top k eigenvectors of the sample covariance matrix.

That's one motivation. Of course, the data need not be Gaussian.

So another way to solve PCA is to compute the sample covariance matrix of the data (with or without re-centering) and compute the top eigenvectors.

Constructing the SVD from eigenvectors

The above shows how we can compute the PCA using eigenvectors from the sample covariance matrix of the data.

We can use that motivation to show how one can construct the SVD.

Given a matrix $X \in \mathbb{R}^{p \times n}$ let $Y = XX^T$. Clearly, Y is both positive semi-definite and symmetric. Thus, we can write $Y = UDU^T$ for its eigen-decomposition.

We will show that $X = UD^{1/2}V^T$ for an appropriate orthogonal matrix V.

We deal with the case $p \leq n$ and X has rank p. The general case is a simple extension using projections.

$Y = XX^T \implies D^{-1/2}U^T XX^T UD^{-1/2} = I$. Possible since X has rank p, so Y has rank p as well, so $D_{(ii)} > 0$ for all i.

Thus, $X^T UD^{-1/2} = (D^{-1/2}U^T X)^{-1} = (D^{-1/2}U^T X)^T$

Therefore, $X^T UD^{1/2} = V$ an orthogonal matrix.

Multiplying yields the desired result.

Conclusion

PCA is a powerful tool for unsupervised learning.

It can be used for visualization, dimensionality reduction, data-denoising, and low-rank matrix approximations.

Given data $X \in \mathbb{R}^{p \times n}$ where we have n data points in p dimensions the top k principal components of X are denoted as U_k which are exactly the top k left singular vectors of X.

If we do dimensionality reduction and project $U_k^T X$, then we are left with

$$\begin{aligned} U_k^T X &= U_k^T U S V^T \\ &= S_k V_k^T \end{aligned}$$

So the lower-dimensional coordinates $\widehat{\alpha}_i$ of example x_i are exactly $[\widehat{\alpha}_i]_{(j)} = \sigma_j [V_k]_{(ij)}$, i.e. σ_j times the i^{th} component of the j^{th} right singular vector.