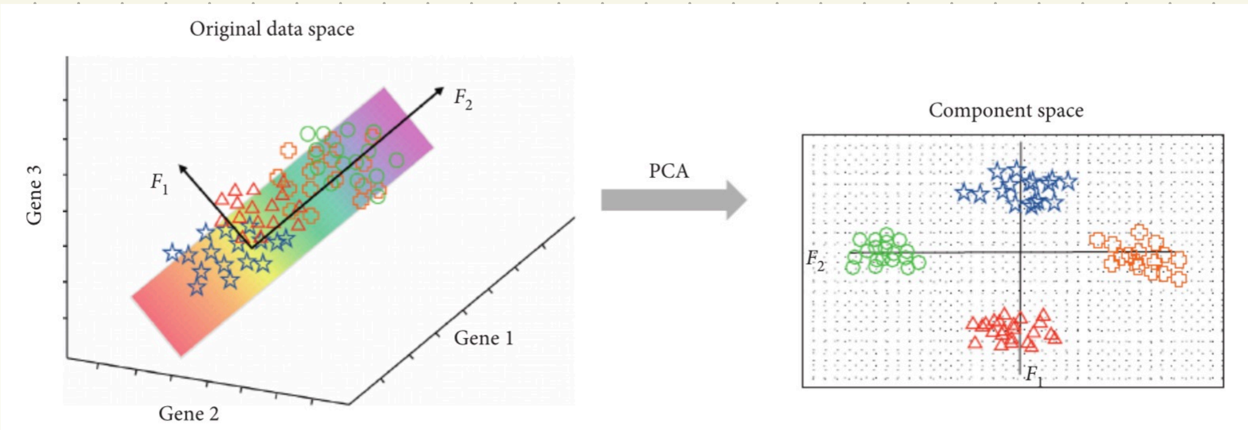
Principal component analysis (PCA)

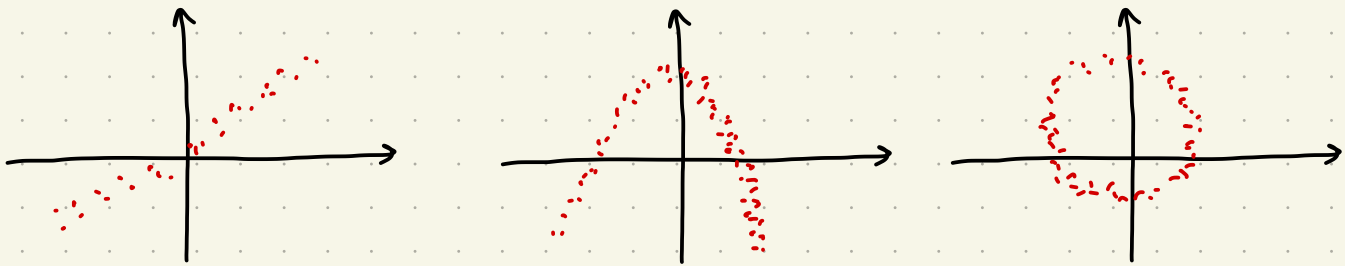


https://onlinelibrary.wiley.com/doi/10.1155/2020/8848324

Dimension reduction

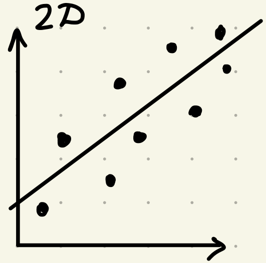
<u>Assumption</u>: The data lies along a low-dimensional manifold.

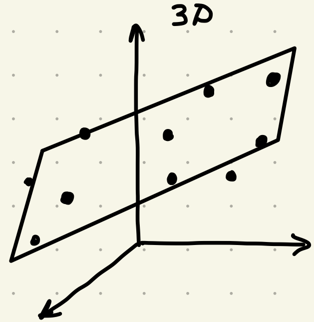
<u>Example</u>: data in 2D, but actually lie on a curve



Principal component analysis (PCA)

Data lies on an r-dimensional affine subspace ("plane")

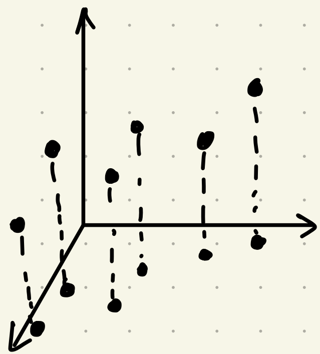




Reduce the dimensionality while preserving important information.

How about removing one of the columns?

$$X = \begin{pmatrix} | & | & \cancel{|} \\ f_1 & f_2 & \cancel{f_3} \\ | & | & \cancel{|} \end{pmatrix}$$



1st view of PCA: maximum variance

Suppose we are given $x_1 \dots x_n \in \mathbb{R}^p$ points

data matrix $X = \begin{pmatrix} -x_1^T- \\ \vdots \\ -x_n^T- \end{pmatrix}$

We assume that X is column-centered, thus,

$$\bar{x} = \frac{X^T \cdot 1}{n} = \frac{\sum_{i=1}^{n} x_i}{n} = 0$$

$$X \longrightarrow X_c = C\, X = \left(I - \frac{11^T}{n}\right) X, \text{ then}$$

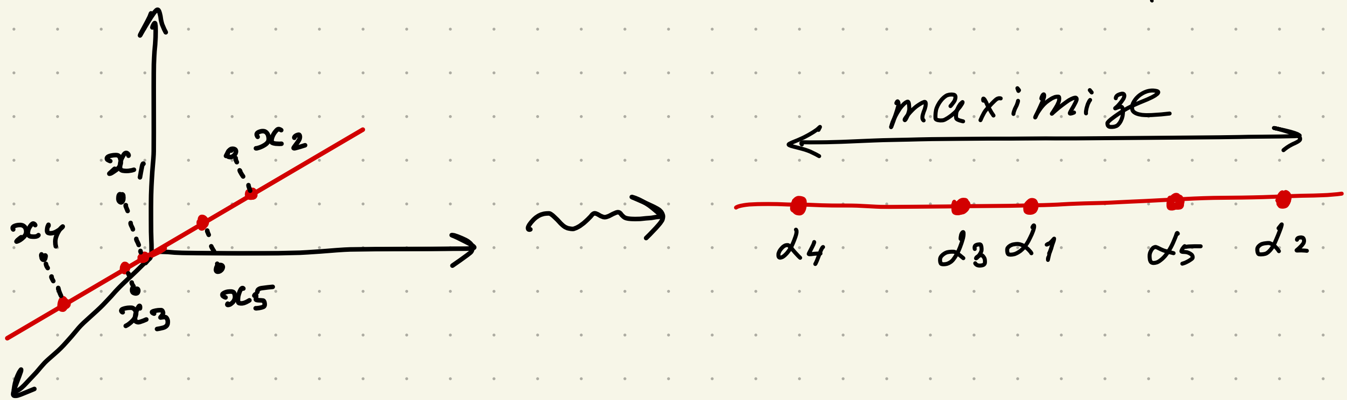
$$\overline{x_c} = (X^c)^T \frac{1}{n} = X^T \cdot \left(I - \frac{11^T}{n}\right) \cdot \frac{1}{n} = X^T \left(\frac{1}{n} - \frac{1}{n}\left(\frac{\overset{1}{\overset{\shortparallel}{1^T 1}}}{n}\right)\right) = 0$$

The covariance matrix is

$$S = \frac{1}{n-1} \sum_{i=1}^{n} x_i x_i^T = \frac{1}{n-1} X^T X.$$

PCA: the first principal component (PC_1)

? Project the data on a line and maximize the variance of the projected points.



The line is defined by $v \in \mathbb{R}^p$, $\|v\| = 1$

The projection on the line is

$$\hat{x}_i = (x_i^T v) \cdot v. = \alpha_i \cdot v$$

$P_v = v v^T$, then $\hat{x}_i = P_v x_i = v \cdot (v^T x_i)$

Note that $\bar{\alpha} = \frac{1}{n} \sum_{i=1}^{n} \alpha_i = 0$

$$\frac{1}{n} \sum_{i=1}^{n} (v^T x_i) = v^T \left(\frac{1}{n} \sum_{i=1}^{n} x_i \right) = 0$$

Let's compute the variance of $z = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} = X \cdot v$

$$\operatorname{var}(z) = \frac{1}{n-1} \sum_{i=1}^{n} (x_i^T v)^2 = v^T \left(\frac{1}{n-1} \sum_{i=1}^{n} x_i x_i^T \right) v = v^T S v$$

The goal is to solve

$$\underset{v \in \mathbb{R}^p}{\text{maximize}} \; v^T S v \quad \text{subject to} \quad \|v\| = 1$$

We will use Lagrange multipliers.

Detour: method of Lagrange multipliers

Consider $x \in \mathbb{R}^p$ and two functions
$f: \mathbb{R}^p \to \mathbb{R}$ and $g: \mathbb{R}^p \to \mathbb{R}$

We want to
maximize $f(x)$ subject to $g(x) = 0$

① Form Lagrangian $\mathcal{L}(x, \lambda) = f(x) + \lambda g(x)$

Lagrange multiplie

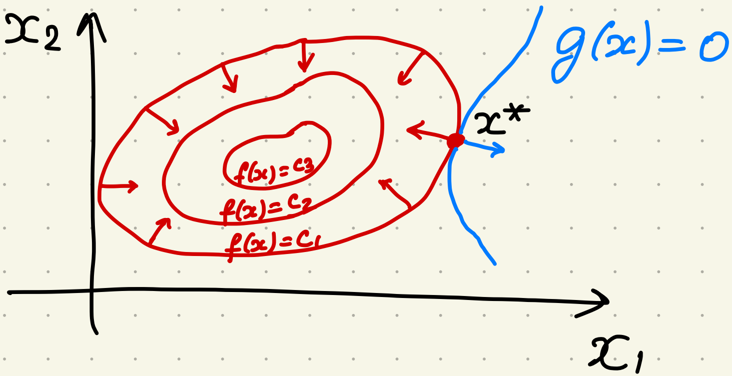
② Find critical points of $\mathcal{L}$: $\nabla \mathcal{L}(x, \lambda) = 0$

③ Consider all critical points (x^*, λ^*) of $\mathcal{L}$
find x^* that results in the smallest $f(x^*)$

- $\nabla \mathcal{L}(x;\lambda) = \begin{pmatrix} \nabla f(x) + \lambda \nabla g(x) \\ g(x) \end{pmatrix} = 0 \Rightarrow \begin{cases} \nabla f(x) = 0 \\ g(x) = 0 \end{cases}$

- contour for $f(x)$ and $g(x)$ are tangent at x^*
$\nabla f(x)$ and $\nabla g(x)$ at x^* are parallel

- there is λ^* such that $\nabla f(x^*) = -\lambda^* \nabla g(x^*)$



For multiple constrains

minimize $f(x)$ subject to $\begin{cases} g_1(x)=0 \\ \dots \\ g_k(x)=0 \end{cases}$

1. Form Lagrangian

$$\mathcal{L}(x, \lambda_1 \dots \lambda_k) = f(x) + \lambda_1 g_1(x) + \dots + \lambda_k g_k(x)$$

2. Solve $\nabla \mathcal{L}(x, \lambda_1 \dots \lambda_k) = 0$, find critical points $x^*, \lambda_1^* \dots \lambda_k^*$

3. Use x^* with the minimum value of $f(x^*)$.

$$\underset{v \in \mathbb{R}^p}{\text{maximize}} \; v^T S v \quad \text{subject to} \quad \|v\|^2 = 1$$

- v is an eigenvector of S

$$\mathcal{L}(v, \lambda) = v^T S v - \lambda(\|v\|^2 - 1) = v^T S v - \lambda(v^T v - 1)$$

$$\nabla_v = 2Sv - 2\lambda v \Longrightarrow Sv = \lambda v$$

- v corresponds to the maximum e. value

$$v^T S v = \lambda \|v\|^2 = \lambda \longrightarrow \max$$

We showed that

$$v = v_1 \quad \text{and} \quad Var(z) = v^T S v = \lambda_1$$

Terminology:

- v_1 is the loading vector / direction of PC_1
- $z_1 = X v_1$ is the score vector (or just PC_1)
- λ_1 is the variance explained by PC_1.

Comment: the population version

given random $x = \begin{pmatrix} x_1 \\ \vdots \\ x_p \end{pmatrix} \in \mathbb{R}^p$ a linear combination $z = x^T v$ with maximum variance and $\|v\|^2 = 1$

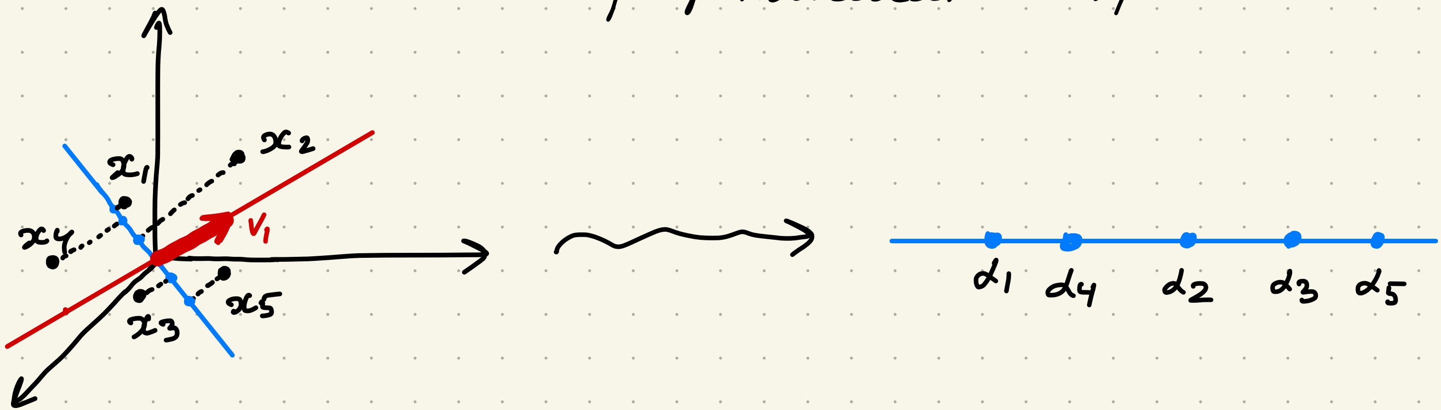
$$var(z) = var(x^T v) = v^T \Sigma v \quad \text{where}$$

$cov(x) = \Sigma$, then v is the first e.vector.

PCA: the second component (PC_2)

? Project the data on a line and maximize the variance of the projected points.

The line shoule be perpendicular to V_1



New optimization goal:

maximize $V^T S V$ subject to $\begin{cases} \|V\| = 1 \\ V^T V_1 = 0 \end{cases}$

- v is (again) an eigenvector of S

$$\mathcal{L}(v, \lambda, \mu) = v^T S v - \lambda(v^T v - 1) - \mu v^T v_1$$

$$\nabla_v \mathcal{L}(v, \lambda, \mu) = 2Sv - 2\lambda v - \mu v_1 = 0 \mid \cdot v_1$$

$$2\underbrace{v_1^T S v}_{0} - 2\lambda \underbrace{v_1^T v}_{0} - \mu \underbrace{v_1^T v_1}_{1} = -\mu = 0$$

So, $Sv = \lambda v$ and v and λ are e.vec/e.val

- v corresponds to the second largest e.val.

$v^T S v = \lambda \|v\|^2 = \lambda \to \max$ and $v^T v_1 = 0$

Both are satisfied by $v = v_2$.

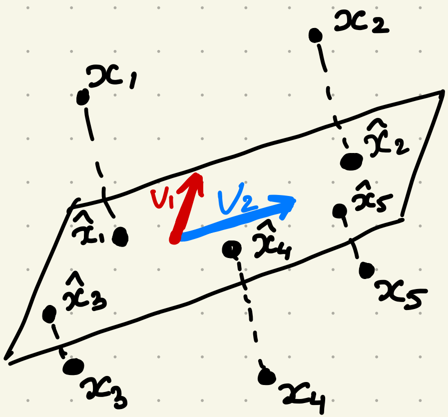
Terminology: v_2 is the direction of PC_2

$z_2 = X v_2$ is the score vector, λ_2 is variance explained.

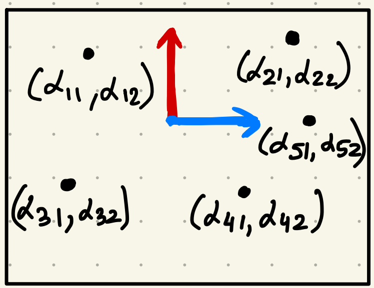
$$V_{(2)} = \begin{pmatrix} | & | \\ v_1 & v_2 \\ | & | \end{pmatrix} \in \mathbb{R}^{p \times 2}$$

$$Z_{(2)} = \begin{pmatrix} | & | \\ z_1 & z_2 \\ | & | \end{pmatrix} = \begin{pmatrix} d_{11} & d_{12} \\ \cdots & \cdots \\ d_{n1} & d_{n2} \end{pmatrix} = X V_{(2)} \in \mathbb{R}^{n \times 2}$$

$$\hat{X}_{(2)} = \begin{pmatrix} - \hat{x}_1^T - \\ \cdots \\ - \hat{x}_n^T - \end{pmatrix} = \underset{\displaystyle (P_{V_{(2)}} X^T)^T}{\underset{\shortparallel}{X V_{(2)} V_{(2)}^T}} \in \mathbb{R}^{n \times p}$$



p-dimensional



2-dimensional

PCA: the r-th component (PC_r)

maximize $v^T S v$ subject to $\begin{cases} \|v\| = 1, \\ v^T v_1 = \dots = v^T v_{r-1} = 0 \end{cases}$

The resulting **directions** $v_1 \dots v_r$ are the eigenvectors of S corresponding to the largest r eigenvalues. The **scores** are $z_i = X v_i$ and the **variances explained** are $\lambda_i = Var(z_i)$

$V_{(r)} = \begin{pmatrix} | & & | \\ v_1 & \dots & v_r \\ | & & | \end{pmatrix} \in \mathbb{R}^{n \times r}$ the basis

$Z_{(r)} = \begin{pmatrix} | & & | \\ z_1 & \dots & z_r \\ | & & | \end{pmatrix} = \begin{pmatrix} \alpha_{11} & \dots & \alpha_{1r} \\ \vdots & & \vdots \\ \alpha_{n1} & \dots & \alpha_{nr} \end{pmatrix} = X V_{(r)} \in \mathbb{R}^{n \times r}$ projections in $\mathbb{R}^r$

$\hat{X}_{(r)} = \begin{pmatrix} - \hat{x}_1^T - \\ \vdots \\ - \hat{x}_r^T - \end{pmatrix} = X V_{(r)} V_{(r)}^T \in \mathbb{R}^{n \times p}$ projections in $\mathbb{R}^p$

PCA via SVD

X is centered, $X = UDV^T = \underset{n\times p}{\boxed{U}}\ \underset{p\times p}{\boxed{D}}\ \underset{p\times p}{\boxed{V^T}}$

$$S = \frac{X^TX}{n-1} = V\frac{D^2}{n-1}V^T$$

- $V_{(r)}$ are the top-r right singular vectors
- the scores $Z_{(r)} = XV_{(r)} = U_{(r)}D_{(r)} = \begin{pmatrix} | & & | \\ u_1 & \cdots & u_r \\ | & & | \end{pmatrix}\begin{pmatrix} d_1 & & \\ & \ddots & \\ & & d_r \end{pmatrix}$

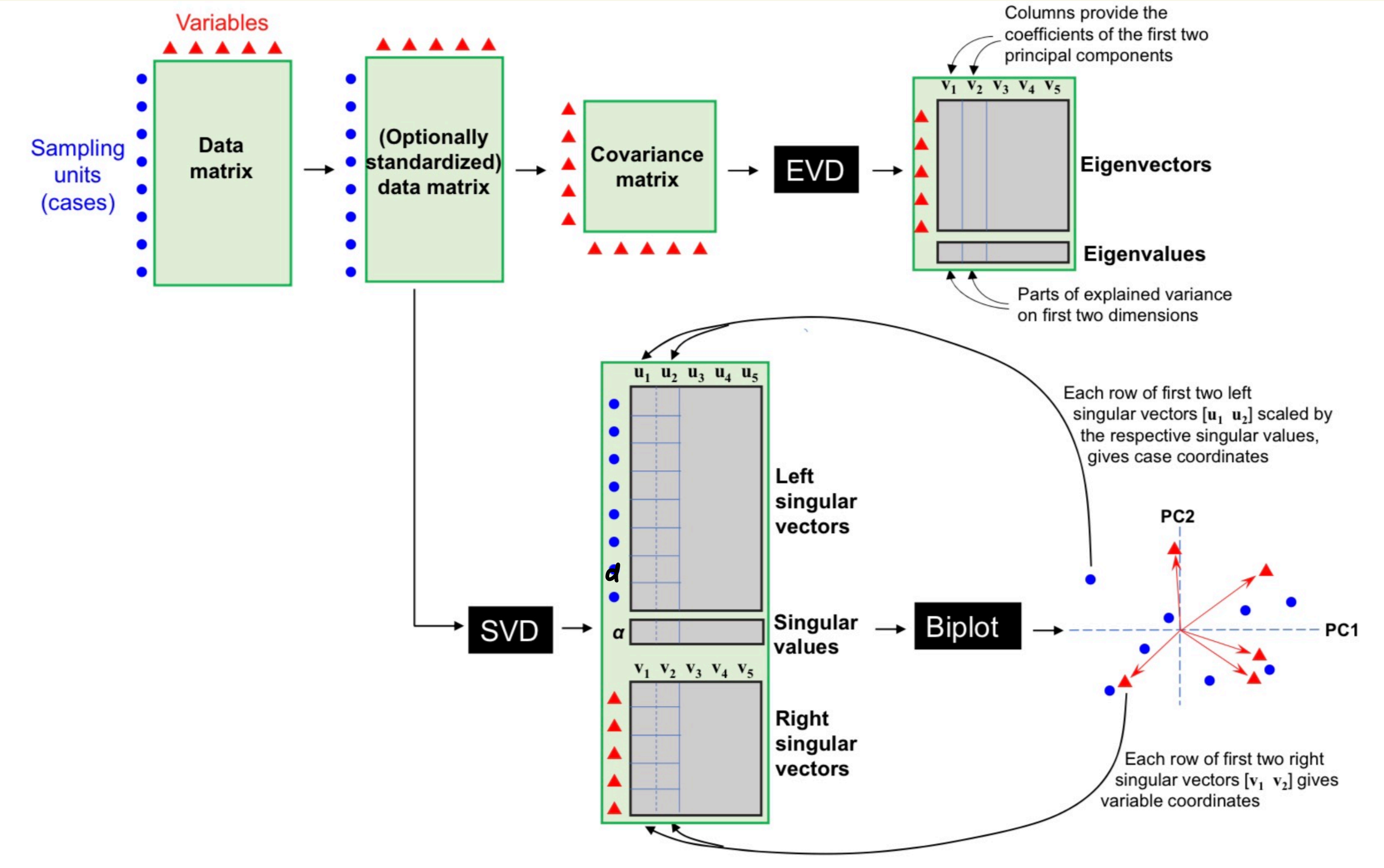
$$V^TV_{(r)} = \left(V_{(r)}\ (V_{(r)})_\perp\right)^T V_{(r)} = \begin{pmatrix} V_{(r)}^T V_{(r)} \\ (V_{(r)})_\perp^T V_{(r)} \end{pmatrix} = \begin{pmatrix} I \\ 0 \end{pmatrix} \in \mathbb{R}^{p\times r}$$

$$XV_{(r)} = UDV^TV_{(r)} = UD\begin{pmatrix} I \\ 0 \end{pmatrix} = U\begin{pmatrix} d_1 & & \\ & \ddots & \\ & & d_n \end{pmatrix}\begin{pmatrix} I \\ 0 \end{pmatrix} = U\begin{pmatrix} D_{(r)} \\ 0 \end{pmatrix}$$

$$= \left(\underbrace{U_{(r)}}_{r}\ \underbrace{U_*}_{p-r}\right)\begin{pmatrix} D_{(r)} \\ 0 \end{pmatrix} = U_{(r)}D_{(r)}$$

- the projections $\hat{X}_{(r)} = U_{(r)}D_{(r)}V_{(r)}^T = SVD_r(X)$

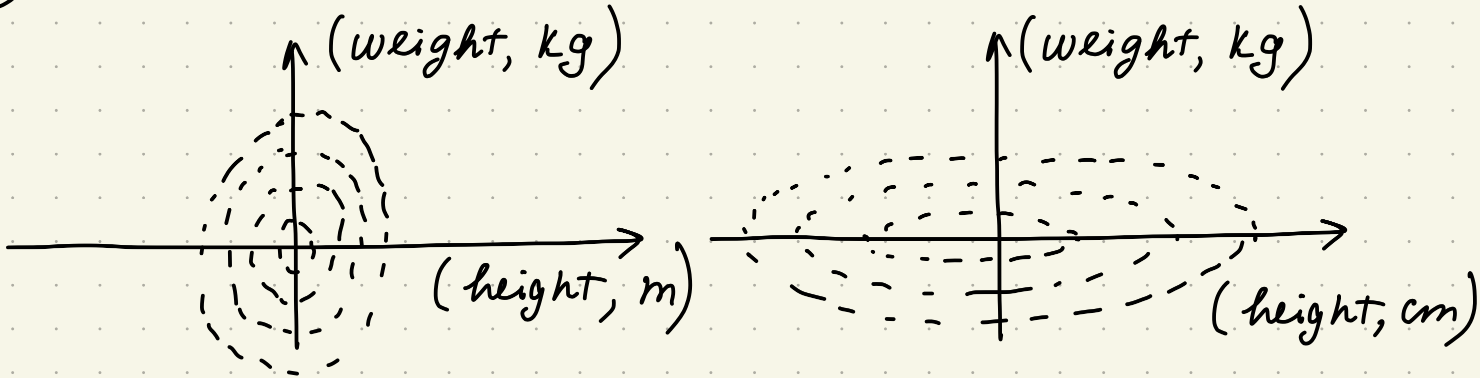
$$\hat{X}_{(r)} = d_1u_1v_1^T + \ldots + d_ru_rv_r^T + \cancel{d_{r+1}u_{r+1}v_{r+1}^T + \ldots + d_pu_pv_p^T} \simeq X$$



Principal component analysis by *M. Greenacre, P. Groenen, T. Hastie, A. d'Enza, A. Markos, and E. Tuzhilina*, Published in *Nature Reviews Methods Primers*, 2022

Practical aspects

(1) PCA is sensitive to the units



Sometimes you need to standardize columns of the data

$$X = \begin{pmatrix} | & & | \\ f_1 & \dots & f_p \\ | & & | \end{pmatrix} \rightsquigarrow \tilde{X} = \begin{pmatrix} | & & | \\ \frac{f_1 - \overline{f_1}}{sd(f_1)} & \dots & \frac{f_p - \overline{f_p}}{sd(f_p)} \\ | & & | \end{pmatrix}$$

② How to pick r?

- For visualization, use $r = 2$ or 3
- For dimention reduction, compute the proportion of variance explained by each PC.

$$\underset{n\times p}{\boxed{X}} \longrightarrow \underset{p\times p}{\boxed{S}} \longrightarrow$$

$$VE(PC_1) = \lambda_1$$

$$\cdots$$

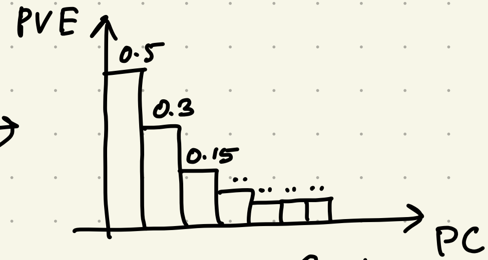
$$VE(PC_p) = \lambda_p$$

$$\text{Total} = \lambda_1 + \ldots + \lambda_p$$

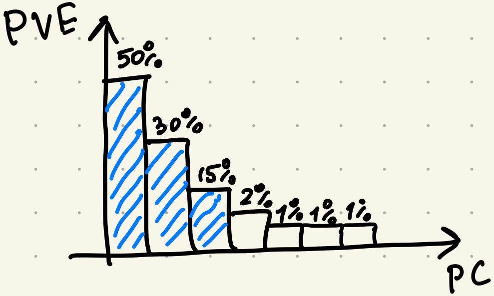
$$PVE(PC_1) = \frac{\lambda_1}{\lambda_1 + \ldots + \lambda_p}$$

$$\cdots$$

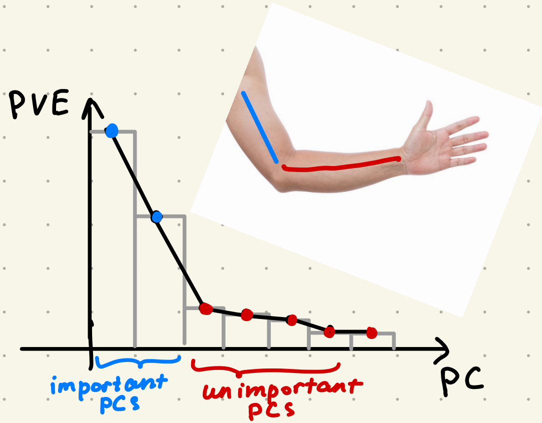
$$PVE(PC_p) = \frac{\lambda_p}{\lambda_1 + \ldots + \lambda_p} \longrightarrow$$



Scree plot

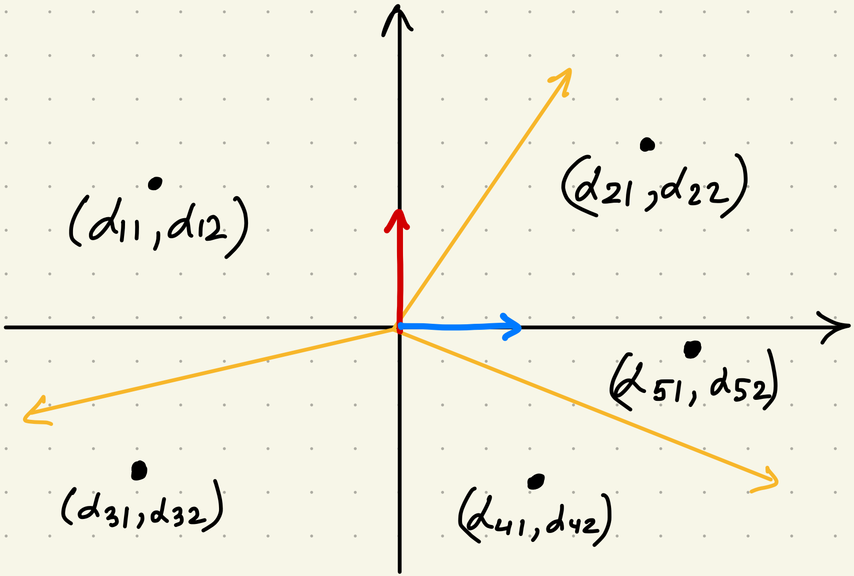


- Select r such that $PC_1 \dots PC_r$ explain, say, $\geq 90\%$ of the total variance.

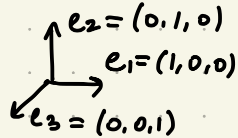


- Select r such that the remaining $n-k$ PVE values are small (elbow method)

② Visualization (2D): Biplot



In orange: the projections of feature axes on the (PC_1, PC_2) space



$$\begin{pmatrix} - e_1^T - \\ - e_2^T - \\ - e_3^T - \end{pmatrix} = I$$

projetions are

$$I \cdot V_{(2)} = \left(V_{(2)} \right)$$

$$= \left(\quad \right)$$

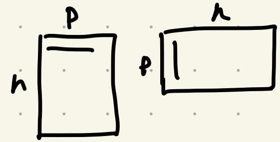
③ PCA complexity

Input: $X \in \mathbb{R}^{n \times p}$

Step 1: Column-centering: compute $\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$

$O(np)$ and $x_i \to x_i - \bar{x}$ for all $i = 1, \dots, n$

Step 2: Compute $S = \frac{1}{n-1} X X^T$



$O(np^2)$

Step 3 Compute eigendecomposition of $S \in \mathbb{R}^{p \times p}$

$O(p^3)$

Output: directions $v_1 \dots v_p$, variance $\lambda_1 \dots \lambda_p$

Total Compexity: $O(np^2 + p^3)$

Often, we need only r eigenvectors of $S \in \mathbb{R}^{p \times p}$

Power iteration

Find λ_1 of $S \in \mathbb{R}^{p \times p}$ assuming $\lambda_1 > \lambda_2 \geq \lambda_3 \ldots \geq \lambda_p \geq 0$

Input: $S \in \mathbb{R}^{p \times p}$ and arbitrary $v \in \mathbb{R}^p$

$$\left.\begin{array}{ll} \text{Step 1} & \tilde{v} = S v \\ \text{Step 2} & v = \dfrac{\tilde{v}}{\|\tilde{v}\|} \end{array}\right\} \text{ repeat until } v \text{ converges}$$

Output: v

Why it works?

Recall $S^k = U \begin{pmatrix} \lambda_1^k & & \\ & \ddots & \\ & & \lambda_p^k \end{pmatrix} U^T$. Assume $v = \sum_{i=1}^{p} \alpha_i v_i$

Then $S^k v = \sum_{i=1}^{k} \lambda_i^k \alpha_i v_i = \lambda_1^k \sum_{i=1}^{k} \left(\frac{\lambda_i}{\lambda_1}\right)^k \alpha_i v_i$

If $i = 1$ then $\frac{\lambda_i}{\lambda_1} = 1$, if $i > 1$ then $\frac{\lambda_i}{\lambda_1} < 1$ and $\left(\frac{\lambda_i}{\lambda_1}\right)^k \xrightarrow[k \to \infty]{} 0$

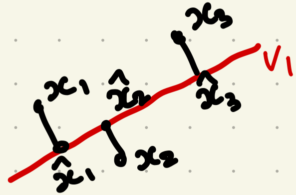
$S^k v \simeq \lambda_1^k \alpha_1 v_1$ and $\dfrac{S^k v}{\|S^k v\|} = \dfrac{\cancel{\lambda_1^k} \cancel{\alpha_1} v_1}{\cancel{\lambda_1^k} \cancel{\alpha_1}} = v_1$

Application to PCA:

1) Use power iteration to find $v_1 \Rightarrow \lambda_1 = v_1^T S v_1$

2) $\tilde{X} = X(I - v_1 v_1^T)$

$| \ \tilde{X} = X - \hat{X} = X - X v_1 v_1^T$



3) $\tilde{S} = \frac{1}{n-1} \tilde{X}^T \tilde{X} = S - \lambda_1 v_1 v_1^T$

$| \ \tilde{S} = (I - v_1 v_1^T) S (I - v_1 v_1^T) = S - v_1 v_1^T S - S v_1 v_1^T + v_1 v_1^T S v_1 v_1^T =$

$= S - \lambda_1 v_1 v_1^T - \lambda_1 v_1 v_1^T + \lambda_1 v_1 \cancel{v_1^T v_1} v_1^T = S - \lambda_1 v_1 v_1^T$

4) Use power iteration to find $\tilde{v}_1$ of $\tilde{S} \Rightarrow$

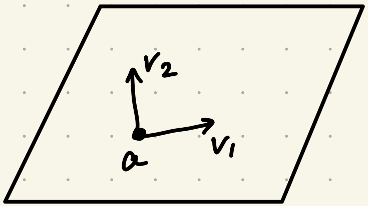
$v_2 = \tilde{v}_1$ and $\lambda_2 = v_2^T S v_2$

2nd view of PCA: minimum reconstruction error

Given n points $x_1 \dots x_n \in \mathbb{R}^p$ find the "best" approximation by an r-dimensional plane.

To construct a plane we need:

- a point on the plane $a \in \mathbb{R}^p$
- a basis $v_1 \dots v_r \in \mathbb{R}^p$, assume $V^T V = I$

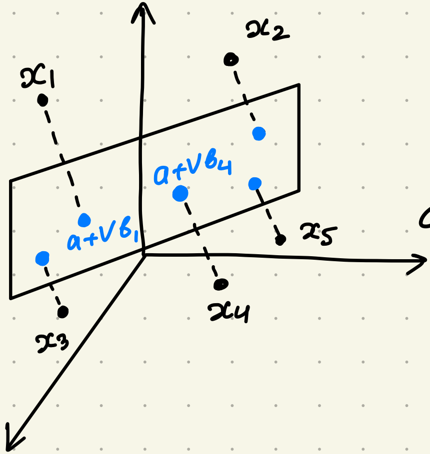


Any point on the plane is:

$a + \beta_1 v_1 + \dots + \beta_r v_r$ for some $\beta_1 \dots \beta_r$

Equivalently, $a + V\beta$, where $V = \begin{pmatrix} | & & | \\ v_1 & \dots & v_r \\ | & & | \end{pmatrix} \in \mathbb{R}^{p \times r}$

$$\beta = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_r \end{pmatrix} \in \mathbb{R}^r$$



The PCA goal

$$\underset{a,V,b_1,\dots,b_n}{\text{minimize}} \sum_{i=1}^{n} \|x_i - (a + V b_i)\|^2 = \ell(a, V, \{b_i\})$$

- If a and V are fixed

$$b_i = V^T(x_i - a)$$

| regression $b_i = (V^T V)^{-1} V^T (x_i - a)$

- $$\ell(a,V) = \sum_{i=1}^{n} \|(I - VV^T)(x_i - a)\|^2$$

minimized at $a = \bar{x} + VV^T c$ for any $c \in \mathbb{R}^p$

$$\|x_i - (a + VV^T(x_i - a))\|^2 = \|(I - VV^T)(x_i - a)\|^2 =$$

$$= (x_i - a)^T (I - VV^T)^{\cancel{2}} (x_i - a)$$

$$\nabla_a \ell(a, V) = 2 \sum_{i=1}^{n} (I - VV^T)(x_i - a) = 2n(I - VV^T)(\bar{x} - a) = 0$$

$\bar{x} - a = VV^T c$ for any $c \in \mathbb{R}^p$, e.g. $c = 0$.

- $\ell(V) = \sum_{i=1}^{n} \| (I - VV^T)(x_i - \bar{x}) \|^2$

  minimized at $V = V_{(r)} = (v_1, \dots, v_r)$ top r e.vecs of S.

  $$\ell(V) = \sum_{i=1}^{n} tr[(x_i - \bar{x})^T (I - VV^T)(x_i - \bar{x})] =$$

  $$= tr[(I - VV^T) \sum_{i=1}^{n} (x_i - \bar{x})(x_i - \bar{x})^T =$$

  $$= n\, tr[(I - VV^T) S] = -n\, tr(V^T S V) + \dots\dots$$

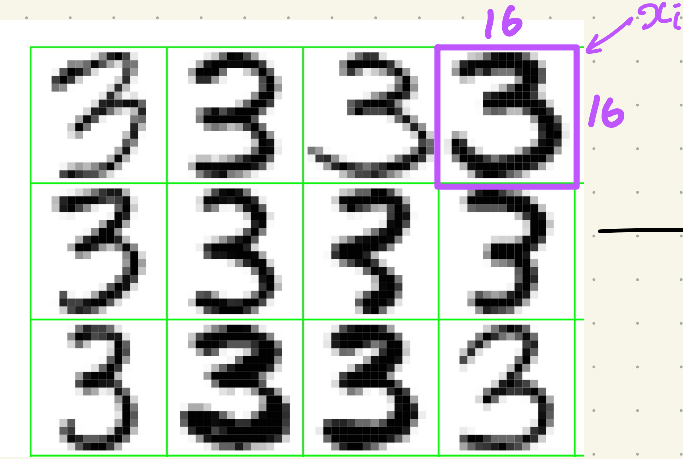
  $$\underset{V \in \mathbb{R}^{p \times r}}{\text{maximize}}\ tr(V^T S V) \quad \text{subject to } V^T V = I$$

- $\hat{x}_i = a + V b_i = \bar{x} + V_{(r)} V_{(r)}^T (x_i - \bar{x})$

  For centered data, $\hat{x}_i = V_{(r)} V_{(r)}^T x_i \iff$

  $$\hat{X} = X \cdot V_{(r)} V_{(r)}^T$$

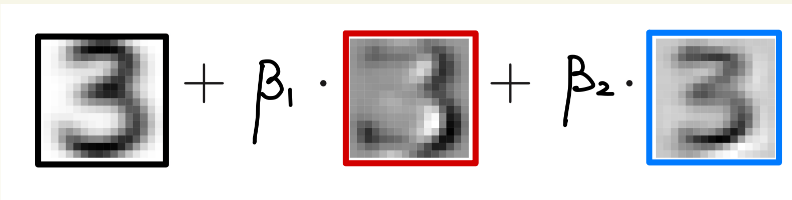
PCA on images (from ESLII)



$X = \begin{pmatrix} \text{—} \\ \text{—} \\ \text{—} \\ \text{—} \\ \text{—} \\ \text{—} \end{pmatrix}$ # of images

256 pixels

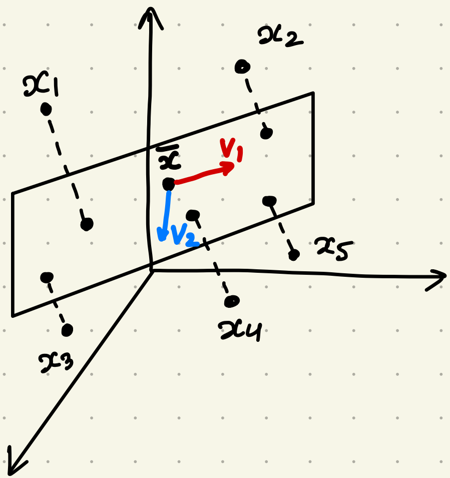
any point in (PC_1, PC_2) plane is



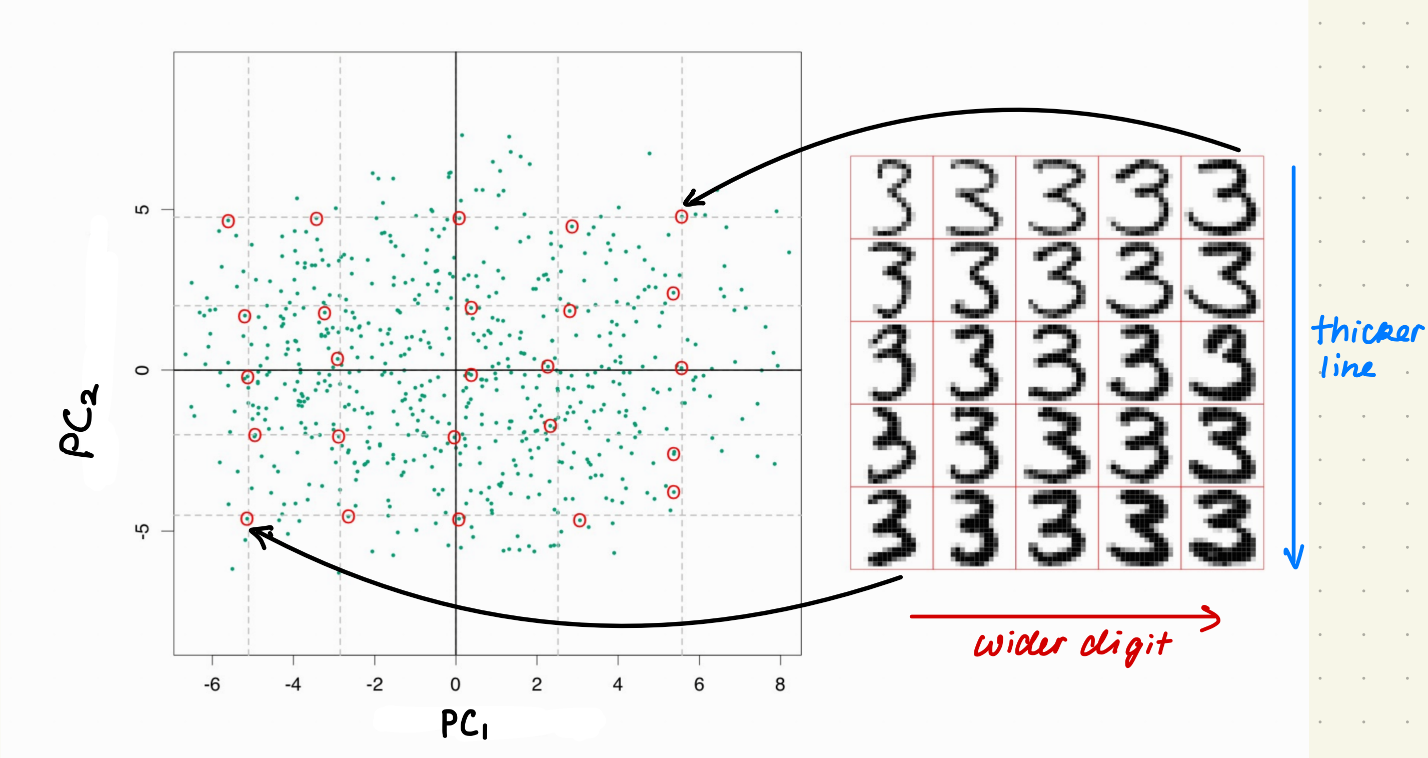
$\bar{x} + \beta_1 \cdot v_1 + \beta_2 \cdot v_2$

"digit width"

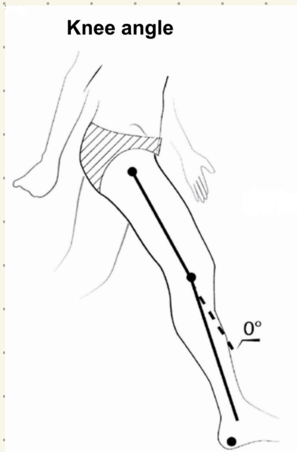
"line thickness"

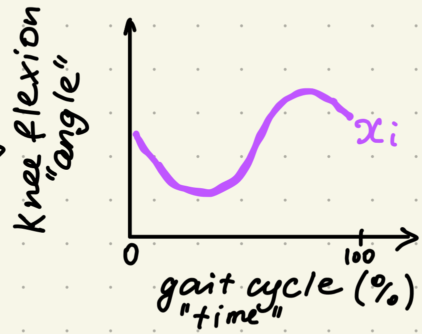


Each point in the biplot corresponds to an image

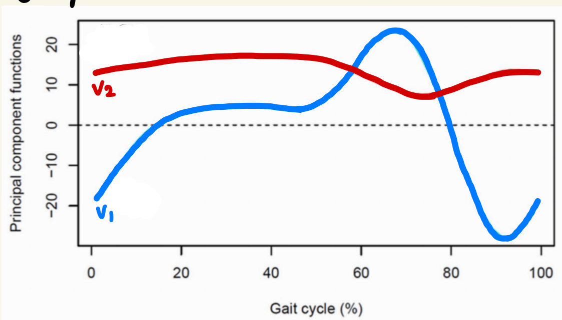


PCA on functional data (from PCA by Greenacre et al)



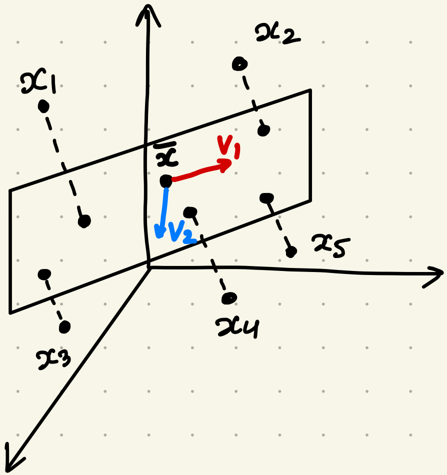


any point in (PC_1, PC_2) plane is

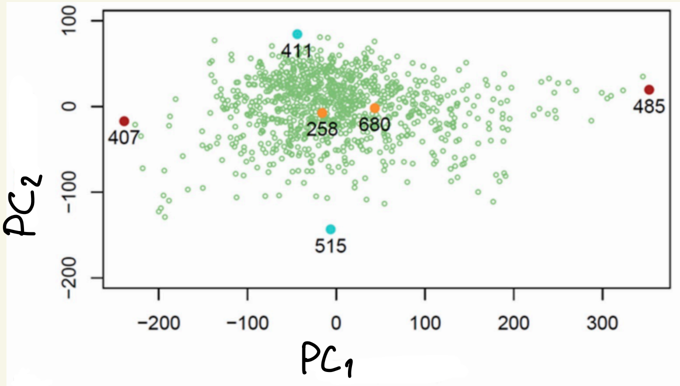


$$\bar{x} \quad + \quad \beta_1 \cdot v_1 \quad + \quad \beta_2 \cdot v_2$$

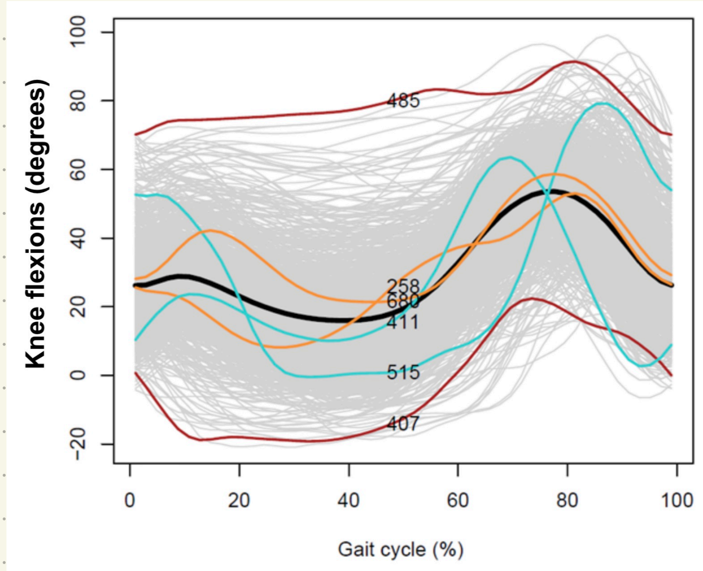
v_1: "size" $\qquad$ v_2: "shape"



horiz. phase shift



vertical shift



Black thick line is $\bar{x}$.

Low-dimensional data

If $x_1 \dots x_n \in \mathbb{R}^p$ belong to an r-dim plane then $X = \begin{pmatrix} - x_1^T - \\ \dots \\ - x_n^T - \end{pmatrix} \in \mathbb{R}^{n \times p}$ has rank r (at most).

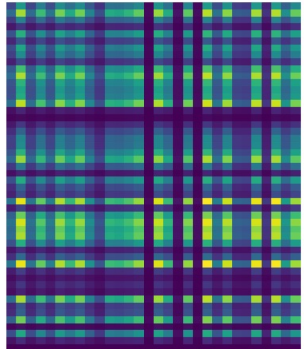
There is basis $v_1 \dots v_r \in \mathbb{R}^p$ such that each $x_i = \beta_{i1} v_1 + \dots + \beta_{ir} v_r$ for some $\beta_{i1} \dots \beta_{ir}$

$$= \begin{pmatrix} | & & | \\ v_1 & \dots & v_r \\ | & & | \end{pmatrix} \cdot b_i \quad \text{where } b_i \in \mathbb{R}^r.$$

Thus $\begin{pmatrix} - x_1^T - \\ - x_n^T - \end{pmatrix} = \begin{pmatrix} - b_1^T - \\ \dots \\ - b_n^T - \end{pmatrix} \begin{pmatrix} - v_1^T - \\ \dots \\ - v_r^T - \end{pmatrix} = \underset{n \times r}{B} \, \underset{r \times p}{V^T}$

$r=1$

$$X = \underset{n\times 1}{\begin{pmatrix} \beta_1 \\ \vdots \\ \beta_n \end{pmatrix}} \underset{1\times p}{v_1^T} = \begin{pmatrix} \beta_{11} v_1^T \\ \cdots \\ \beta_{n1} v_1^T \end{pmatrix}$$

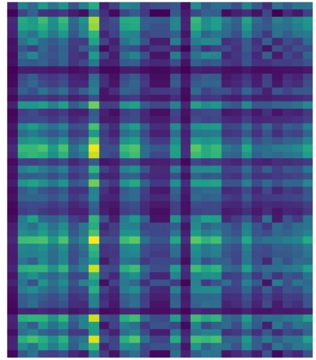
$X = $ 

$\underset{50\times 30}{}$

$\mathrm{rank}(X) = 1$

$r=2$

$$X = \underset{n\times 2}{\begin{pmatrix} - \beta_1^T - \\ \cdots \\ - \beta_n^T - \end{pmatrix}} \underset{2\times p}{\begin{pmatrix} - v_1^T - \\ - v_2^T - \end{pmatrix}} = \begin{pmatrix} \beta_{11} v_1^T + \beta_{12} v_2^T \\ \cdots \\ \beta_{n1} v_1^T + \beta_{n2} v_2^T \end{pmatrix}$$

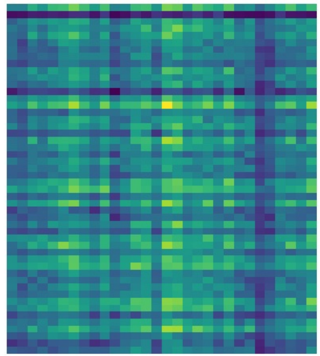
$X = $ 

$\underset{50\times 30}{}$

$\mathrm{rank}(X) = 2$

$r=10$

$$X = \underset{n\times 10}{\begin{pmatrix} - \beta_1^T - \\ \cdots \\ - \beta_n^T - \end{pmatrix}} \underset{10\times p}{\begin{pmatrix} - v_1^T - \\ \cdots \\ - v_{10}^T - \end{pmatrix}} = \begin{pmatrix} \beta_{11} v_1^T + \ldots + \beta_{110} v_r^T \\ \cdots \\ \beta_{n1} v_1^T + \ldots + \beta_{n10} v_r^T \end{pmatrix}$$

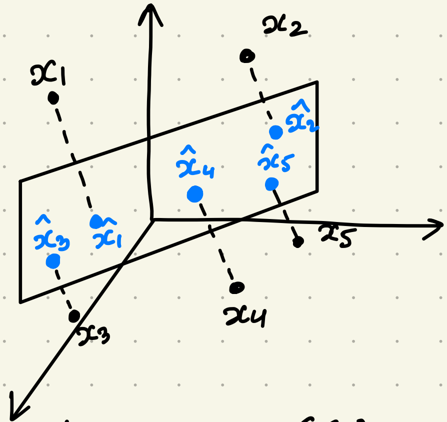
$X = $ 

$\underset{50\times 30}{}$

$\mathrm{rank}(X) = 10$

3rd view of PCA : low-rank approximation

Given $X \in \mathbb{R}^{n \times p}$ find the "best" approximation by $\hat{X} \in \mathbb{R}^{n \times p}$ with rank of $\hat{X}$ equal to r.



The approximation error

$$\sum_{i=1}^{n} \sum_{j=1}^{p} (X_{ij} - \hat{X}_{ij})^2 = \| X - \hat{X} \|_F^2$$

We aim to solve:

$$\underset{\hat{X}}{\text{minimize}} \; \| X - \hat{X} \|_F^2 \quad \text{subject to } \operatorname{rank}(\hat{X}) = r$$

- we can decompose $\underset{}{\hat{X}} = \underset{n \times r}{Q} \, \underset{r \times p}{A}$ with $Q^T Q = I$

  rank factorization + QR $\quad \hat{X} = \underset{n \times r}{C} \, \underset{r \times p}{F^T} = \underset{n \times r}{Q} (\underset{r \times r}{R} \, \underset{r \times p}{F^T}) = Q A$

- given Q find A as $A = Q^T X$

  $\| X - QA \|_F^2 = \sum_{j=1}^{p} \| x_j - Q a_j \|_F^2$ where $X = \begin{pmatrix} | & & | \\ x_1 & \cdots & x_p \\ | & & | \end{pmatrix}$, $A = \begin{pmatrix} | & & | \\ a_1 & \cdots & a_p \\ | & & | \end{pmatrix}$

  It's a regression problem, so $A = (Q^T Q)^{-1} Q^T X$

- minimizing $\|X - QQ^TX\|_F^2$ w.r.t orthogonal Q is equivalent to maximizing $tr(D^2H)$ w.r.t. orthogonal projection $H \in \mathbb{R}^{n\times n}$ onto r-dim space.

Here $D^2 = \begin{bmatrix} \begin{matrix} d_1^2 & & \\ & \ddots & \\ & & d_p^2 \end{matrix} & 0 \\ 0 & 0 \end{bmatrix}$ where $d_1 \ldots d_p$ are s.values of X.

$$\|X - QQ^TX\|_F^2 = tr(X^TX) - 2\,tr(X^TQQ^TX) + tr(X^TQ\cancel{Q^TQ}Q^TX) =$$

$$= -tr(Q^TXX^TQ) + \ldots$$

$X X^T = \underset{n\times n}{U} D^2 U^T$ then the goal of minimizing

$$tr(Q^TXX^TQ) = tr(\underbrace{Q^TU}_{Q_*^T} D^2 \underbrace{U^TQ}_{Q_*}) = tr(Q_*^T D^2 Q_*) = tr(D^2 \underbrace{Q_* Q_*^T}_{H})$$

Here $Q_* \in \mathbb{R}^{n\times r}$ and $Q_*^TQ_* = I$ and H is projection onto Q_*.

- Optimal Q is $Q = U_{(r)}$, where $U_{(r)} = (u_1 \ldots u_r)$ are left s. vectors.

$$tr(D^2H) = \sum_{i=1}^{p} d_i^2 h_{ii} + \cancel{\sum_{i=p+1}^{n} 0 \cdot h_{ii}} \quad \text{where } d_1^2 \geq \ldots \geq d_p^2$$

$tr(H) = \sum_{i=1}^{n} h_{ii} = r \; [= tr(\underset{r\times n}{Q_*^T} \underset{n\times r}{Q_*})]$ and $0 \le h_{ii} \le 1$

Optimal H would have $h_{11} = \dots = h_{rr} = 1$

What H is projection with $h_{11} = \dots = h_{rr} = 1$?

Take $Q = U_{(r)}$, then $Q^* = U^T Q = \left(U_{(r)} \; (U_{(r)})_\perp\right)^T U_{(r)} = \begin{pmatrix} I_r \\ 0 \end{pmatrix}$

$H = Q_* Q_*^T = \begin{pmatrix} I_r \\ 0 \end{pmatrix} (I_r \; 0) = \begin{pmatrix} I_r & 0 \\ 0 & 0 \end{pmatrix}$ follows the condition.

- $A = D_{(r)} V_{(r)}^T = \begin{pmatrix} d_1 & & \\ & \ddots & \\ & & d_r \end{pmatrix} \begin{pmatrix} - v_1^T - \\ \vdots \\ - v_r^T - \end{pmatrix}$

$A = Q^T X = U_{(r)}^T U D V^T = (I_r \; 0) D V^T = (D_{(r)} \; 0) V^T = D_{(r)} V_{(r)}^T$

- The solution is $\hat{X} = U_{(r)} D_{(r)} V_{(r)}^T$.

Thus $\hat{X}$ has rank r.