

# Principal Component Analysis (PCA)

2IMW30 - Foundations of data mining  
TU Eindhoven, Quartile 3, 2016-2017

Anne Driemel

# Why reduce the dimension?

Representation of input data often is often high dimensional (images, documents, etc.)

There are two main reasons to reduce the dimension:

- some algorithms have **running time** exponential in the dimension
- we want to **visualize** inherent structure in the data

# Why reduce the dimension?

Representation of input data often is often high dimensional (images, documents, etc.)

There are two main reasons to reduce the dimension:

- some algorithms have **running time** exponential in the dimension
- we want to **visualize** inherent structure in the data

# Overview of this lecture

- Principal Component Analysis (PCA)
- Interpretation of Principle Components
- Computing Principal Components
- Singular-Value Decomposition (SVD)
- Power Method
- Eigenvectors of the Sample Covariance Matrix
- Multidimensional scaling
- Isomap

# Principal Component Analysis (PCA)

Principal components provide a sequence of best linear approximations to a data set

Given a data set  $P = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$ , we want to represent  $P$  using a  $k$ -dimensional linear model

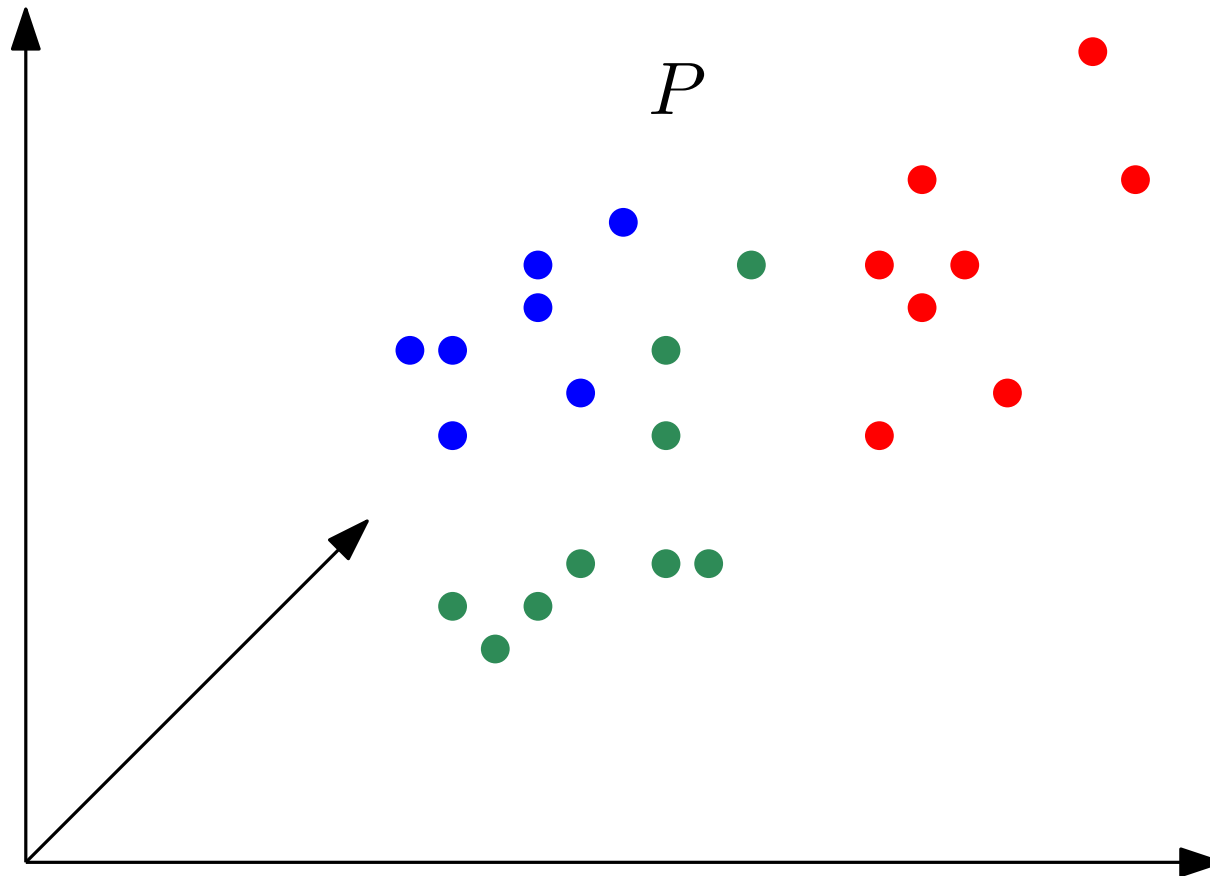
$$f(\lambda) = \mu + \mathbf{V}\lambda,$$

where

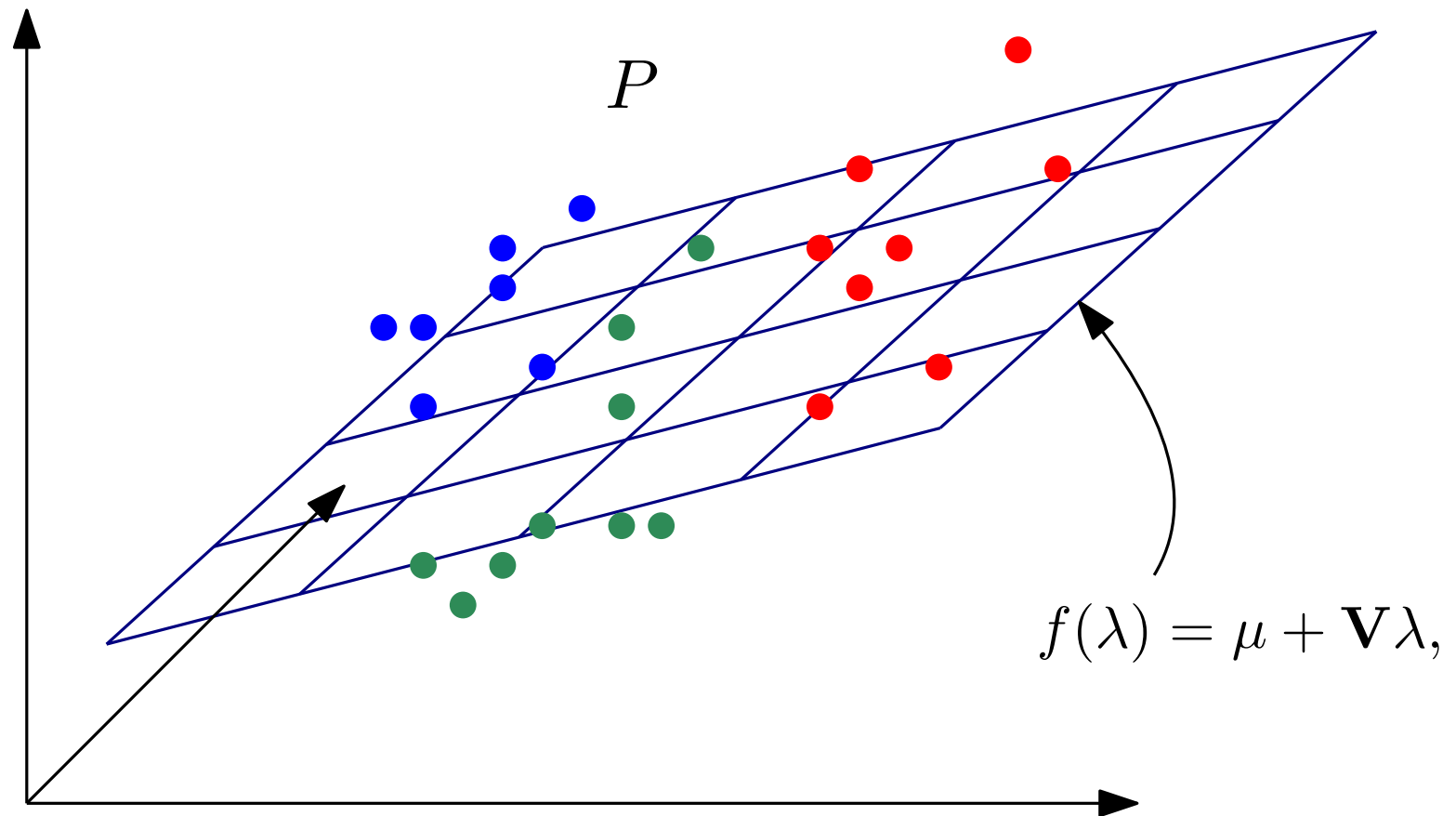
- $\mu$  is a location vector in  $\mathbb{R}^d$
- $\mathbf{V}$  is a  $d \times k$  orthonormal matrix
- $\lambda$  is a  $k$  vector of parameters

The above is a parametric representation of an affine hyperplane of dimension  $k$

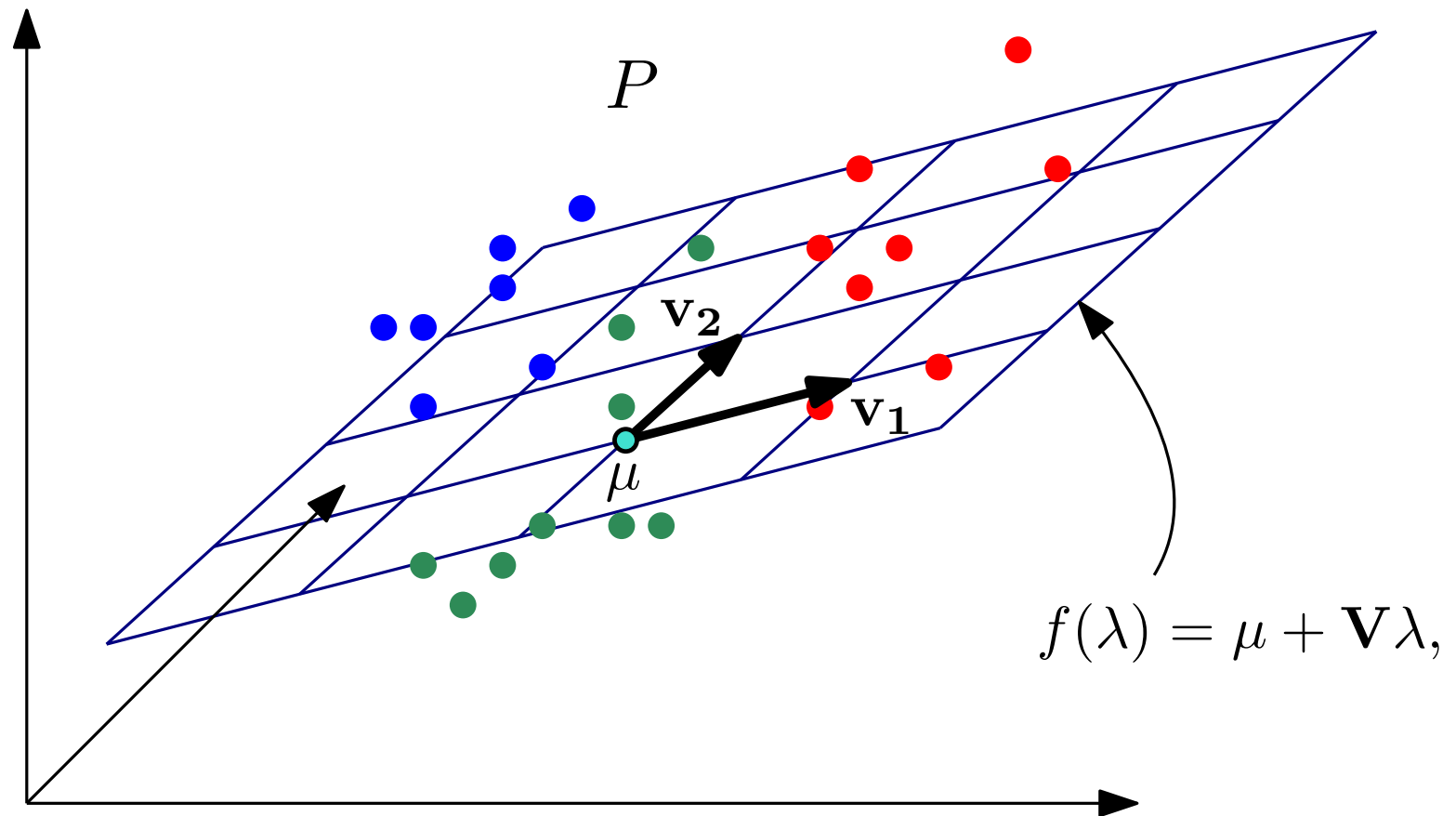
# Principal Component Analysis (PCA)



# Principal Component Analysis (PCA)

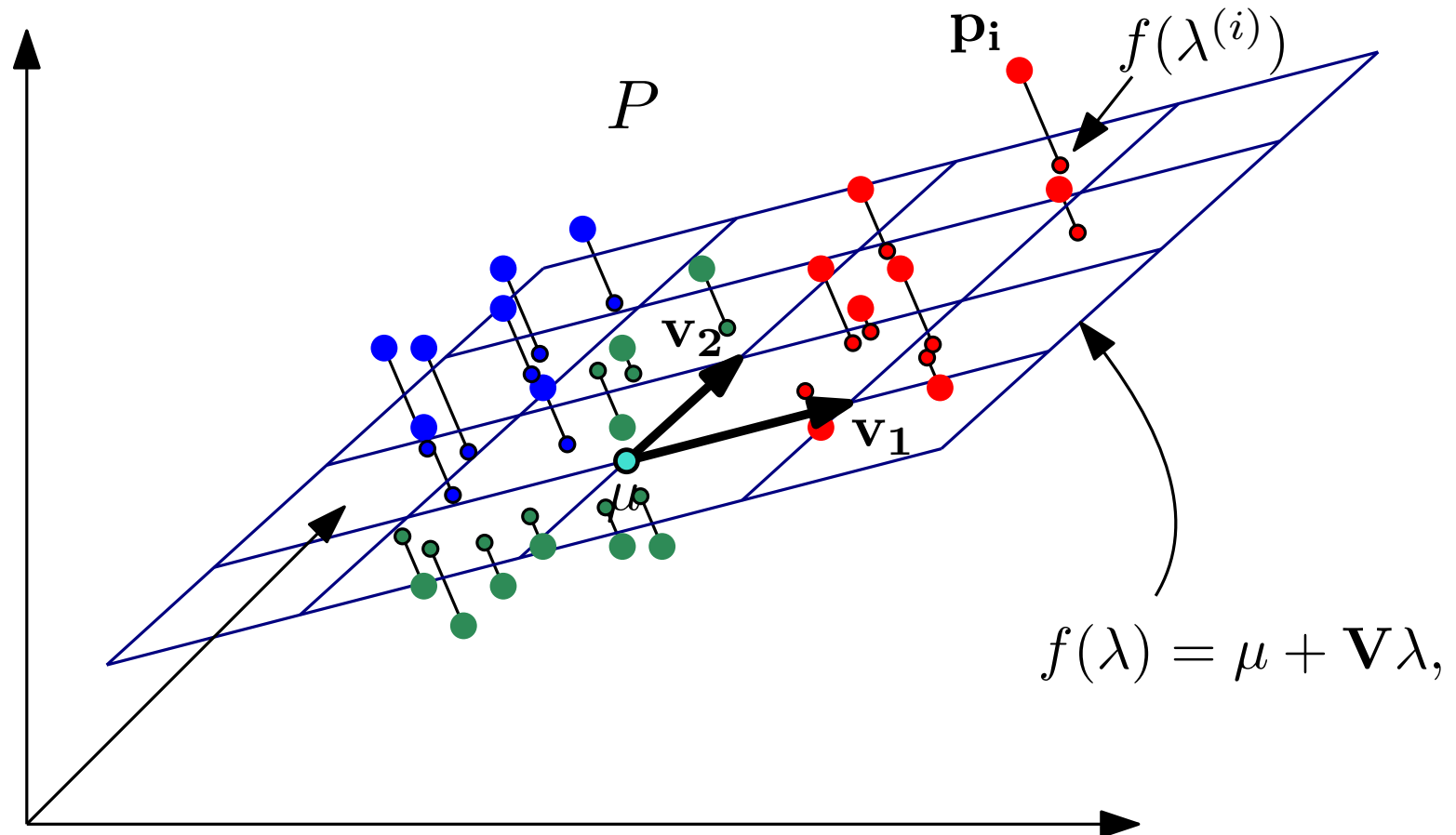


# Principal Component Analysis (PCA)





# Principal Component Analysis (PCA)

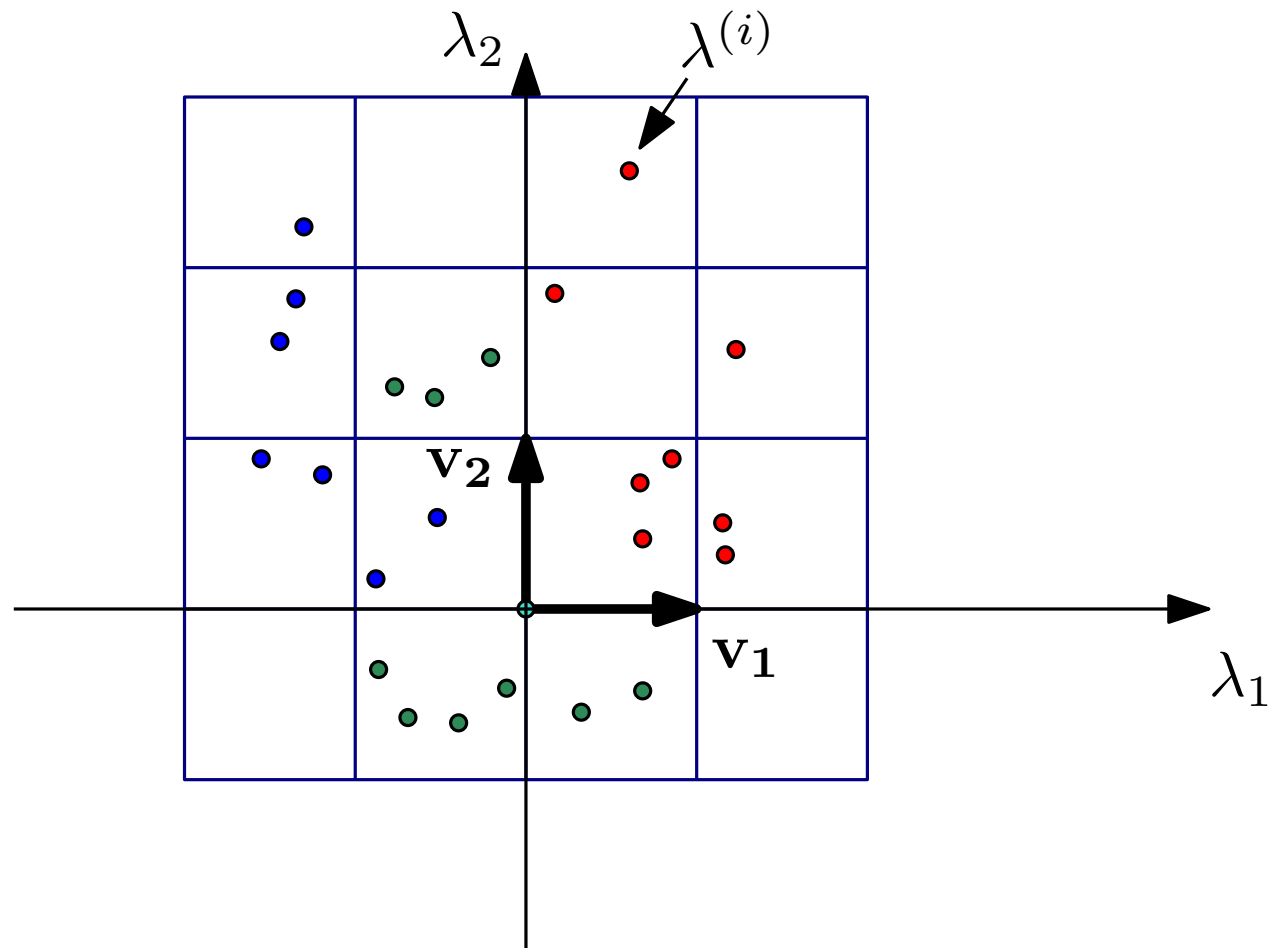


Want to find the hyperplane which minimizes sum of squared distances ("best fitting")

$$\sum_{1 \leq i \leq n} \|\mathbf{p}_i - f(\lambda^{(i)})\|^2$$

# Principal Component Analysis (PCA)

We can visualize  $P$  in the subspace spanned by  $\mathbf{v}_1$  and  $\mathbf{v}_2$  by plotting the principle coordinates  $\lambda$ .



# Principal Component Analysis (PCA)

We have our linear model

$$f(\lambda) = \mu + \mathbf{V}\lambda,$$

where

- $\mu$  is a location vector in  $\mathbb{R}^d$
- $\mathbf{V}$  is a  $d \times k$  matrix
- $\lambda$  is a  $k$  vector of parameters

We have a function that defines "best fitting"

$$\min_{\mu, \mathbf{V}_k, \lambda} \sum_{1 \leq i \leq n} \|\mathbf{p}_i - f(\lambda^{(i)})\|^2$$

# Principal Component Analysis (PCA)

We have our linear model

$$f(\lambda) = \mu + \mathbf{V}\lambda,$$

where

- $\mu$  is a location vector in  $\mathbb{R}^d$
- $\mathbf{V}$  is a  $d \times k$  matrix
- $\lambda$  is a  $k$  vector of parameters

We have a function that defines "best fitting"

$$\min_{\mu, \mathbf{V}_k, \lambda} \sum_{1 \leq i \leq n} \|\mathbf{p}_i - f(\lambda^{(i)})\|^2$$

Optimizing for  $\mu$  and  $\lambda$  gives

$$\mu = \frac{1}{n} \sum_{1 \leq i \leq n} \mathbf{p}_i \quad \text{and} \quad \lambda^{(i)} = \mathbf{V}^T (\mathbf{p}_i - \mu)$$

# Principal Component Analysis (PCA)

We have our linear model

$$f(\lambda) = \mu + \mathbf{V}\lambda,$$

where

- $\mu$  is a location vector in  $\mathbb{R}^d$
- $\mathbf{V}$  is a  $d \times k$  matrix
- $\lambda$  is a  $k$  vector of parameters

We can assume that  $\mu$  is the mean of the data

We have a function that defines "best fitting"

$$\min_{\mu, \mathbf{V}_k, \lambda} \sum_{1 \leq i \leq n} \|\mathbf{p}_i - f(\lambda^{(i)})\|^2$$

Optimizing for  $\mu$  and  $\lambda$  gives

$$\mu = \frac{1}{n} \sum_{1 \leq i \leq n} \mathbf{p}_i$$

and

$$\lambda^{(i)} = \mathbf{V}^T(\mathbf{p}_i - \mu)$$

# Principal Component Analysis (PCA)

We have our linear model

$$f(\lambda) = \mu + \mathbf{V}\lambda,$$

where

- $\mu$  is a location vector in  $\mathbb{R}^d$
- $\mathbf{V}$  is a  $d \times k$  matrix
- $\lambda$  is a  $k$  vector of parameters

We can assume that  $\mu$  is the mean of the data

... and we use the projection onto  $\mathbf{V}$  for  $\lambda$

We have a function that defines "best fitting"

$$\min_{\mu, \mathbf{V}_k, \lambda} \sum_{1 \leq i \leq n} \|\mathbf{p}_i - f(\lambda^{(i)})\|^2$$

Optimizing for  $\mu$  and  $\lambda$  gives

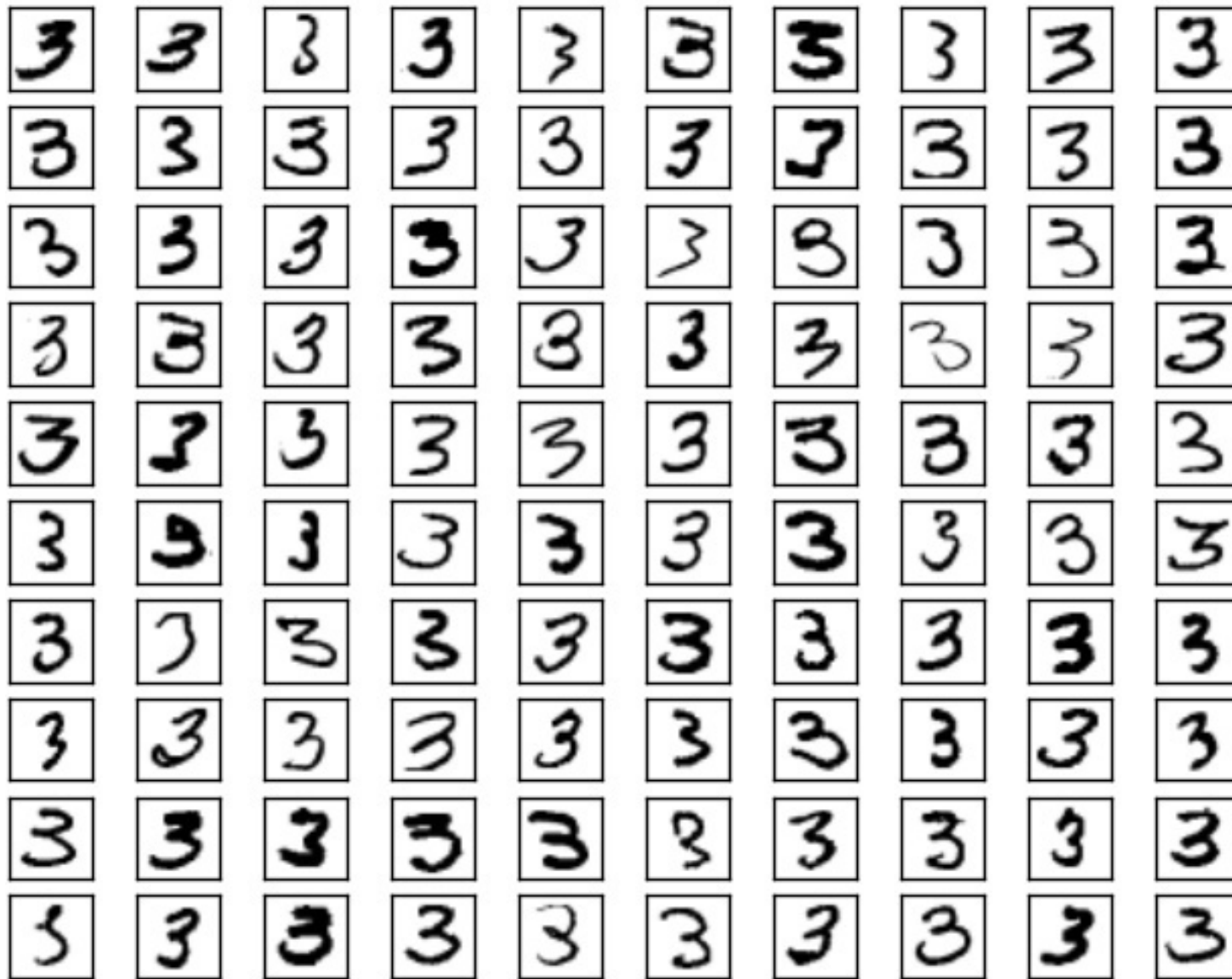
$$\mu = \frac{1}{n} \sum_{1 \leq i \leq n} \mathbf{p}_i$$

and

$$\lambda^{(i)} = \mathbf{V}^T (\mathbf{p}_i - \mu)$$

# Principal Component Analysis (PCA)

**Example:** handwritten digits



# Principal Component Analysis (PCA)

**Example:** handwritten digits

Assume we computed the first two principal components

We obtain an interpretable representation

$$\hat{f}(\lambda) = \mu + \mathbf{V}\lambda,$$

$$= \mu + \lambda_1 \mathbf{v}_1 + \lambda_2 \mathbf{v}_2$$

$$= \begin{array}{c} \boxed{\text{3}} \end{array} + \lambda_1 \cdot \begin{array}{c} \boxed{\text{3}} \end{array} + \lambda_2 \cdot \begin{array}{c} \boxed{\text{3}} \end{array}$$

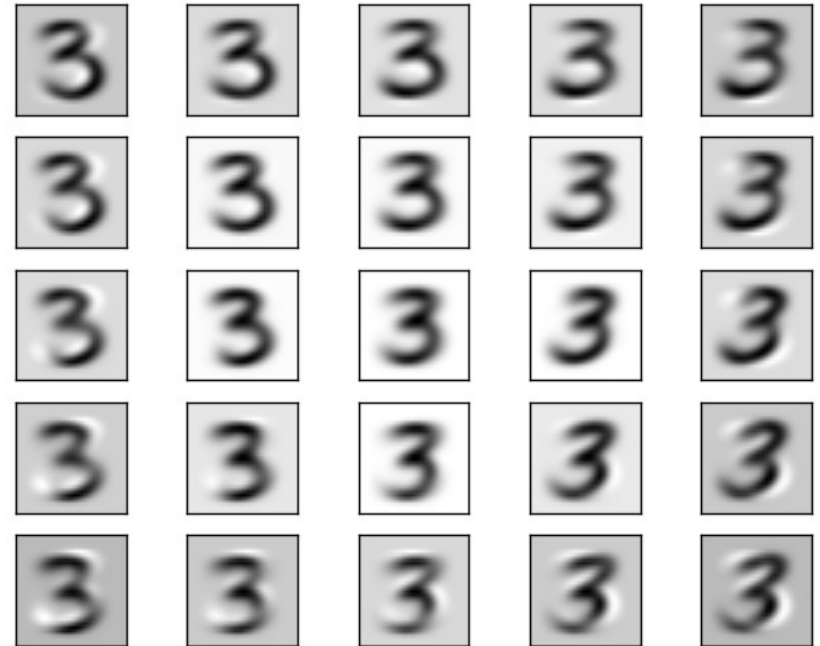
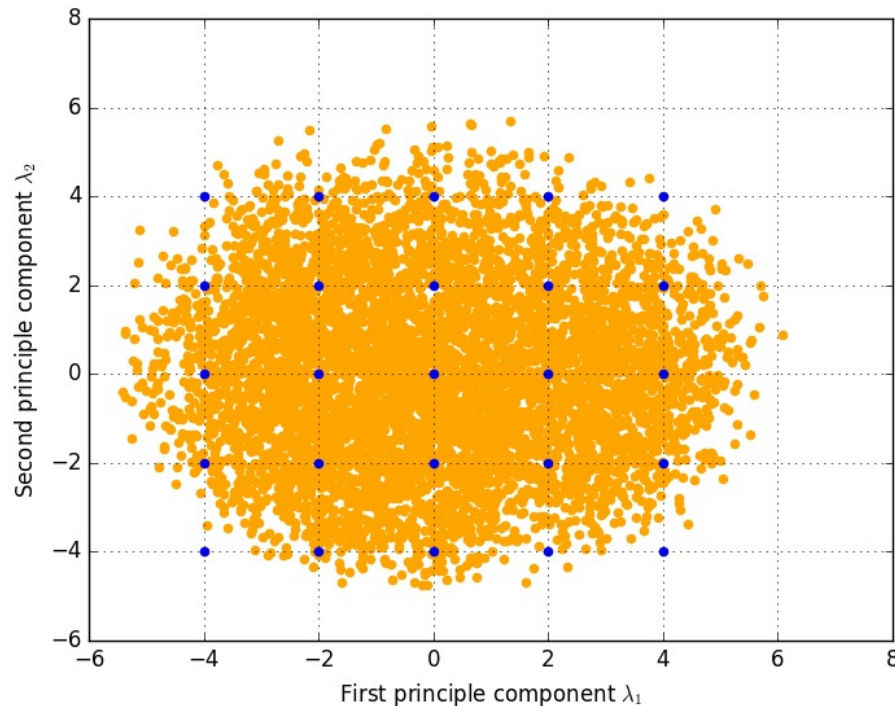
↑  
mean

↑      ↑  
principle components



# Principal Component Analysis (PCA)

**Example:** handwritten digits

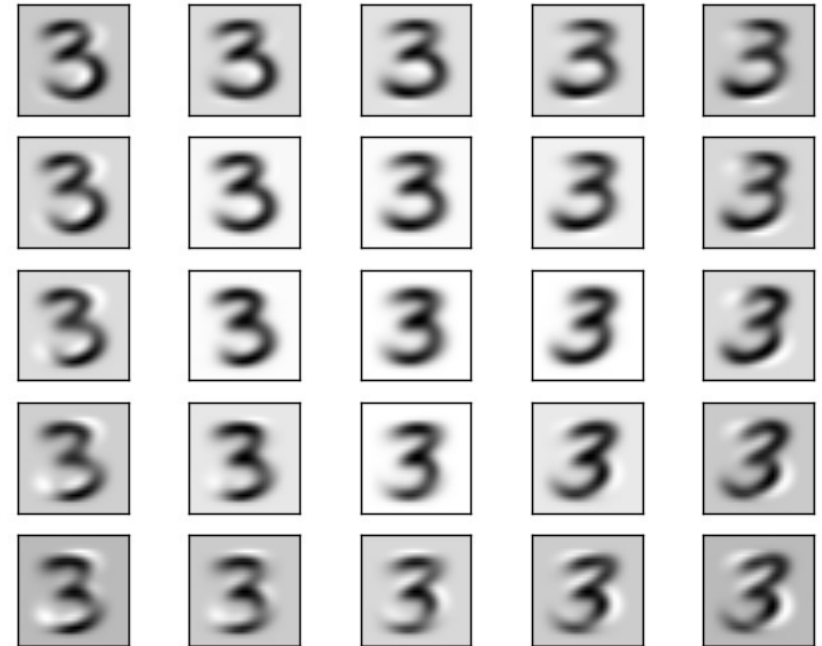
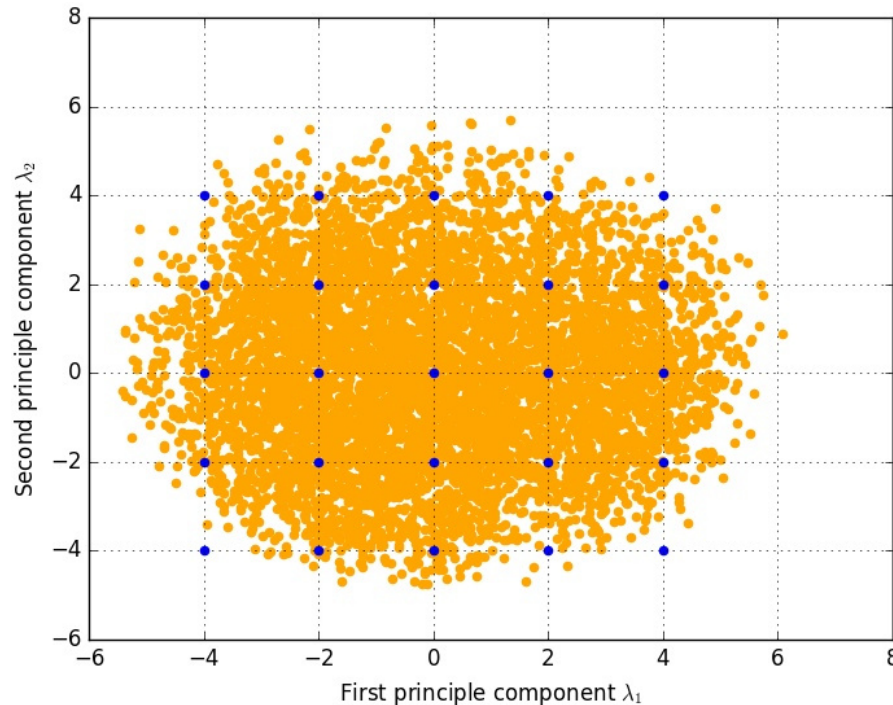


$$\begin{array}{|c|} \hline \text{3} \\ \hline \end{array} + \lambda_1 \cdot \begin{array}{|c|} \hline \text{3} \\ \hline \end{array} + \lambda_2 \cdot \begin{array}{|c|} \hline \text{3} \\ \hline \end{array}$$

**Interpretation?**

# Principal Component Analysis (PCA)

**Example:** handwritten digits



$$\begin{array}{c} \text{3} \end{array} + \lambda_1 \cdot \begin{array}{c} \text{3} \end{array} + \lambda_2 \cdot \begin{array}{c} \text{3} \end{array}$$

**Interpretation?**

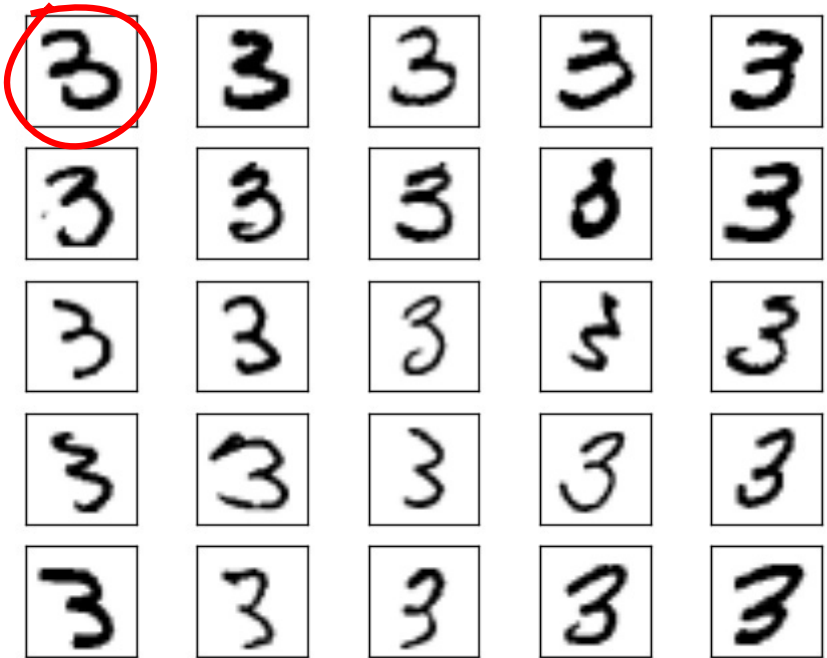
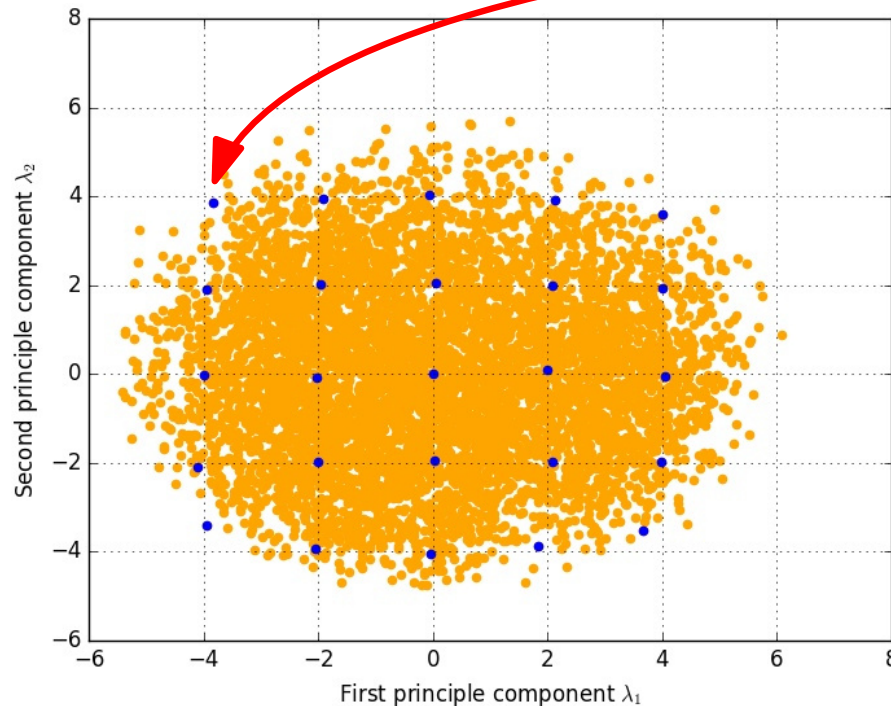
"slanting"

"lengthening of lower tail"

# Principal Component Analysis (PCA)

**Example:** handwritten digits

Instances of which the projections are closest to the grid points



$$\begin{array}{|c|} \hline 3 \\ \hline \end{array} + \lambda_1 \cdot \begin{array}{|c|} \hline 3 \\ \hline \end{array} + \lambda_2 \cdot \begin{array}{|c|} \hline 3 \\ \hline \end{array}$$

**Interpretation?**

"slanting"

"lengthening of lower tail"

# Principal Component Analysis (PCA)

We have defined PCA as an optimization problem:  
Fitting a  $k$ -dimensional hyperplane to the data

$$f(\lambda) = \mu + \mathbf{V}\lambda,$$

# Principal Component Analysis (PCA)

We have defined PCA as an optimization problem:  
Fitting a  $k$ -dimensional hyperplane to the data

$$f(\lambda) = \mu + \mathbf{V}\lambda,$$

How do we compute  $\mathbf{V}$ ?

# Principal Component Analysis (PCA)

We have defined PCA as an optimization problem:  
Fitting a  $k$ -dimensional hyperplane to the data

$$f(\lambda) = \mu + \mathbf{V}\lambda,$$

How do we compute  $\mathbf{V}$ ?

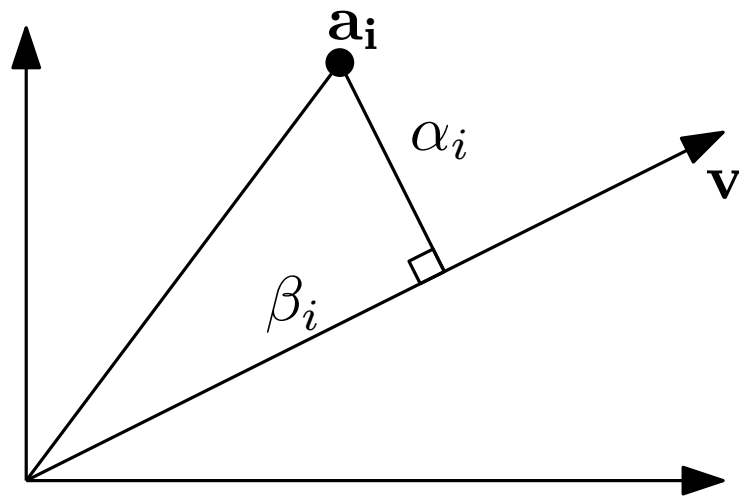
In the following, let  $\mathbf{A}$  be a  $n \times d$  matrix with row vectors  $\mathbf{a}_i$  with

$$\mathbf{a}_i = \mathbf{p}_i - \mu$$

$\mathbf{A}$  is a **centered** version of  $P$

# Computing the principal components

Simplest case: fitting a line through the origin to  $\mathbf{A}$

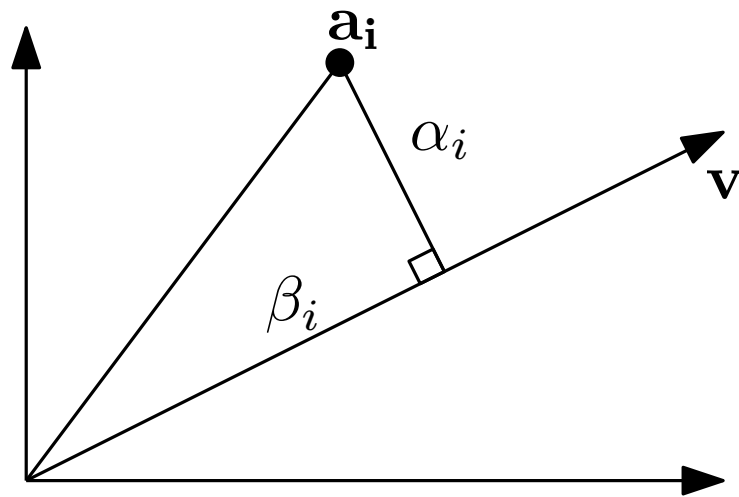


(Pythagoras)

$$\|\mathbf{a}_i\|^2 = \alpha_i^2 + \beta_i^2$$

# Computing the principal components

Simplest case: fitting a line through the origin to  $\mathbf{A}$



(Pythagoras)

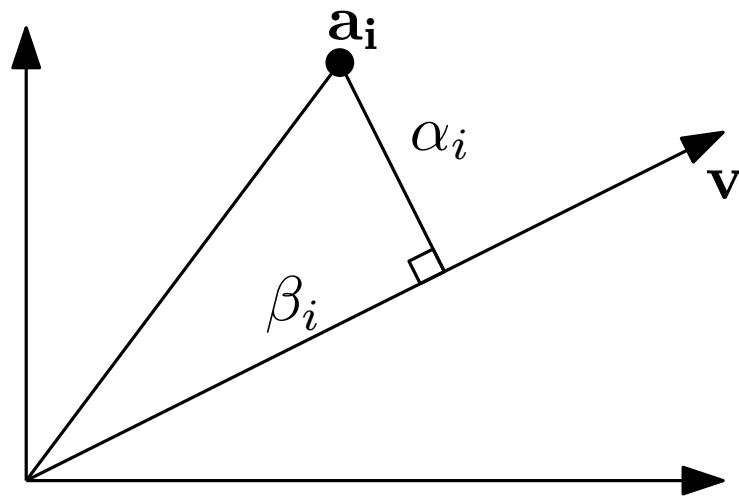
$$\|\mathbf{a}_i\|^2 = \alpha_i^2 + \beta_i^2$$

$$\Leftrightarrow \alpha_i^2 = \|\mathbf{a}_i\|^2 - \beta_i^2$$



# Computing the principal components

Simplest case: fitting a line through the origin to  $\mathbf{A}$



(Pythagoras)

$$\|\mathbf{a}_i\|^2 = \alpha_i^2 + \beta_i^2$$

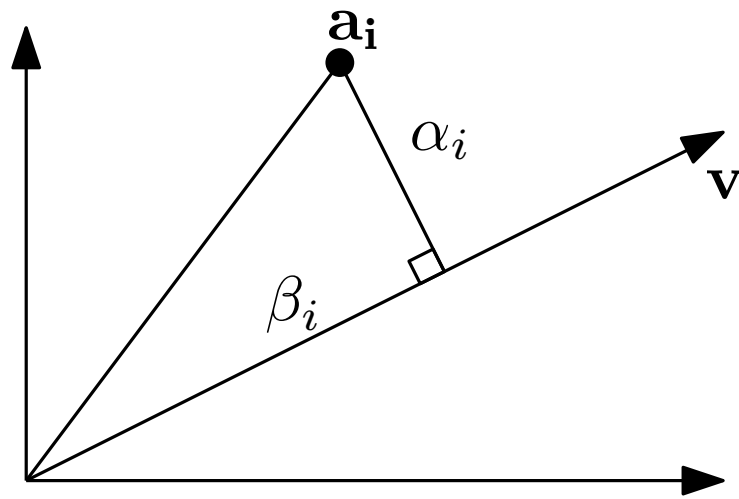
$$\Leftrightarrow \alpha_i^2 = \|\mathbf{a}_i\|^2 - \beta_i^2$$

$$\operatorname{argmin}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} \alpha_i^2 = \operatorname{argmin}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} \|\mathbf{a}_i\|^2 - \beta_i^2$$

"best fitting"

# Computing the principal components

Simplest case: fitting a line through the origin to  $\mathbf{A}$



(Pythagoras)

$$\|\mathbf{a}_i\|^2 = \alpha_i^2 + \beta_i^2$$

$$\Leftrightarrow \alpha_i^2 = \|\mathbf{a}_i\|^2 - \beta_i^2$$

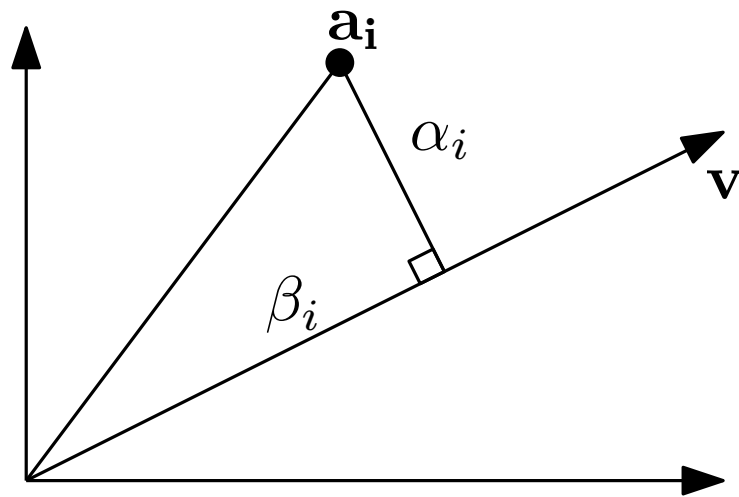
$$\operatorname{argmin}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} \alpha_i^2 = \operatorname{argmin}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} \|\mathbf{a}_i\|^2 - \beta_i^2$$

"best fitting"

$$= \operatorname{argmax}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} \beta_i^2$$

# Computing the principal components

Simplest case: fitting a line through the origin to  $\mathbf{A}$



(Pythagoras)

$$\|\mathbf{a}_i\|^2 = \alpha_i^2 + \beta_i^2$$

$$\Leftrightarrow \alpha_i^2 = \|\mathbf{a}_i\|^2 - \beta_i^2$$

$$\operatorname{argmin}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} \alpha_i^2 = \operatorname{argmin}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} \|\mathbf{a}_i\|^2 - \beta_i^2$$

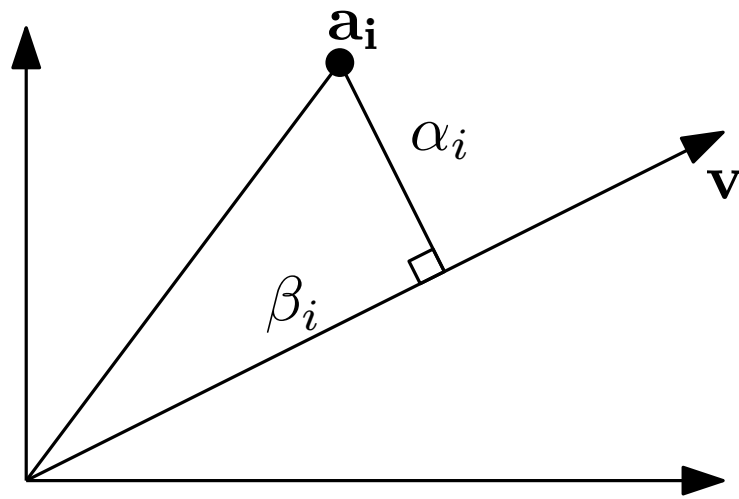
"best fitting"

$$= \operatorname{argmax}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} \beta_i^2$$

$$= \operatorname{argmax}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} (\mathbf{a}_i \mathbf{v})^2$$

# Computing the principal components

Simplest case: fitting a line through the origin to  $\mathbf{A}$



(Pythagoras)

$$\|\mathbf{a}_i\|^2 = \alpha_i^2 + \beta_i^2$$

$$\Leftrightarrow \alpha_i^2 = \|\mathbf{a}_i\|^2 - \beta_i^2$$

$$\operatorname{argmin}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} \alpha_i^2 = \operatorname{argmin}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} \|\mathbf{a}_i\|^2 - \beta_i^2$$

"best fitting"

$$= \operatorname{argmax}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} \beta_i^2$$

$$= \operatorname{argmax}_{\|\mathbf{v}\|=1} \sum_{1 \leq i \leq n} (\mathbf{a}_i \mathbf{v})^2 = \operatorname{argmax}_{\|\mathbf{v}\|=1} \|\mathbf{A} \mathbf{v}\|^2$$

# Computing the principal components

$\mathbf{A}$  is a  $n \times d$  matrix with row vectors  $\mathbf{a}_i$

The first singular vector of  $A$  is:

$$\mathbf{v}_1 = \operatorname{argmax}_{\|\mathbf{v}\|=1} \|\mathbf{A}\mathbf{v}\|$$

The first singular value of  $A$  is:

$$\sigma_1 = \|\mathbf{A}\mathbf{v}_1\|$$

# Computing the principal components

$\mathbf{A}$  is a  $n \times d$  matrix with row vectors  $\mathbf{a}_i$

The first singular vector of  $A$  is:

$$\mathbf{v}_1 = \operatorname{argmax}_{\|\mathbf{v}\|=1} \|\mathbf{A}\mathbf{v}\|$$

The first singular value of  $A$  is:

$$\sigma_1 = \|\mathbf{A}\mathbf{v}_1\|$$

The second singular vector of  $A$  is:

$$\mathbf{v}_2 = \operatorname{argmax}_{\substack{\|\mathbf{v}\|=1 \\ \mathbf{v} \perp \mathbf{v}_1}} \|\mathbf{A}\mathbf{v}\|$$

# Computing the principal components

$\mathbf{A}$  is a  $n \times d$  matrix with row vectors  $\mathbf{a}_i$

The first singular vector of  $A$  is:

$$\mathbf{v}_1 = \operatorname{argmax}_{\|\mathbf{v}\|=1} \|\mathbf{A}\mathbf{v}\|$$

The first singular value of  $A$  is:

$$\sigma_1 = \|\mathbf{A}\mathbf{v}_1\|$$

The second singular vector of  $A$  is:

$$\mathbf{v}_2 = \operatorname{argmax}_{\substack{\|\mathbf{v}\|=1 \\ \mathbf{v} \perp \mathbf{v}_1}} \|\mathbf{A}\mathbf{v}\|$$

The second singular value of  $A$  is:

$$\sigma_2 = \|\mathbf{A}\mathbf{v}_2\|$$

# Computing the principal components

$\mathbf{A}$  is a  $n \times d$  matrix with row vectors  $\mathbf{a}_i$

The first singular vector of  $A$  is:

$$\mathbf{v}_1 = \operatorname{argmax}_{\|\mathbf{v}\|=1} \|\mathbf{A}\mathbf{v}\|$$

The first singular value of  $A$  is:

$$\sigma_1 = \|\mathbf{A}\mathbf{v}_1\|$$

The second singular vector of  $A$  is:

$$\mathbf{v}_2 = \operatorname{argmax}_{\substack{\|\mathbf{v}\|=1 \\ \mathbf{v} \perp \mathbf{v}_1}} \|\mathbf{A}\mathbf{v}\|$$

The second singular value of  $A$  is:

$$\sigma_2 = \|\mathbf{A}\mathbf{v}_2\|$$

...



# Computing the principal components

$\mathbf{A}$  is a  $n \times d$  matrix with row vectors  $\mathbf{a}_i$

The first singular vector of  $A$  is:

$$\mathbf{v}_1 = \operatorname{argmax}_{\|\mathbf{v}\|=1} \|\mathbf{A}\mathbf{v}\|$$

The first singular value of  $A$  is:

$$\sigma_1 = \|\mathbf{A}\mathbf{v}_1\|$$

The second singular vector of  $A$  is:

$$\mathbf{v}_2 = \operatorname{argmax}_{\substack{\|\mathbf{v}\|=1 \\ \mathbf{v} \perp \mathbf{v}_1}} \|\mathbf{A}\mathbf{v}\|$$

The second singular value of  $A$  is:

$$\sigma_2 = \|\mathbf{A}\mathbf{v}_2\|$$

...

The process stops when we have found singular vectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_r$  and singular values

$\sigma_1, \sigma_2, \dots, \sigma_r$  and

$$\max_{\substack{\|\mathbf{v}\|=1 \\ \mathbf{v} \perp \mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_r}} \|\mathbf{A}\mathbf{v}\| = 0$$

# Singular Value Decomposition (SVD)

SVD is the factorization of a matrix  $A$  into three matrices

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$$

where

- $\mathbf{U}$  and  $\mathbf{V}$  are orthonormal
- $\mathbf{D}$  is diagonal with positive real entries  $\sigma_i$
- $\sigma_i$  are in descending order

A diagram illustrating the SVD factorization. It shows a large box on the left containing the matrix  $\mathbf{A}$  with dimensions  $n \times d$ . To its right is an equals sign. Further right are three boxes: the first contains  $\mathbf{U}$  with dimensions  $n \times k$ , the second contains  $\mathbf{D}$  with dimensions  $k \times k$ , and the third contains  $\mathbf{V}^T$  with dimensions  $k \times d$ .

$$\begin{array}{|c|} \hline \mathbf{A} \\ \hline n \times d \\ \hline \end{array} = \begin{array}{|c|} \hline \mathbf{U} \\ \hline n \times k \\ \hline \end{array} \begin{array}{|c|} \hline \mathbf{D} \\ \hline k \times k \\ \hline \end{array} \begin{array}{|c|} \hline \mathbf{V}^T \\ \hline k \times d \\ \hline \end{array}$$

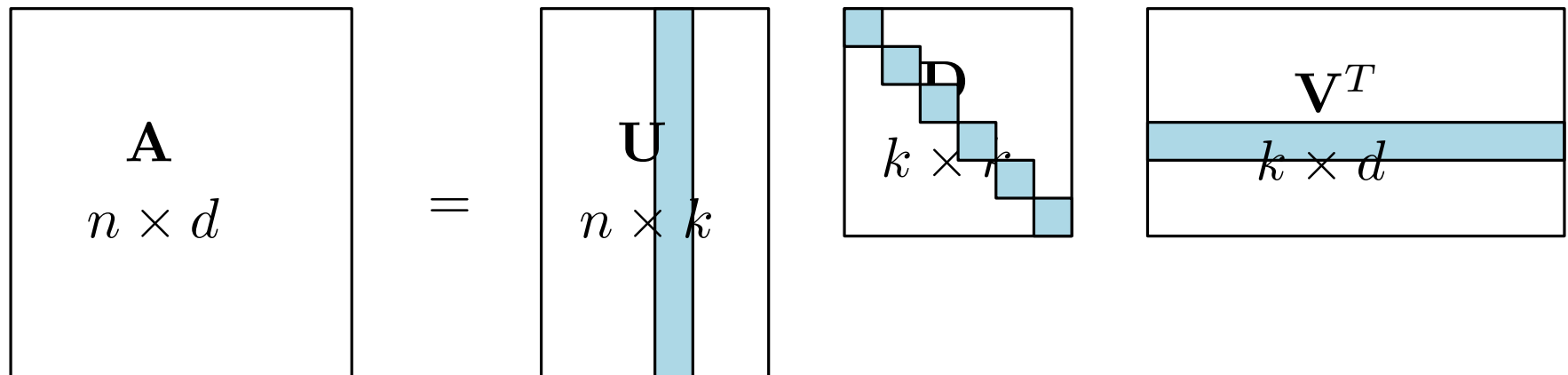
# Singular Value Decomposition (SVD)

SVD is the factorization of a matrix  $A$  into three matrices

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$$

where

- $\mathbf{U}$  and  $\mathbf{V}$  are orthonormal
- $\mathbf{D}$  is diagonal with positive real entries  $\sigma_i$
- $\sigma_i$  are in descending order



Columns of  $\mathbf{V}$  are called **singular vectors**  $\mathbf{v}_1, \mathbf{v}_2, \dots$

Diagonal entries of  $\mathbf{D}$  are called **singular values**  $\sigma_1, \sigma_2, \dots$

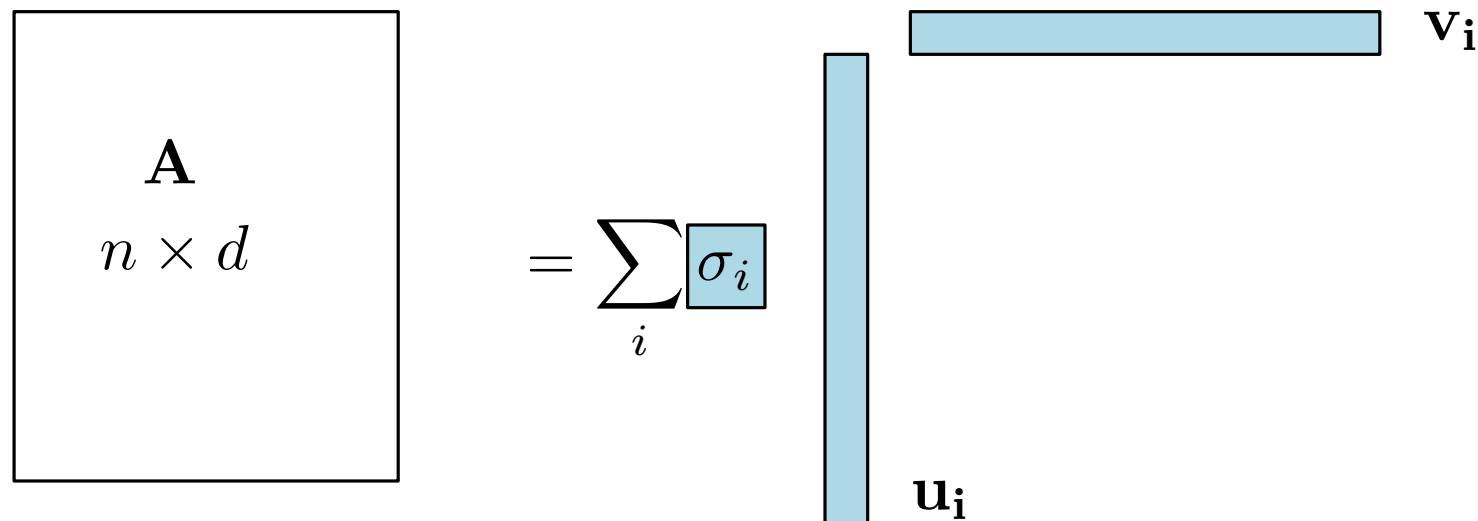
# Singular Value Decomposition (SVD)

$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$  can be rewritten using the sum of outer products

$$\mathbf{A} = \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^T$$

where  $\mathbf{u}_i$  and  $\mathbf{v}_i$  are columns of  $\mathbf{U}$  and  $\mathbf{V}$

The  $i^{th}$  term in the above sum can be viewed as giving the components of the rows of  $\mathbf{A}$  along  $\mathbf{v}_i$



# Power Method

The first principal component  $\mathbf{v}_1$  can be computed using the **power method**:

$$\mathbf{B} = \mathbf{A}^T \mathbf{A} = \left( \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^T \right) \left( \sum_j \sigma_j \mathbf{u}_j \mathbf{v}_j^T \right)$$

# Power Method

The first principal component  $\mathbf{v}_1$  can be computed using the **power method**:

$$\begin{aligned}\mathbf{B} = \mathbf{A}^T \mathbf{A} &= \left( \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^T \right) \left( \sum_j \sigma_j \mathbf{u}_j \mathbf{v}_j^T \right) \\ &= \sum_i \sum_j \sigma_i \sigma_j \mathbf{v}_i (\mathbf{u}_i^T \mathbf{u}_j) \mathbf{v}_j^T\end{aligned}$$

orthogonal  
for  $i \neq j$

# Power Method

The first principal component  $\mathbf{v}_1$  can be computed using the **power method**:

$$\begin{aligned}\mathbf{B} &= \mathbf{A}^T \mathbf{A} = \left( \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^T \right) \left( \sum_j \sigma_j \mathbf{u}_j \mathbf{v}_j^T \right) \\ &= \sum_i \sum_j \sigma_i \sigma_j \mathbf{v}_i (\mathbf{u}_i^T \mathbf{u}_j) \mathbf{v}_j^T \quad \text{orthogonal for } i \neq j \\ &= \sum_i \sigma_i^2 \mathbf{v}_i \mathbf{v}_i^T\end{aligned}$$

# Power Method

The first principal component  $\mathbf{v}_1$  can be computed using the **power method**:

$$\begin{aligned}\mathbf{B} &= \mathbf{A}^T \mathbf{A} = \left( \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^T \right) \left( \sum_j \sigma_j \mathbf{u}_j \mathbf{v}_j^T \right) \\ &= \sum_i \sum_j \sigma_i \sigma_j \mathbf{v}_i (\mathbf{u}_i^T \mathbf{u}_j) \mathbf{v}_j^T \quad \text{orthogonal for } i \neq j \\ &= \sum_i \sigma_i^2 \mathbf{v}_i \mathbf{v}_i^T \\ \mathbf{B}^2 &= \sum_i \sum_j \sigma_i^2 \sigma_j^2 \mathbf{v}_i (\mathbf{v}_i^T \mathbf{v}_j) \mathbf{v}_j^T = \sum_i \sigma_i^4 \mathbf{v}_i \mathbf{v}_i^T\end{aligned}$$



# Power Method

The first principal component  $\mathbf{v}_1$  can be computed using the **power method**:

$$\begin{aligned}\mathbf{B} &= \mathbf{A}^T \mathbf{A} = \left( \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^T \right) \left( \sum_j \sigma_j \mathbf{u}_j \mathbf{v}_j^T \right) \\ &= \sum_i \sum_j \sigma_i \sigma_j \mathbf{v}_i (\mathbf{u}_i^T \mathbf{u}_j) \mathbf{v}_j^T \quad \text{orthogonal for } i \neq j \\ &= \sum_i \sigma_i^2 \mathbf{v}_i \mathbf{v}_i^T \\ \mathbf{B}^2 &= \sum_i \sum_j \sigma_i^2 \sigma_j^2 \mathbf{v}_i (\mathbf{v}_i^T \mathbf{v}_j) \mathbf{v}_j^T = \sum_i \sigma_i^4 \mathbf{v}_i \mathbf{v}_i^T \\ \mathbf{B}^k &= \sum_i \sigma_i^{2k} \mathbf{v}_i \mathbf{v}_i^T \rightarrow \sigma_1^{2k} \mathbf{v}_1 \mathbf{v}_1^T \\ &\quad \text{(using } \sigma_1 > \sigma_2 \text{)}\end{aligned}$$

# Power Method

The first principal component  $\mathbf{v}_1$  can be computed using the **power method**:

$$\begin{aligned}\mathbf{B} &= \mathbf{A}^T \mathbf{A} = \left( \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^T \right) \left( \sum_j \sigma_j \mathbf{u}_j \mathbf{v}_j^T \right) \\ &= \sum_i \sum_j \sigma_i \sigma_j \mathbf{v}_i (\mathbf{u}_i^T \mathbf{u}_j) \mathbf{v}_j^T\end{aligned}$$

orthogonal  
for  $i \neq j$

$$= \sum_i \sigma_i^2 \mathbf{v}_i \mathbf{v}_i^T$$

$$\mathbf{B}^2 = \sum_i \sum_j \sigma_i^2 \sigma_j^2 \mathbf{v}_i (\mathbf{v}_i^T \mathbf{v}_j) \mathbf{v}_j^T =$$

$$\mathbf{B}^k = \sum_i \sigma_i^{2k} \mathbf{v}_i \mathbf{v}_i^T \rightarrow \sigma_1^{2k} \mathbf{v}_1 \mathbf{v}_1^T$$

(using  $\sigma_1 > \sigma_2$ )

We can estimate  $\mathbf{v}_1$  using the first column of  $\mathbf{B}^k$  normalized to unit length

# Interpretation of principal components (again)

**Example:** handwritten digits

Assume we computed the first two principal components

We obtain an interpretable representation

$$\hat{f}(\lambda) = \mu + \mathbf{V}\lambda,$$

$$= \mu + \lambda_1 \mathbf{v}_1 + \lambda_2 \mathbf{v}_2$$

$$= \begin{array}{|c|} \hline \text{3} \\ \hline \end{array} + \lambda_1 \cdot \begin{array}{|c|} \hline \text{3} \\ \hline \end{array} + \lambda_2 \cdot \begin{array}{|c|} \hline \text{3} \\ \hline \end{array}$$

↑  
mean

↑      ↑  
principal components

# An Alternative View

We can view  $\mathbf{a}_i$  as an observation of a multivariate distribution

$\mathbf{A}$  contains  $n$  observations of  $d$  random variables  $X_1, X_2, \dots, X_d$

The **covariance** of two variables  $X_i, X_j$  is defined as

$$\text{cov}(X_i, X_j) = \text{E}[(X_i - \mu_i)(X_j - \mu_j)]$$

with  $\mu_i = \text{E}[X_i]$

The **sample covariance matrix** is defined as

$$\mathbf{M} = \frac{1}{n-1} \underbrace{\sum_{1 \leq i \leq n} (\mathbf{a}_i - \mu)^T (\mathbf{a}_i - \mu)}_{\mathbf{A}^T \mathbf{A}}$$

# An Alternative View

A vector  $\mathbf{v}$  such that

$$B\mathbf{v} = \gamma \mathbf{v}$$

is called an **eigenvector** of  $B$  and  $\gamma$  is called the **eigenvalue**

# An Alternative View

A vector  $\mathbf{v}$  such that

$$B\mathbf{v} = \gamma \mathbf{v}$$

is called an **eigenvector** of  $B$  and  $\gamma$  is called the **eigenvalue**

The following holds true since  $\mathbf{V}^T = \mathbf{V}^{-1}$

$$\mathbf{A}\mathbf{v}_i = \sigma_i \mathbf{u}_i \quad \text{and} \quad \mathbf{A}^T \mathbf{u}_i = \sigma_i \mathbf{v}_i$$

together this implies

$$\mathbf{A}^T \mathbf{A} \mathbf{v}_i = \sigma_i^2 \mathbf{v}_i$$

Therefore, the **singular vectors** of  $\mathbf{A}$  are the **eigenvectors** of the sample covariance matrix

# Multidimensional scaling (Torgerson (1952))

Assume matrix  $\mathbf{A}$  is not available, but instead we are given all squared pairwise distances as  $n \times n$  matrix  $\Delta$

$$\Delta_{ij} = \|\mathbf{a}_i - \mathbf{a}_j\|^2$$

# Multidimensional scaling (Torgerson (1952))

Assume matrix  $\mathbf{A}$  is not available, but instead we are given all squared pairwise distances as  $n \times n$  matrix  $\Delta$

$$\Delta_{ij} = \|\mathbf{a}_i - \mathbf{a}_j\|^2$$

We can recover inner products  $\mathbf{a}_i \mathbf{a}_j^T$  of unknown  $\mathbf{A}$  as follows



# Multidimensional scaling (Torgerson (1952))

Assume matrix  $\mathbf{A}$  is not available, but instead we are given all squared pairwise distances as  $n \times n$  matrix  $\Delta$

$$\Delta_{ij} = \|\mathbf{a}_i - \mathbf{a}_j\|^2$$

We can recover inner products  $\mathbf{a}_i \mathbf{a}_j^T$  of unknown  $\mathbf{A}$  as follows

The following matrix is a **double-centering** of  $\Delta$

$$\mathbf{B} = \left( \mathbf{I} - \frac{\mathbf{J}}{n} \right) \Delta \left( \mathbf{I} - \frac{\mathbf{J}}{n} \right)$$

where

- $\mathbf{I}$  denotes the  $n \times n$  identity matrix
- $\mathbf{J}$  be the  $n \times n$  matrix of all  $\mathbf{1}$ 's

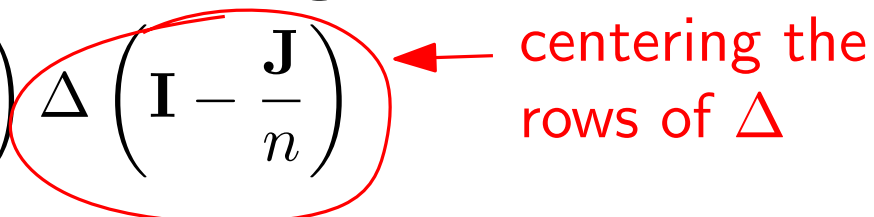
# Multidimensional scaling (Torgerson (1952))

Assume matrix  $\mathbf{A}$  is not available, but instead we are given all squared pairwise distances as  $n \times n$  matrix  $\Delta$

$$\Delta_{ij} = \|\mathbf{a}_i - \mathbf{a}_j\|^2$$

We can recover inner products  $\mathbf{a}_i \mathbf{a}_j^T$  of unknown  $\mathbf{A}$  as follows

The following matrix is a **double-centering** of  $\Delta$

$$\mathbf{B} = \left( \mathbf{I} - \frac{\mathbf{J}}{n} \right) \Delta \left( \mathbf{I} - \frac{\mathbf{J}}{n} \right)$$


where

- $\mathbf{I}$  denotes the  $n \times n$  identity matrix
- $\mathbf{J}$  be the  $n \times n$  matrix of all 1's

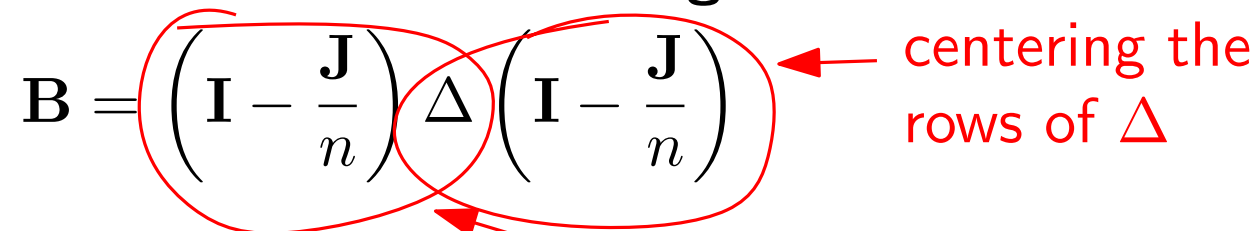
# Multidimensional scaling (Torgerson (1952))

Assume matrix  $\mathbf{A}$  is not available, but instead we are given all squared pairwise distances as  $n \times n$  matrix  $\Delta$

$$\Delta_{ij} = \|\mathbf{a}_i - \mathbf{a}_j\|^2$$

We can recover inner products  $\mathbf{a}_i \mathbf{a}_j^T$  of unknown  $\mathbf{A}$  as follows

The following matrix is a **double-centering** of  $\Delta$

$$\mathbf{B} = \left( \mathbf{I} - \frac{\mathbf{J}}{n} \right) \Delta \left( \mathbf{I} - \frac{\mathbf{J}}{n} \right)$$


where

- $\mathbf{I}$  denotes the  $n \times n$  identity matrix
- $\mathbf{J}$  be the  $n \times n$  matrix of all 1's

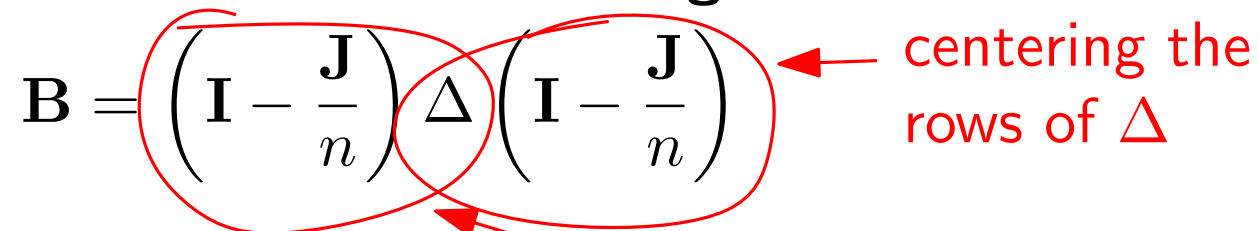
# Multidimensional scaling (Torgerson (1952))

Assume matrix  $\mathbf{A}$  is not available, but instead we are given all squared pairwise distances as  $n \times n$  matrix  $\Delta$

$$\Delta_{ij} = \|\mathbf{a}_i - \mathbf{a}_j\|^2$$

We can recover inner products  $\mathbf{a}_i \mathbf{a}_j^T$  of unknown  $\mathbf{A}$  as follows

The following matrix is a **double-centering** of  $\Delta$

$$\mathbf{B} = \left( \mathbf{I} - \frac{\mathbf{J}}{n} \right) \Delta \left( \mathbf{I} - \frac{\mathbf{J}}{n} \right)$$


where

- $\mathbf{I}$  denotes the  $n \times n$  identity matrix
- $\mathbf{J}$  be the  $n \times n$  matrix of all 1's

centering the  
columns of  $\Delta$

If  $\mathbf{A}$  is mean-centered, one can show that  $(-\frac{1}{2})\mathbf{B} = \mathbf{A}\mathbf{A}^T$

# Multidimensional scaling (Torgerson (1952))

Recall that from SVD we have

$$\mathbf{A}\mathbf{v}_i = \sigma_i \mathbf{u}_i \quad \text{and} \quad \mathbf{A}^T \mathbf{u}_i = \sigma_i \mathbf{v}_i$$

which implies

$$\mathbf{A}^T \mathbf{A} \mathbf{v}_i = \sigma_i^2 \mathbf{v}_i$$

# Multidimensional scaling (Torgerson (1952))

Recall that from SVD we have

$$\mathbf{A}\mathbf{v}_i = \sigma_i \mathbf{u}_i \quad \text{and} \quad \mathbf{A}^T \mathbf{u}_i = \sigma_i \mathbf{v}_i$$

which implies

$$\mathbf{A}^T \mathbf{A} \mathbf{v}_i = \sigma_i^2 \mathbf{v}_i$$

symmetrically, this also implies

$$\mathbf{A} \mathbf{A}^T \mathbf{u}_i = \sigma_i^2 \mathbf{u}_i$$

# Multidimensional scaling (Torgerson (1952))

Recall that from SVD we have

$$\mathbf{A}\mathbf{v}_i = \sigma_i \mathbf{u}_i \quad \text{and} \quad \mathbf{A}^T \mathbf{u}_i = \sigma_i \mathbf{v}_i$$

which implies

$$\mathbf{A}^T \mathbf{A} \mathbf{v}_i = \sigma_i^2 \mathbf{v}_i$$

symmetrically, this also implies

$$\mathbf{A} \mathbf{A}^T \mathbf{u}_i = \sigma_i^2 \mathbf{u}_i$$

Thus, the eigenvectors of  $\mathbf{A} \mathbf{A}^T$  are the vectors  $\mathbf{u}_i$  of the SVD of  $\mathbf{A}$

# Multidimensional scaling (Torgerson (1952))

Recall that from SVD we have

$$\mathbf{A}\mathbf{v}_i = \sigma_i \mathbf{u}_i \quad \text{and} \quad \mathbf{A}^T \mathbf{u}_i = \sigma_i \mathbf{v}_i$$

which implies

$$\mathbf{A}^T \mathbf{A} \mathbf{v}_i = \sigma_i^2 \mathbf{v}_i$$

symmetrically, this also implies

$$\mathbf{A} \mathbf{A}^T \mathbf{u}_i = \sigma_i^2 \mathbf{u}_i$$

Thus, the eigenvectors of  $\mathbf{A} \mathbf{A}^T$  are the vectors  $\mathbf{u}_i$  of the SVD of  $\mathbf{A}$

Using this, we can compute the SVD of  $\mathbf{A}$  and perform PCA



# Multidimensional scaling (Torgerson (1952))

Recall that from SVD we have

$$\mathbf{A}\mathbf{v}_i = \sigma_i \mathbf{u}_i \quad \text{and} \quad \mathbf{A}^T \mathbf{u}_i = \sigma_i \mathbf{v}_i$$

which implies

$$\mathbf{A}^T \mathbf{A} \mathbf{v}_i = \sigma_i^2 \mathbf{v}_i$$

symmetrically, this also implies

$$\mathbf{A} \mathbf{A}^T \mathbf{u}_i = \sigma_i^2 \mathbf{u}_i$$

Thus, the eigenvectors of  $\mathbf{A} \mathbf{A}^T$  are the vectors  $\mathbf{u}_i$  of the SVD of  $\mathbf{A}$

Using this, we can compute the SVD of  $\mathbf{A}$  and perform PCA

The result is called an **embedding** of  $\mathbf{A}$  and the process is called classical multidimensional scaling (MDS).

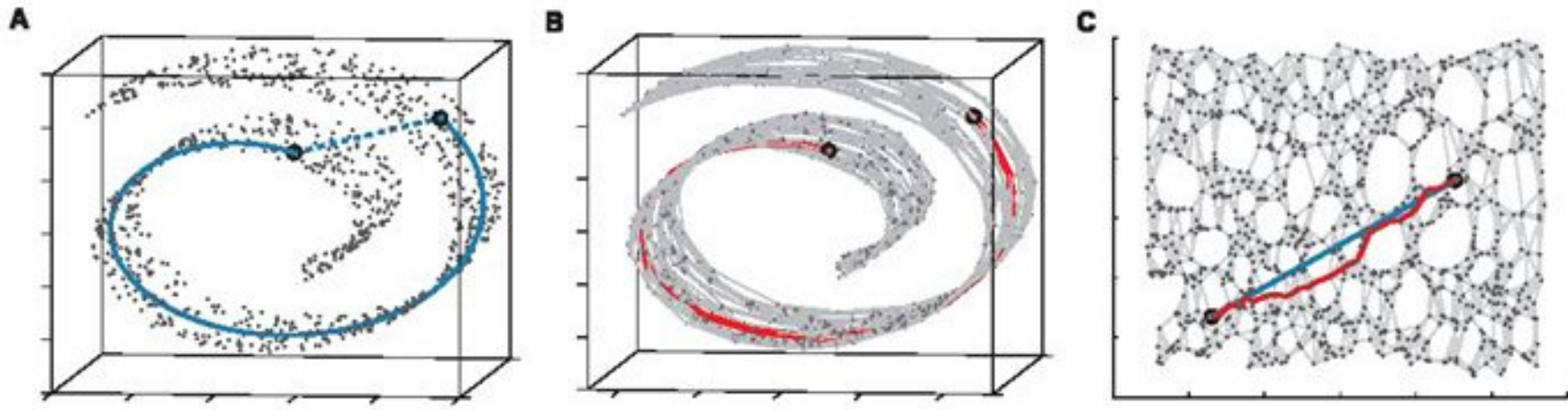
# Isomap

Isomap is a non-linear embedding algorithm which assumes that the data lies on an Euclidean manifold

Isomap is due to Tenenbaum, Silva and Langford (2000)

## Algorithm:

- Compute the  $k$ -nearest neighbor graph  $G$
- Compute all pairwise shortest paths in  $G$
- Use Multidimensional scaling on the obtained distances



# Summary

- Principal Component Analysis (PCA)
- Interpretation of Principal Components
- Computing Principal Components
- Singular-Value Decomposition (SVD)
- Power Method
- Eigenvectors of the Sample Covariance Matrix
- Multidimensional scaling
- Isomap

# References

- Avrim Blum, John Hopcroft, Ravindran Khannan: *Foundations of Data Science*
- Trevor Hastie, Robert Tibshirani, Jerome Friedman: *Elements of Statistical Learning*
- J. B. Tenenbaum, V. de Silva, J. C. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction", *Science* 290, (2000).