Geometric Data Analysis

Principal Component Analysis

MAT 6480W / STT 6705V

Guy Wolf
guy.wolf@umontreal.ca

Université de Montréal
Fall 2019

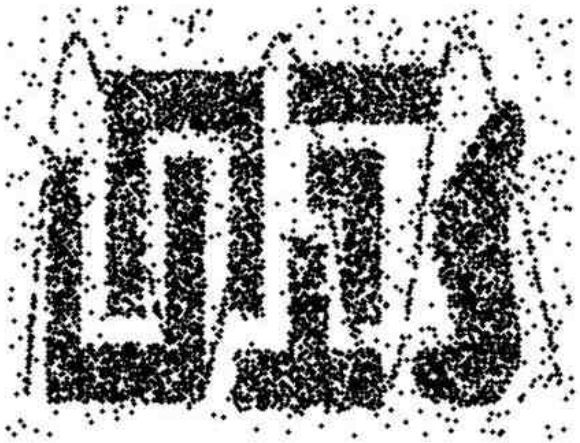
Outline

1. Preprocessing for data simplification
   - Sampling
   - Aggregation
   - Discretization
   - Density estimation
   - Dimensionality reduction
2. Principal component analysis (PCA)
   - Autoencoder
   - Variance maximization
   - Singular value decomposition (SVD)

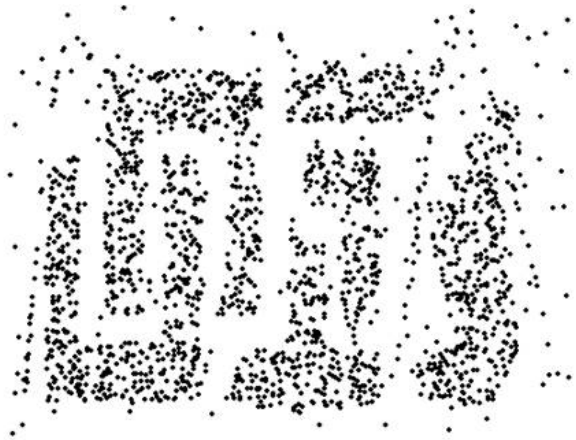
Sampling

Select a subset of representative data points instead of processing the entire data.

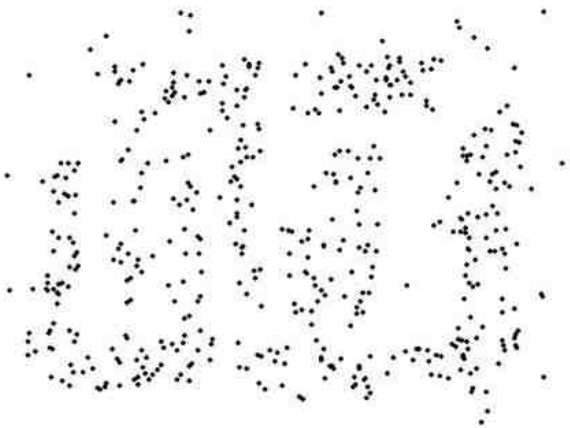
A sampled subset is useful only if its analysis yields the same patterns, results, conclusions, etc., as the analysis of the entire data.



8000 points



2000 points



500 points

Sampling

Select a subset of representative data points instead of processing the entire data.

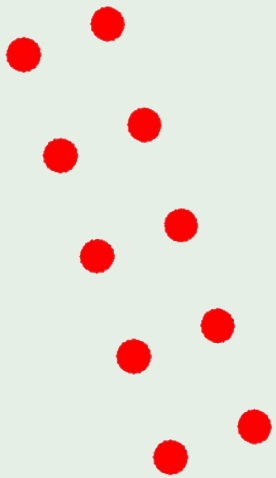
Common sampling approaches

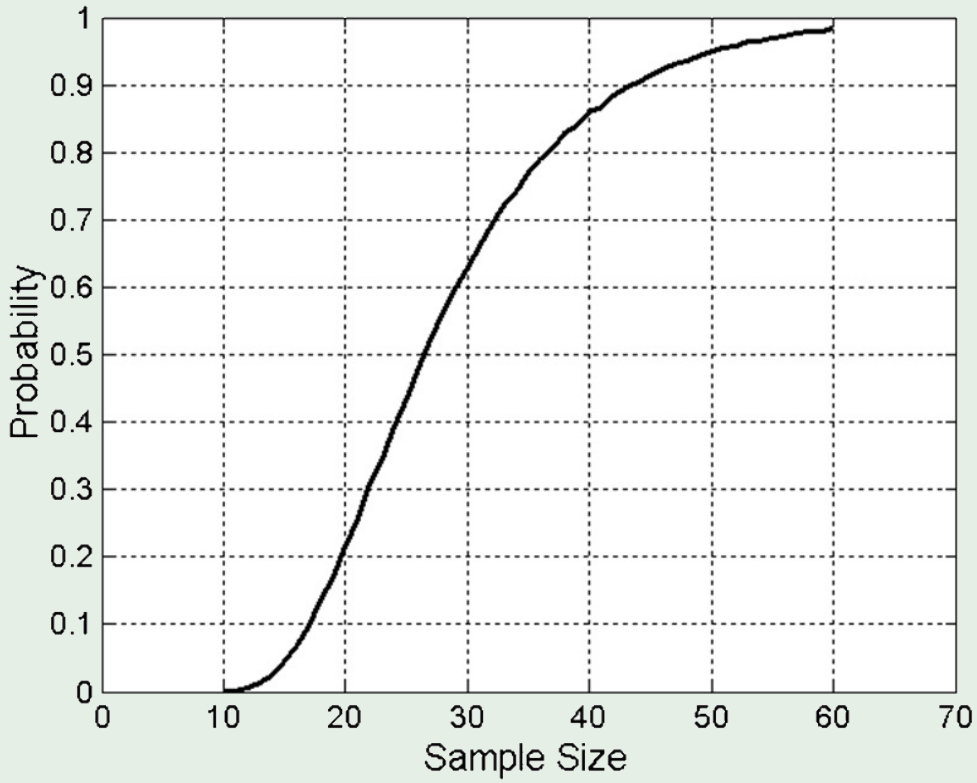
- Random: an equal probability of selecting any particular item.
- Without replacement: iteratively selected & remove items.
- With replacement: selected items remain in the population.
- Stratified: draw random samples from each partition.

Choosing a sufficient sample size is often crucial for effective sampling.

Example

Choose enough samples guarantee at least one representative is selected from each distinct group/cluster/profile in the data.





Preprocessing for data simplification

Aggregation

Instead of sampling representative data points we can coarse-grain the data by aggregating together attributes or data points.

Aggregation

Combining several attributes to a single feature, or several data points into a single observation.

Examples

- Change monthly revenues to annual revenues
- Analyze neighborhoods instead of houses
- Provide average rating of a season (not per episode)

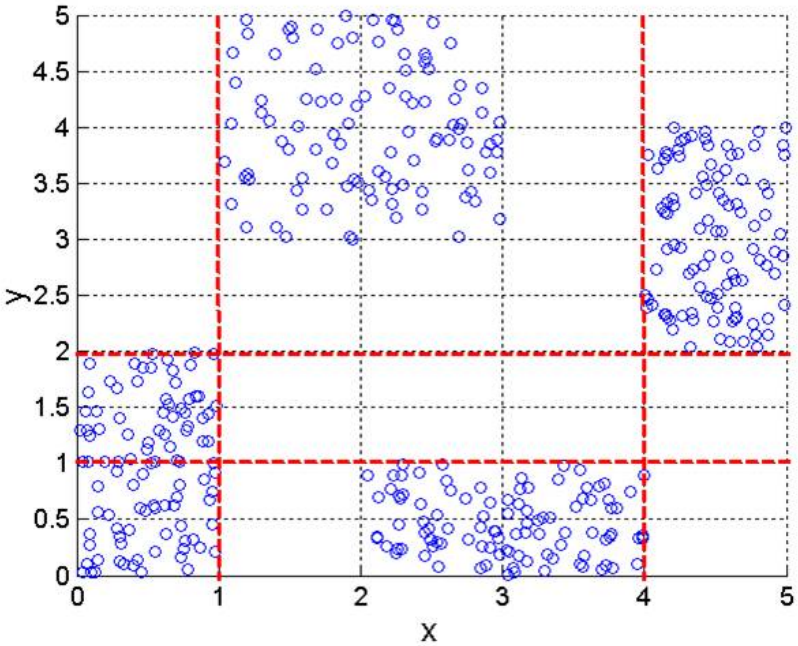
It is sometimes convenient to transform the entire data to nominal (or ordinal) attributes.

Discretization

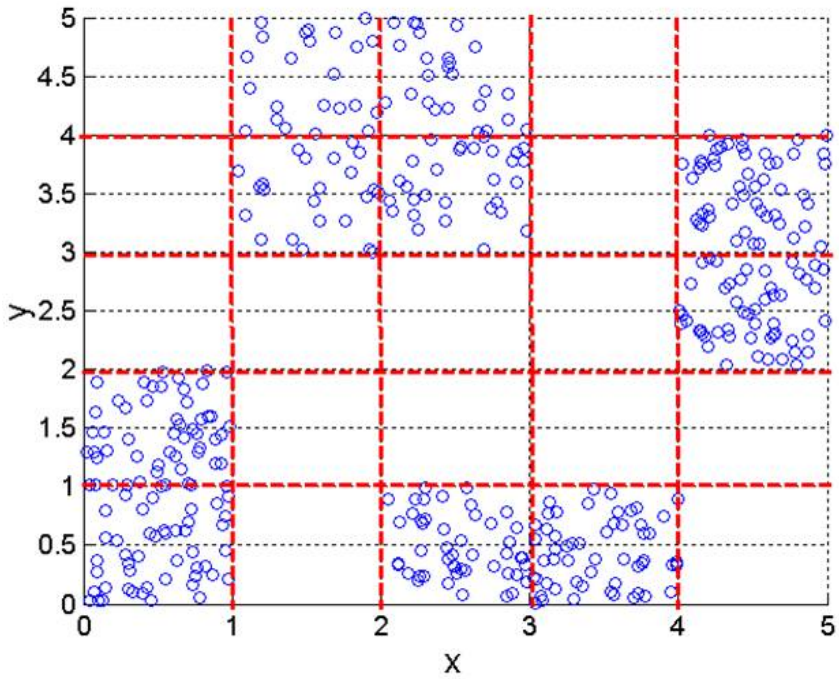
Transformation of continuous attributes (or ones with infinite range) to discrete ones with a finite range.

Discretization can be done in a supervised discretization (e.g., using class labels) or in an unsupervised manner (e.g., using clustering).

Supervised discretization based on minimizing impurity:



3 values per axis

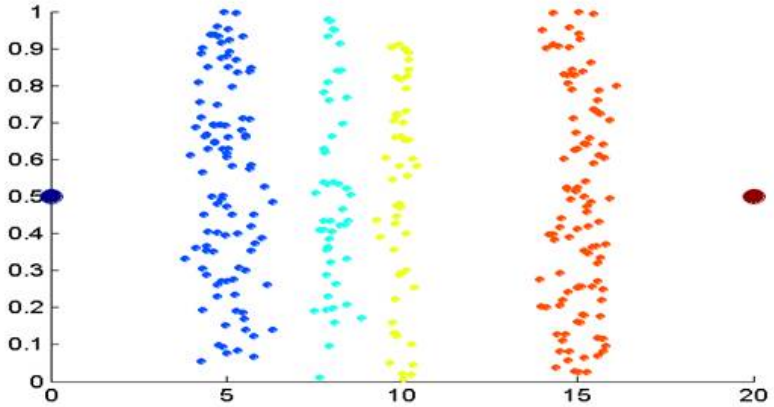


5 values per axis

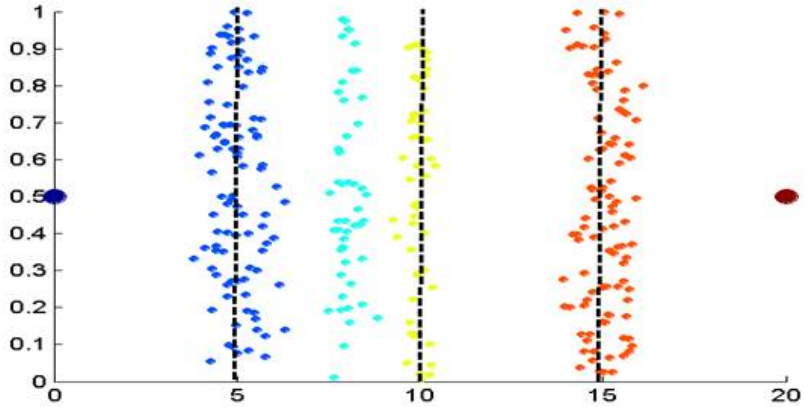
Preprocessing for data simplification

Discretization

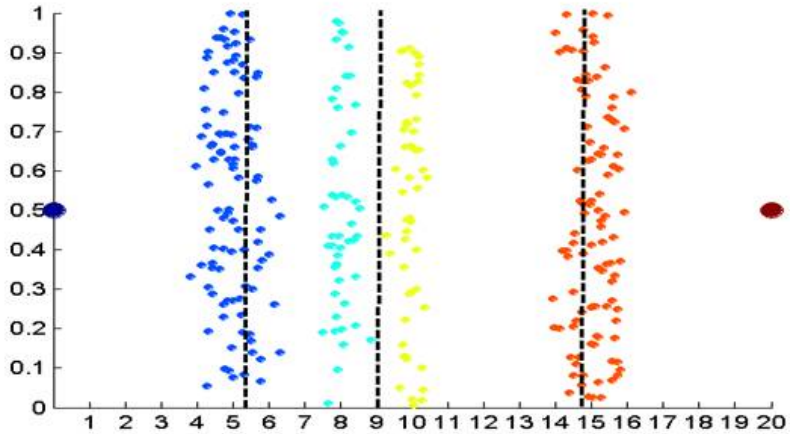
Unsupervised discretization:



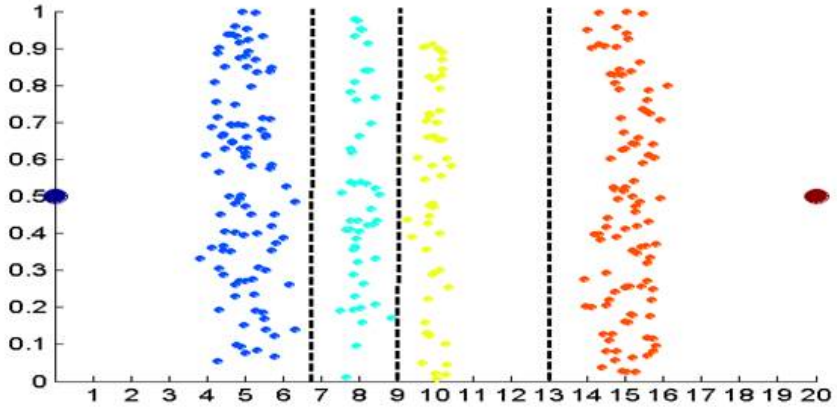
Data



Equal interval width



Equal frequency



K-means

Preprocessing for data simplification

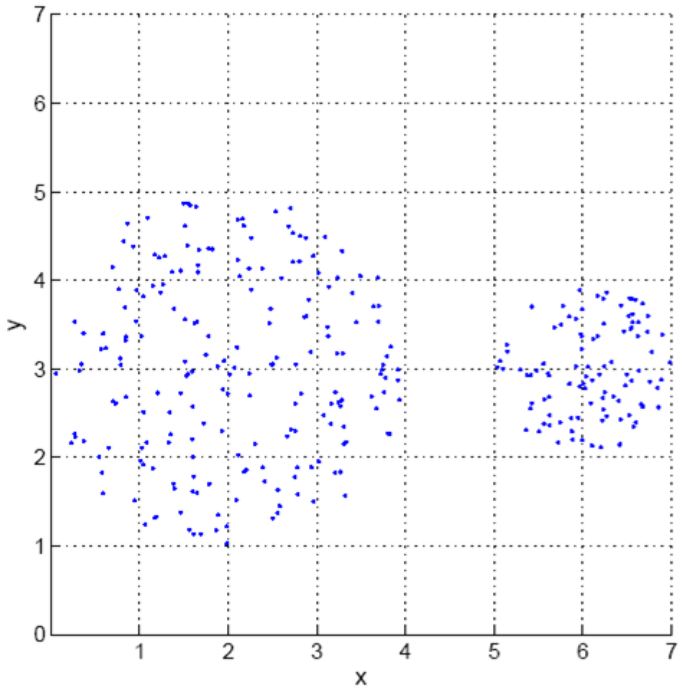
Density estimation

Transforming attributes from raw vales to densities can be used to coarse-grain the data and bring its features to comparable scales between zero and one.

Preprocessing for data simplification

Density estimation

Transforming attributes from raw vales to densities can be used to coarse-grain the data and bring its features to comparable scales between zero and one.



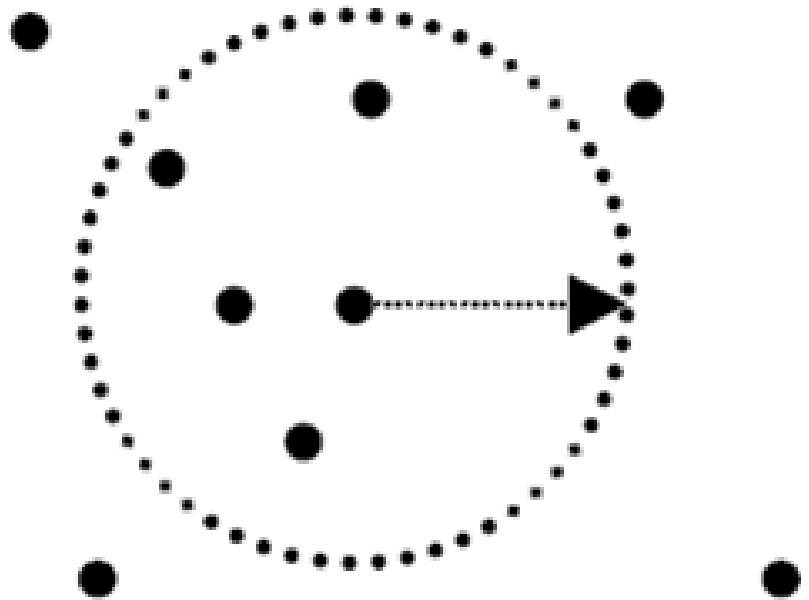
0	0	0	0	0	0	0
0	0	0	0	0	0	0
4	17	18	6	0	0	0
14	14	13	13	0	18	27
11	18	10	21	0	24	31
3	20	14	4	0	0	0
0	0	0	0	0	0	0

Cell-based density estimation

Preprocessing for data simplification

Density estimation

Transforming attributes from raw vales to densities can be used to coarse-grain the data and bring its features to comparable scales between zero and one.



Center-based density estimation

Preprocessing for data simplification

Dimensionality reduction

Dimensionality of data is generally determined by the number of attributes or features that represent each data point.

Curse of dimensionality

A general term for various phenomena that arise when analyzing and processing high-dimensional data.

- Common theme - statistical significance is difficult, impractical, or even impossible to obtain due to sparsity of the data in high-dimensions
- Causes poor performance of classical statistical methods compared to low-dimensional data

Common solution - reduce the dimensionality of the data as part of its (pre)processing.

Preprocessing for data simplification

Dimensionality reduction

There are several approaches to represent the data in a lower dimension, which can generally be split into two types:

Dimensionality reduction approaches

- Feature selection/weighting - select a subset of existing features and only use them in the analysis, while possibly also assigning them importance weights to eliminate redundant information
- Feature extraction/construction - create new features by extracting relevant information from the original features

PCA and MDS are two of the most common dimensionality reduction methods in data analysis, but many others exist as well.

Preprocessing for data simplification

Feature subset selection

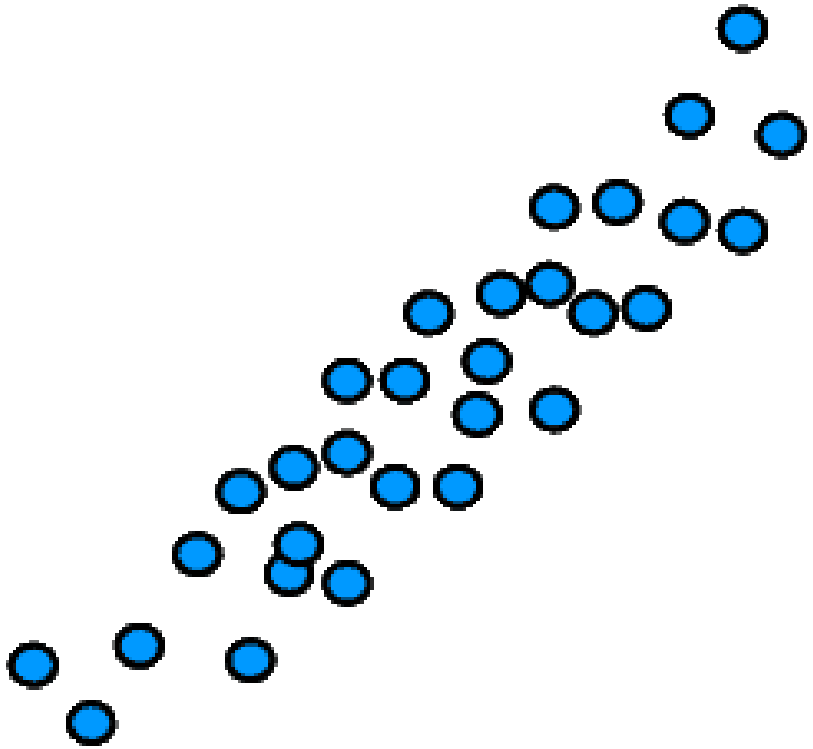
Ideally - choose the best feature subset out of all possible combinations. Impractical - there are 2^n choices for n attributes!

Feature selection approaches

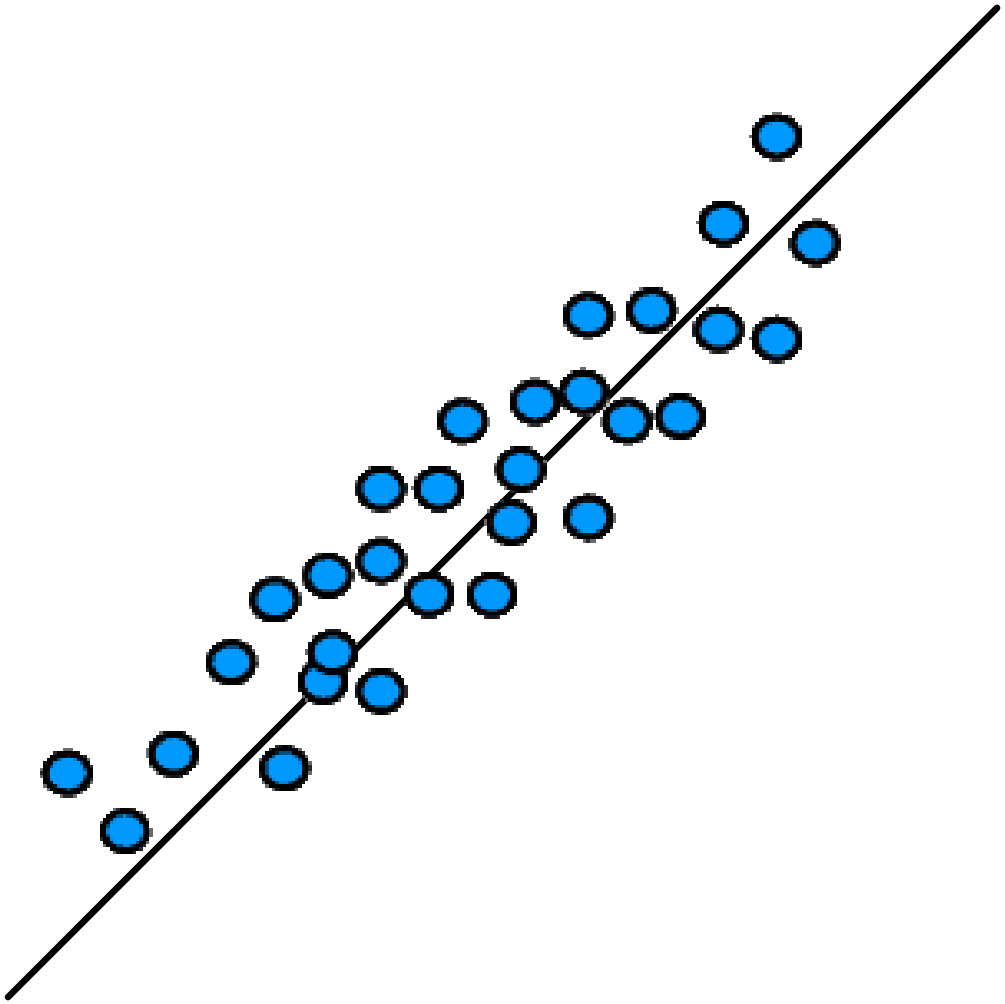
- Embedded methods - choose the best features for a task as part of the data mining algorithm (e.g., decision trees).
- Filter methods - choose features that optimize a general criterion (e.g., min correlation) as part of data preprocessing using an efficient search algorithm.
- Wrapper methods - first formulate & handle a data mining task to select features, and then use the resulting subset to solve the real task.

Alternatively, expert knowledge can sometimes be used to eliminate redundant and unnecessary features.

Principal component analysis



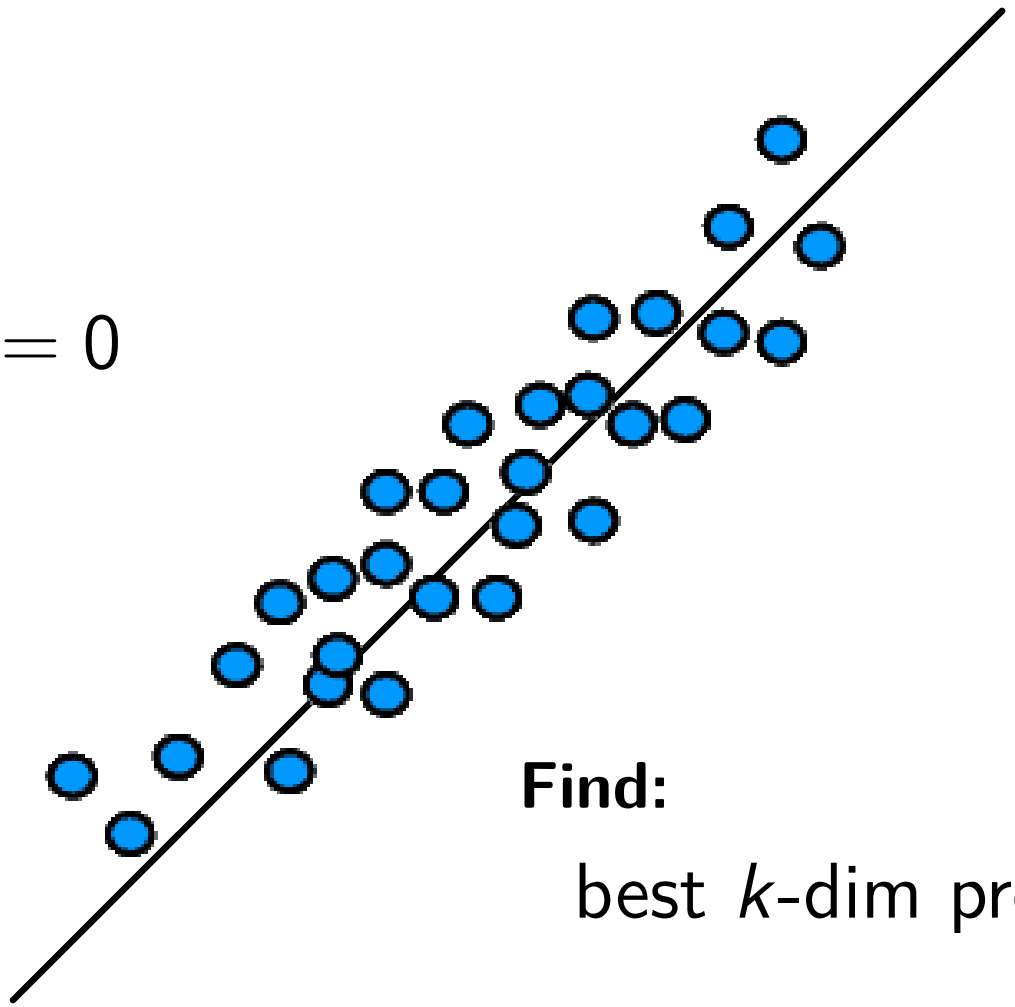
Principal component analysis



Principal component analysis

Assume:

$avg = 0$

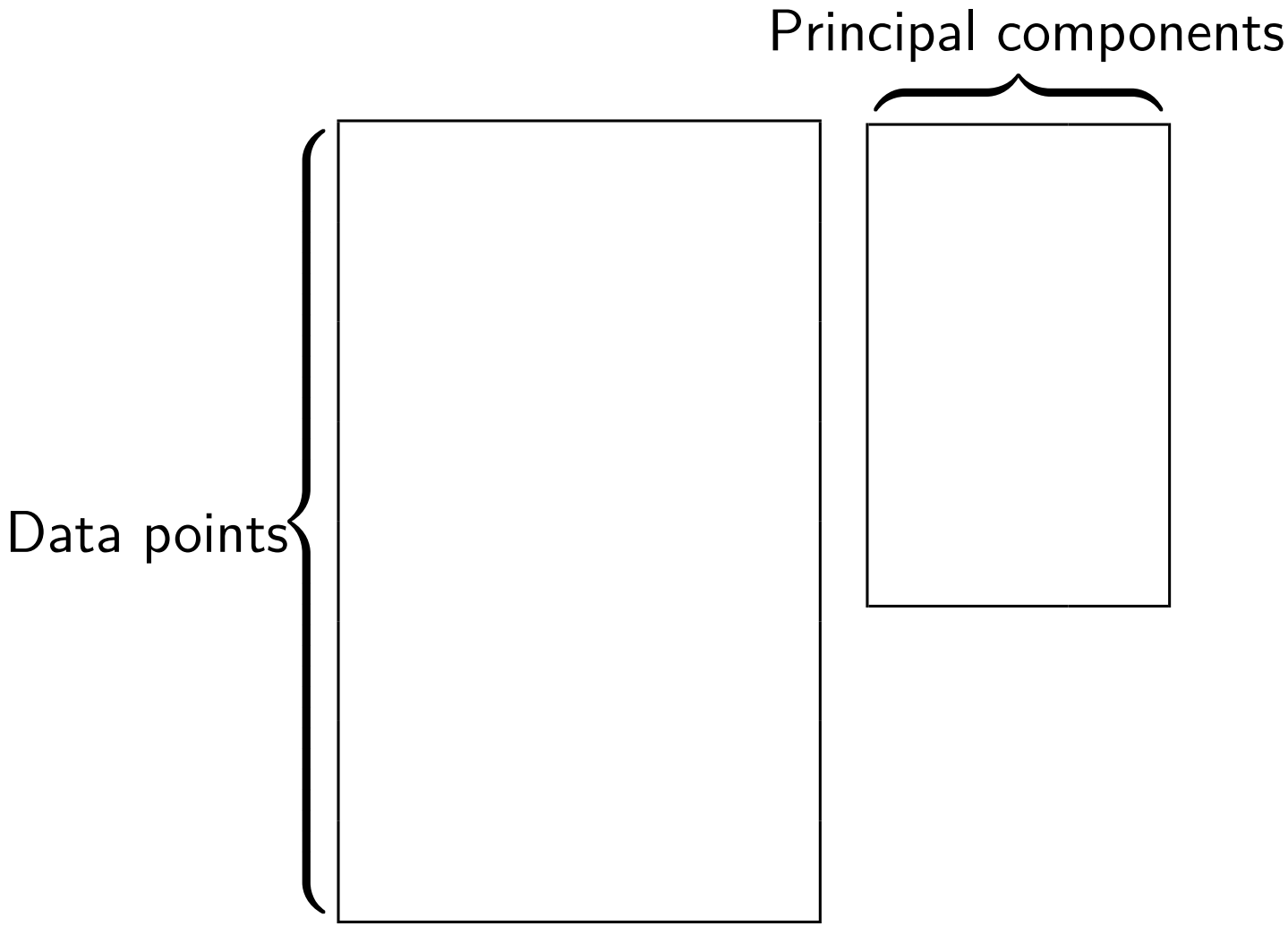


Find:

best k-dim projection

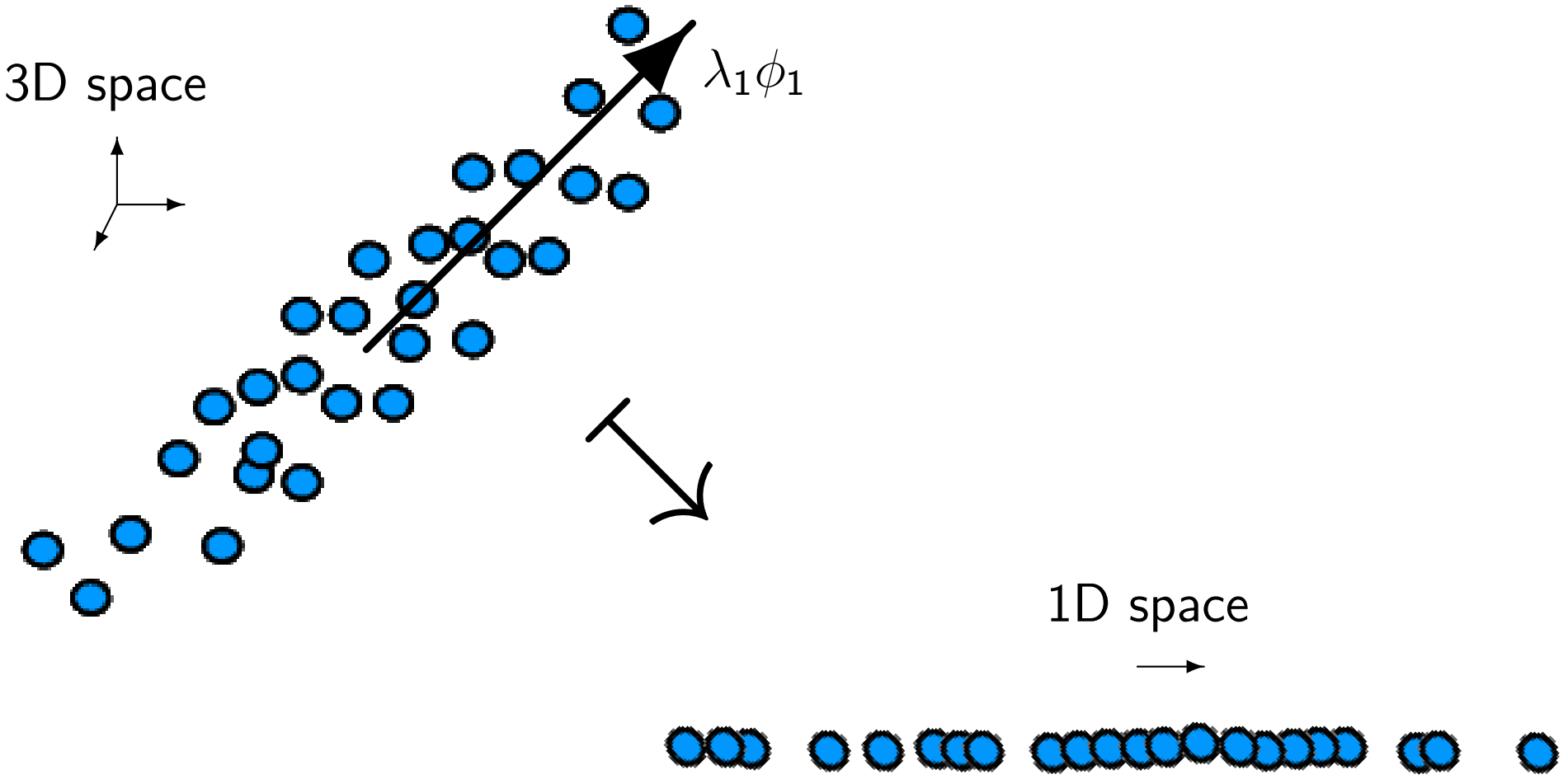
Principal component analysis

Projection on principal components:



Principal component analysis

Projection on principal components:



Principal component analysis

What is the best projection?

Find subspace $S \subseteq \mathbb{R}^n$ s.t. $\dim(S) = k$ and the data is well approximated by $\hat{x} = \text{proj}_S\, x$.

Principal component analysis

What is the best projection?

Find subspace $S \subseteq \mathbb{R}^n$ s.t. $\dim(S) = k$ and the data is well approximated by $\hat{x} = \text{proj}_S\, x$.

$$\Downarrow$$

Find subspace $S \subseteq \mathbb{R}^n$ s.t. $S = \text{span}\{u_1, \ldots, u_k\}$ and the data is $\|x - \hat{x}\|$ is minimal over the data with $\hat{x} = \text{proj}_S\, x$.

Principal component analysis

What is the best projection?

Find subspace $S \subseteq \mathbb{R}^n$ s.t. $\dim(S) = k$ and the data is well approximated by $\hat{x} = \text{proj}_S x$.

$$\Downarrow$$

Find subspace $S \subseteq \mathbb{R}^n$ s.t. $S = \text{span}\{u_1, \ldots, u_k\}$ and the data is $\|x - \hat{x}\|$ is minimal over the data with $\hat{x} = \text{proj}_S x$.

$$\Downarrow$$

Find k vectors $u_1, \ldots, u_k$ s.t. $N^{-1} \sum_{i=1}^{N} \|x_i - \hat{x}_i\|^2$ is minimal with $\hat{x} = \text{proj}_{\text{span}\{u_1,\ldots,u_k\}} x$.

Principal component analysis

What is the best projection?

Find subspace $S \subseteq \mathbb{R}^n$ s.t. $\dim(S) = k$ and the data is well approximated by $\hat{x} = \text{proj}_S x$.

$$\Downarrow$$

Find subspace $S \subseteq \mathbb{R}^n$ s.t. $S = \text{span}\{u_1, \ldots, u_k\}$ and the data is $\|x - \hat{x}\|$ is minimal over the data with $\hat{x} = \text{proj}_S x$.

$$\Downarrow$$

Find k vectors $u_1, \ldots, u_k$ s.t. $N^{-1} \sum_{i=1}^{N} \|x_i - \hat{x}_i\|^2$ is minimal with $\hat{x} = \text{proj}_{\text{span}\{u_1,\ldots,u_k\}} x$.

How do we find these vectors $u_1, \ldots, u_k$?

Principal component analysis

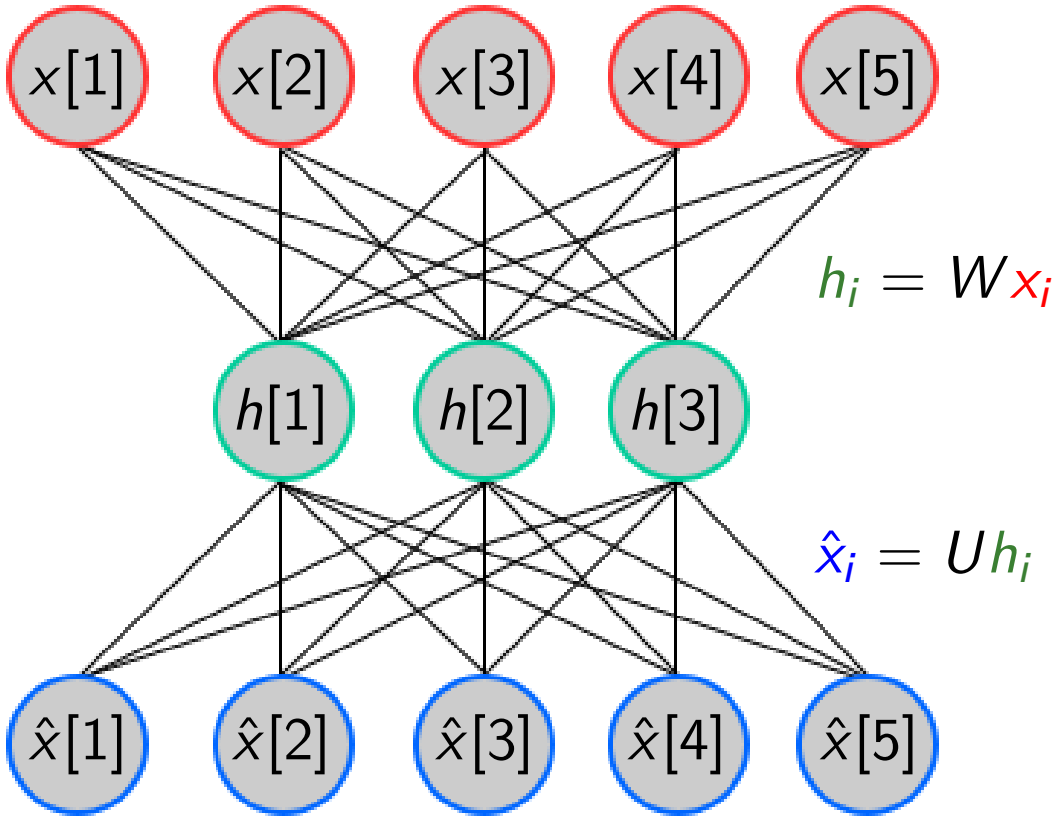
Autoencoder

Minimize $N^{-1}\sum_{i=1}^{n}\|x_i-\hat{x}_i\|^2$ s.t. $\hat{x}=\text{proj}_{\text{span}\{u_1,\ldots,u_k\}}x$

Input layer:

Hidden layer:

Output layer:



$$\underset{W\in\mathbb{R}^{k\times n},U\in\mathbb{R}^{n\times k}}{\arg\min}\sum_{i=1}^{N}\|x_i-UWx_i\|^2$$

Principal component analysis

Reconstruction error minimization

We only need to consider orthonormal vectors $u_1, \ldots, u_k$ (i.e., $\|u_i\| = 1$, $\langle u_i, u_j \rangle = 0$ for $i \neq j$) that form a basis for the subspace. We can then extend this set to form a basis $u_1, \ldots, u_n$ for the entire $\mathbb{R}^n$.

Then, we can write $x = \sum_{j=1}^{n} \langle x, u_j \rangle u_j = \sum_{j=1}^{n} u_j u_j^T x$ and $\text{proj}_{\text{span}\{u_1,\ldots,u_k\}} x = \sum_{j=1}^{k} u_j u_j^T x$.

We now consider the reconstruction error $N^{-1} \sum_{i=1}^{N} \|x_i - \hat{x}_i\|^2$.

Principal component analysis

Reconstruction error minimization

First, notice that $x - \hat{x} = \sum_{j=1}^{n} u_j u_j^T x - \sum_{j=1}^{k} u_j u_j^T x = \sum_{j=k+1}^{n} u_j u_j^T x$

$$\Downarrow$$

$$\|x - \hat{x}\|^2 = \sum_{q=1}^{n} \Big(\sum_{j=k+1}^{n} u_j[q] u_j^T x \Big)^2$$

$$= \sum_{j=k+1}^{n} \sum_{j'=k+1}^{n} \Big(\sum_{q=1}^{n} u_j[q] u_{j'}[q] \Big) (u_j^T x)(u_{j'}^T x)$$

$$= \sum_{j=k+1}^{n} (u_j^T x)^2 = \sum_{j=1}^{n} (u_j^T x)^2 - \sum_{j=1}^{k} (u_j^T x)^2 = \|x\|^2 - \|\hat{x}\|^2$$

$$\Downarrow$$

Minimizing the reconstruction error is equivalent to maximizing $N^{-1} \sum_{i=1}^{N} \|\hat{x}_i\|^2 = \sum_{j=1}^{k} N^{-1} \sum_{i=1}^{N} (u_j^T x_i)^2 = \sum_{j=1}^{k} \text{variance}(u_j^T x)$

Principal component analysis

Variance maximization

Find a direction that maximizes the variance in the projected data.

Principal component analysis

Variance maximization

Find a direction that maximizes the variance in the projected data.

$$\Downarrow$$

Find a unit vector $u \in \mathbb{R}^n$ that maximizes variance$(u^T x) = u^T \Sigma u$, where Σ is the covariance matrix.

Principal component analysis

Variance maximization

Find a direction that maximizes the variance in the projected data.

$$\Downarrow$$

Find a unit vector $u \in \mathbb{R}^n$ that maximizes:

$$\text{variance}(u^T x) = N^{-1} \sum_{i=1}^{N} (u^T x_i)^2 = N^{-1} \sum_{i=1}^{N} (u^T x_i)(x_i^T u)$$

$$= u^T \left(N^{-1} \sum_{i=1}^{N} x_i x_i^T \right) u = u^T \Sigma u$$

where Σ is the covariance matrix.

Principal component analysis

Variance maximization

Find a direction that maximizes the variance in the projected data.

$$\Downarrow$$

Find a unit vector $u \in \mathbb{R}^n$ that maximizes variance$(u^T x) = u^T \Sigma u$, where Σ is the covariance matrix.

Principal component analysis

Variance maximization

Find a direction that maximizes the variance in the projected data.

$$\Downarrow$$

Find a unit vector $u \in \mathbb{R}^n$ that maximizes variance$(u^T x) = u^T \Sigma u$, where Σ is the covariance matrix.

$$\Downarrow$$

Solve the maximization problem:

$$\begin{array}{ll} \text{maximize} & u^T \Sigma u \\ \text{s.t.} & \|u\| = 1 \end{array}$$

Principal component analysis

Variance maximization

Solve the maximization problem:

$$\begin{aligned} \text{maximize} \quad & u^T \Sigma u \\ \text{s.t.} \quad & \|u\| = 1 \end{aligned}$$

Apply Lagrange multipliers method:

$$\begin{aligned} f(u, \alpha) &= u^T \Sigma u + \alpha(1 - u^T u) \\ \nabla_u f(u, \alpha) &= 2(\Sigma u - \alpha u) \\ \nabla_u f(u, \alpha) = 0 &\Rightarrow \Sigma u = \alpha u \end{aligned}$$

Therefore, u is an eigenvector of Σ with eigenvalue α, which has to be the maximal eigenvalue to maximize $u^T \Sigma u = \alpha$.

Principal component analysis

Variance maximization

Similarly, a second direction is found via:

$$\begin{aligned} \text{maximize} \quad & u_2^T \Sigma u_2 \\ \text{s.t.} \quad & \|u_2\| = 1 \\ & \langle u_2, u_1 \rangle = 0 \end{aligned}$$

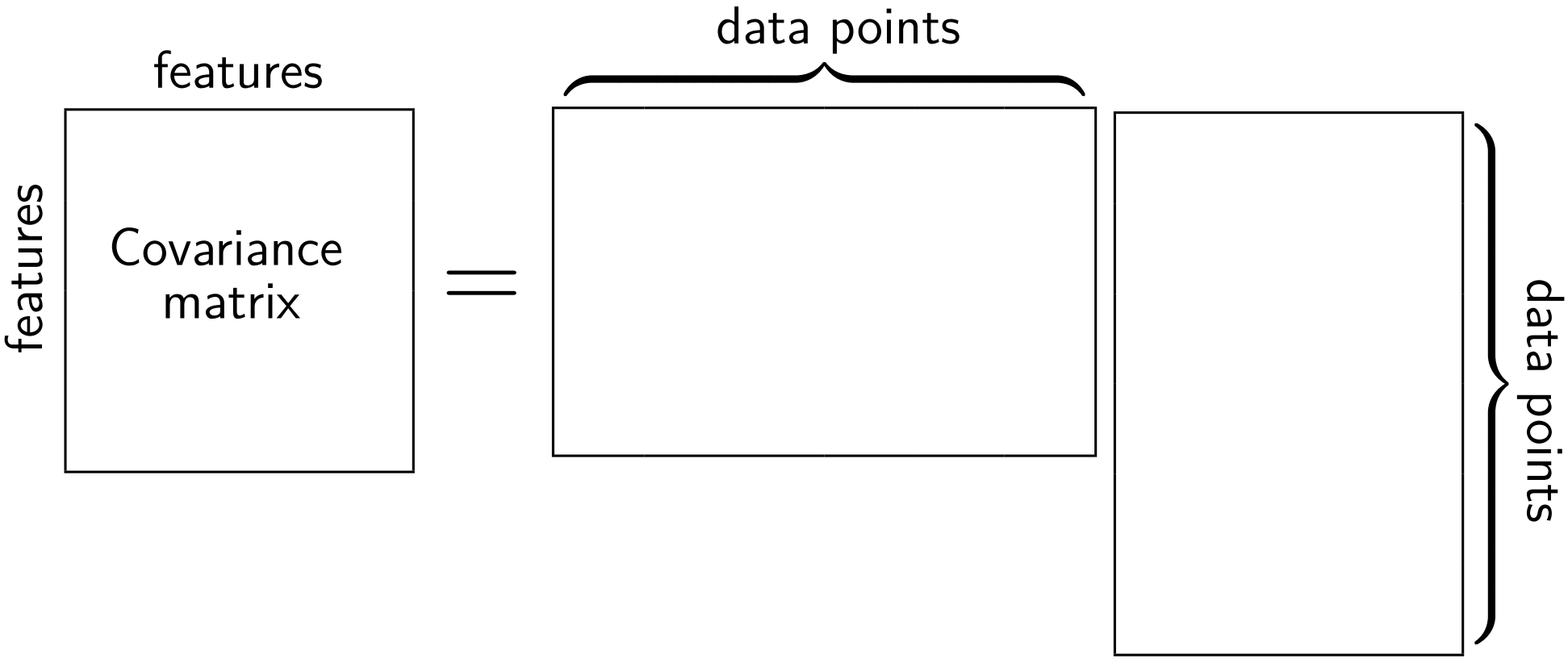
Apply Lagrange multipliers method:

$$\begin{aligned} f(u_2, \alpha, \beta) &= u_2^T \Sigma u_2 + \alpha(1 - u_2^T u_2) - \beta u_2^T u_1 \\ \nabla_{u_2} f(u_2, \alpha, \beta) &= 2(\Sigma u_2 - \alpha u_2) - \beta u_1 \\ \nabla_{u_2} f(u_2, \alpha, \beta) = 0 &\Rightarrow \beta = -\langle u_1, \nabla_{u_2} f(u_2, \alpha, \beta) \rangle = 0 \\ &\Rightarrow \Sigma u_2 = \alpha u_2 \end{aligned}$$

Therefore, u_2 is an eigenvector of Σ with the second largest eigenvalue.

Principal component analysis

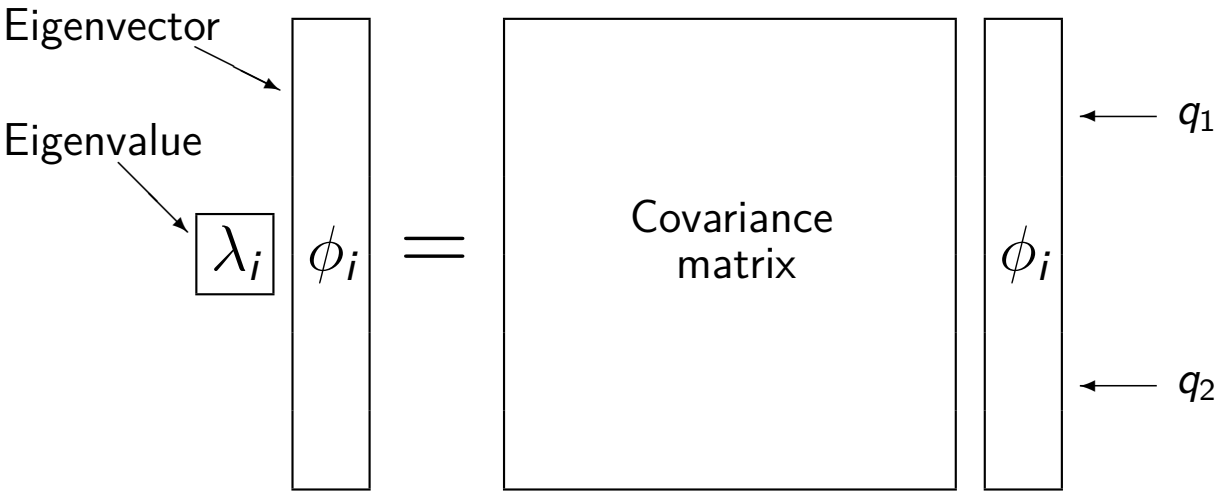
Eigendecomposition and SVD



$$cov(q_1, q_2) \triangleq \sum_i x_i[q_1] \cdot x_i[q_2]$$

Principal component analysis

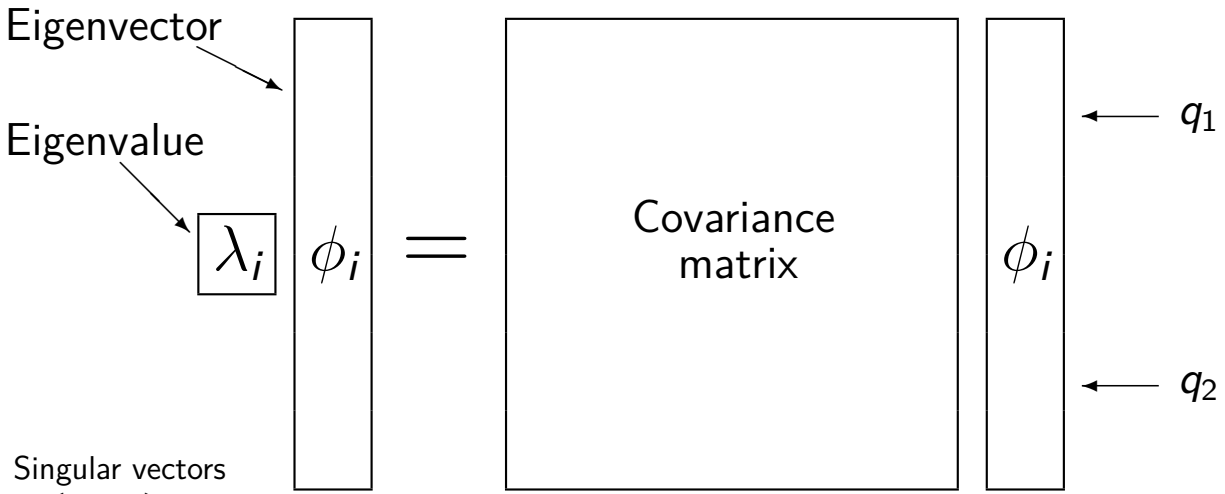
Eigendecomposition and SVD

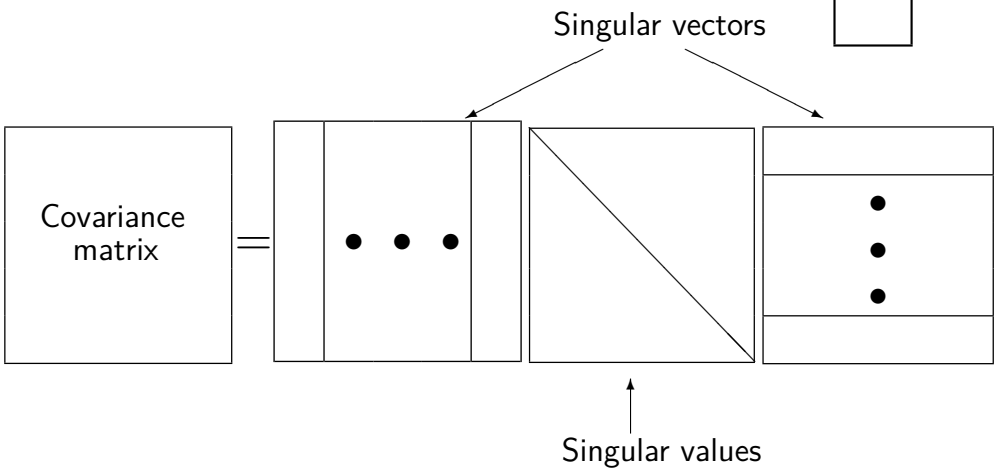


Principal component analysis

Eigendecomposition and SVD

Spectral theorem applies to cov. matrices:





SVD (Singular Value Decomposition)

Spectral Theorem: $cov(q_1, q_2) = \sum_i \lambda_i \, \phi_i[q_1] \, \phi_i[q_2]$

Principal component analysis

Singular value decomposition

Any matrix $M \in \mathbb{R}^{n \times k}$ can be decomposed to $U, S, V \leftarrow\!\shortmid \text{SVD}(M)$ as

$$M = U\,S\,V^T$$

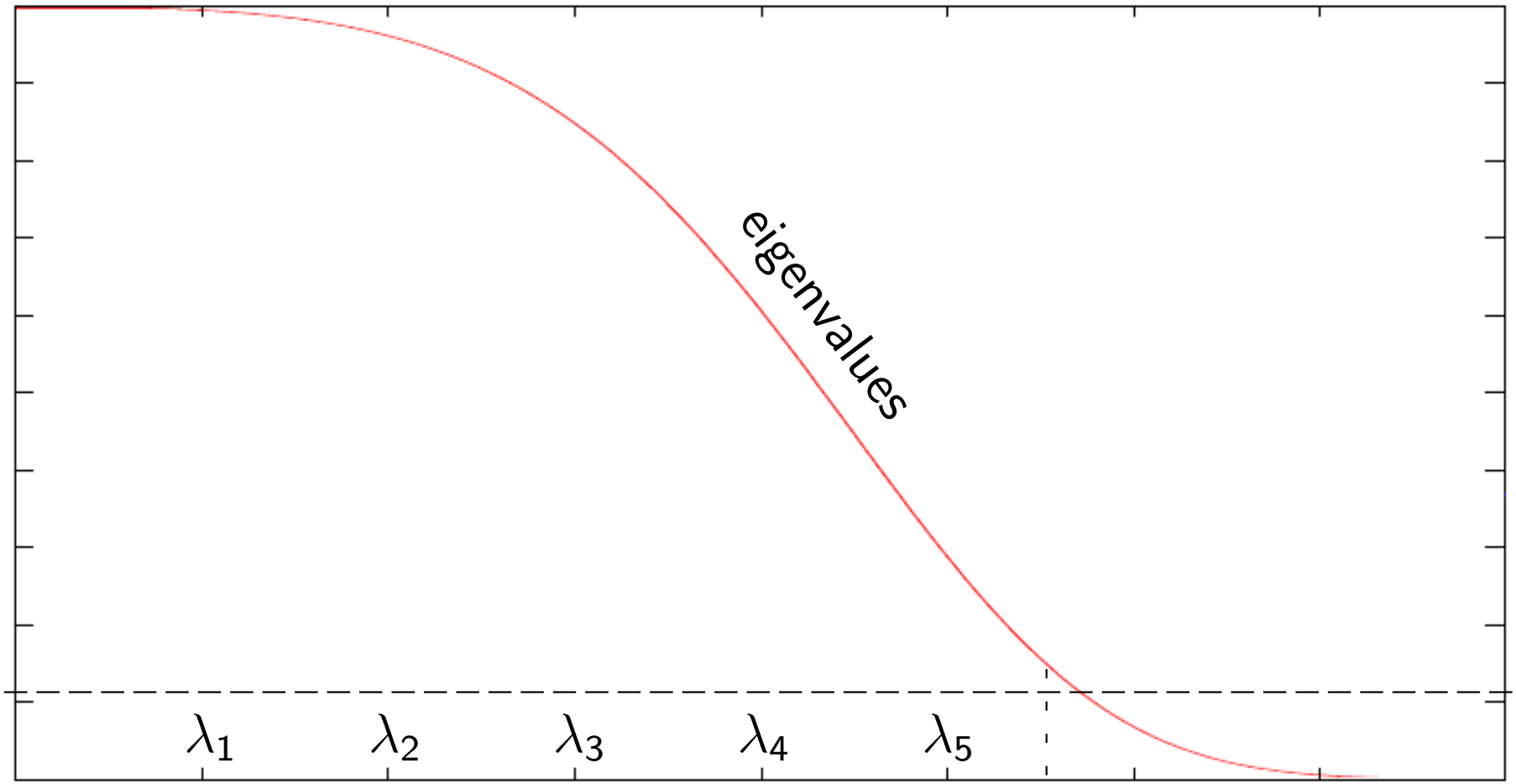
U: $n \times n$ orthogonal; S: $n \times k$ diagonal; V: $k \times k$ orthogonal

- The singular values in S are the square root of the (nonnegative) eigenvalues of both MM^T and M^TM.
- The singular vectors in (the columns of) U are the eigenvectors of MM^T.
- The singular vectors in (the columns of) V are the eigenvectors of M^TM.

Proof and more details about SVD can be found on Wikipedia.

Principal component analysis

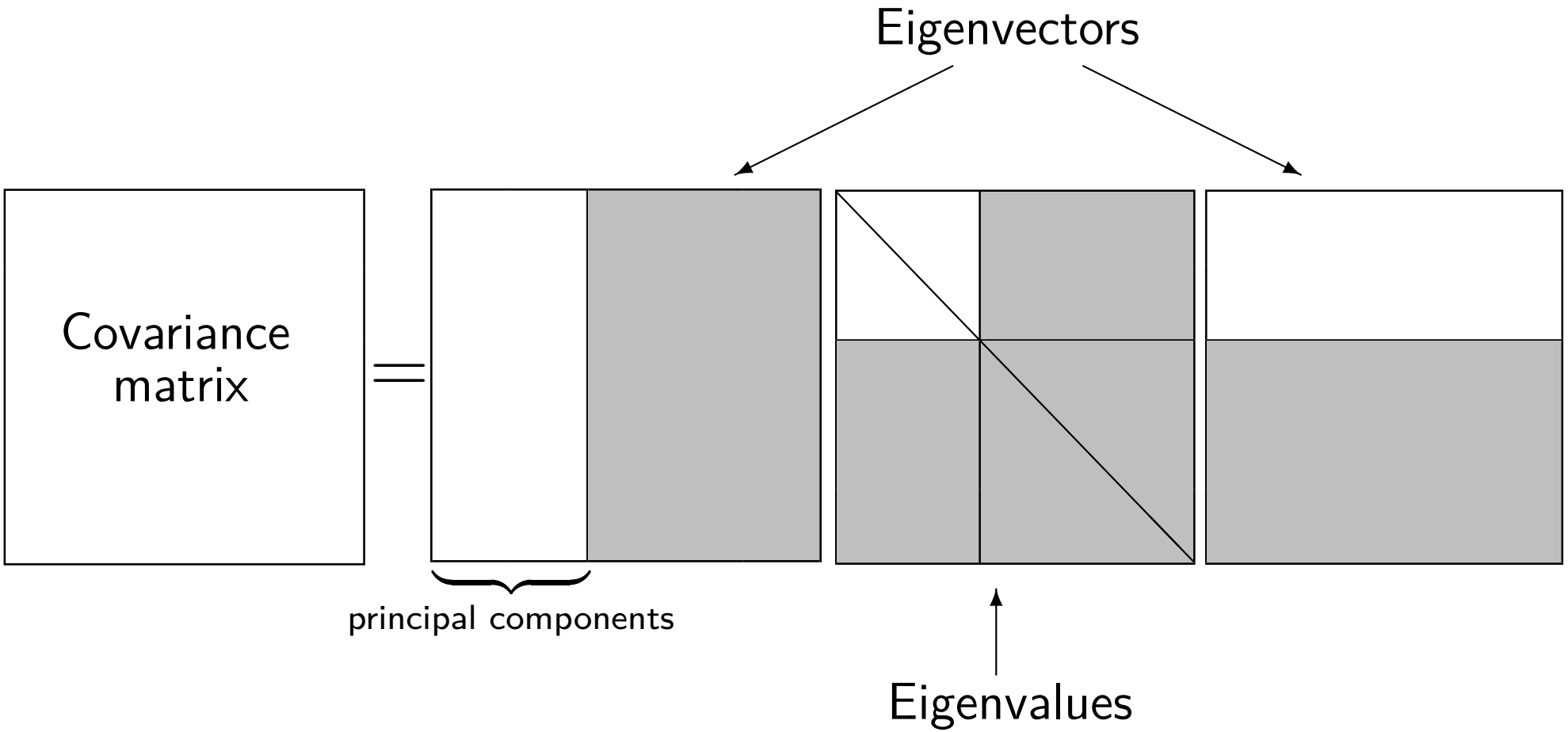
Singular value decomposition



Decaying covariance spectrum reveals (low) dimensionality

Principal component analysis

Singular value decomposition



Covariance matrix can be approximated by a truncated SVD

Principal component analysis

Trivial example

Consider simple case of data points that are all on the same high dimensional line

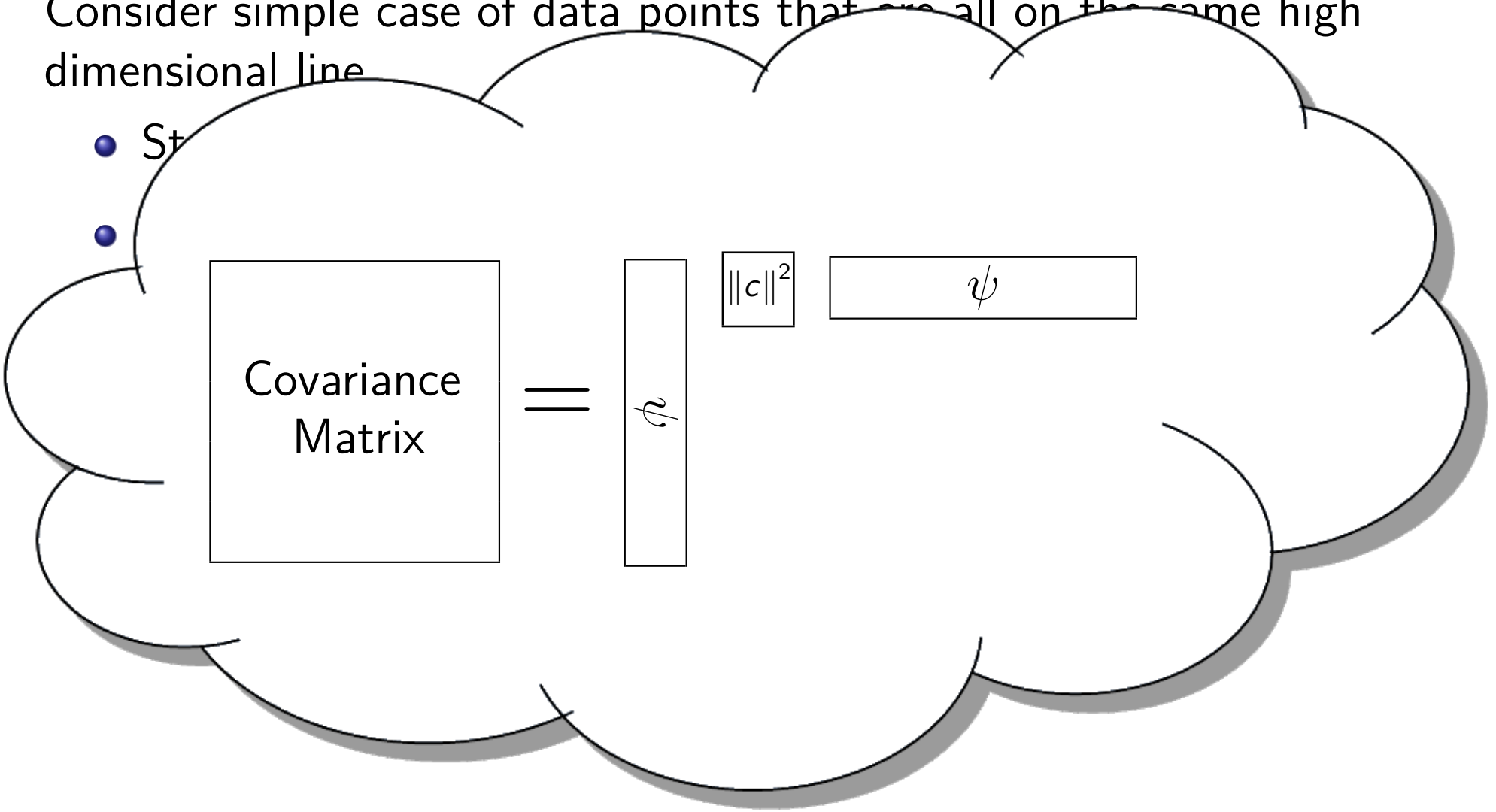
- Straight line is defined by a unit vector $\left\|\vec{\psi}\right\| = 1$
- Points on the line are defined by multiplying $\vec{\psi}$ by scalars
- The points can be formulated as $x_i = c_i\vec{\psi}$
- Covariance: $\text{cov}(t_1, t_2) = \sum_i x_i[t_1]x_i[t_2] = \sum_i c_i\vec{\psi}[t_1]c_i\vec{\psi}[t_2] = (\sum_i c_i^2)\vec{\psi}[t_1]\vec{\psi}[t_2] = \|\vec{c}\|^2\,\vec{\psi}(t_1)\vec{\psi}(t_2) \qquad \vec{c} \triangleq (c_1, c_2, \ldots)$

Principal component analysis

Trivial example

Consider simple case of data points that are all on the same high dimensional line

- St
-



Principal component analysis

Trivial example

Consider simple case of data points that are all on the same high dimensional line

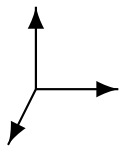
- Straight line is defined by a unit vector $\left\|\vec{\psi}\right\| = 1$
- Points on the line are defined by multiplying $\vec{\psi}$ by scalars
- The points can be formulated as $x_i = c_i\vec{\psi}$
- Covariance: $\text{cov}(t_1, t_2) = \sum_i x_i[t_1]x_i[t_2] = \sum_i c_i\vec{\psi}[t_1]c_i\vec{\psi}[t_2] = (\sum_i c_i^2)\vec{\psi}[t_1]\vec{\psi}[t_2] = \|\vec{c}\|^2 \vec{\psi}(t_1)\vec{\psi}(t_2) \qquad \vec{c} \triangleq (c_1, c_2, \ldots)$

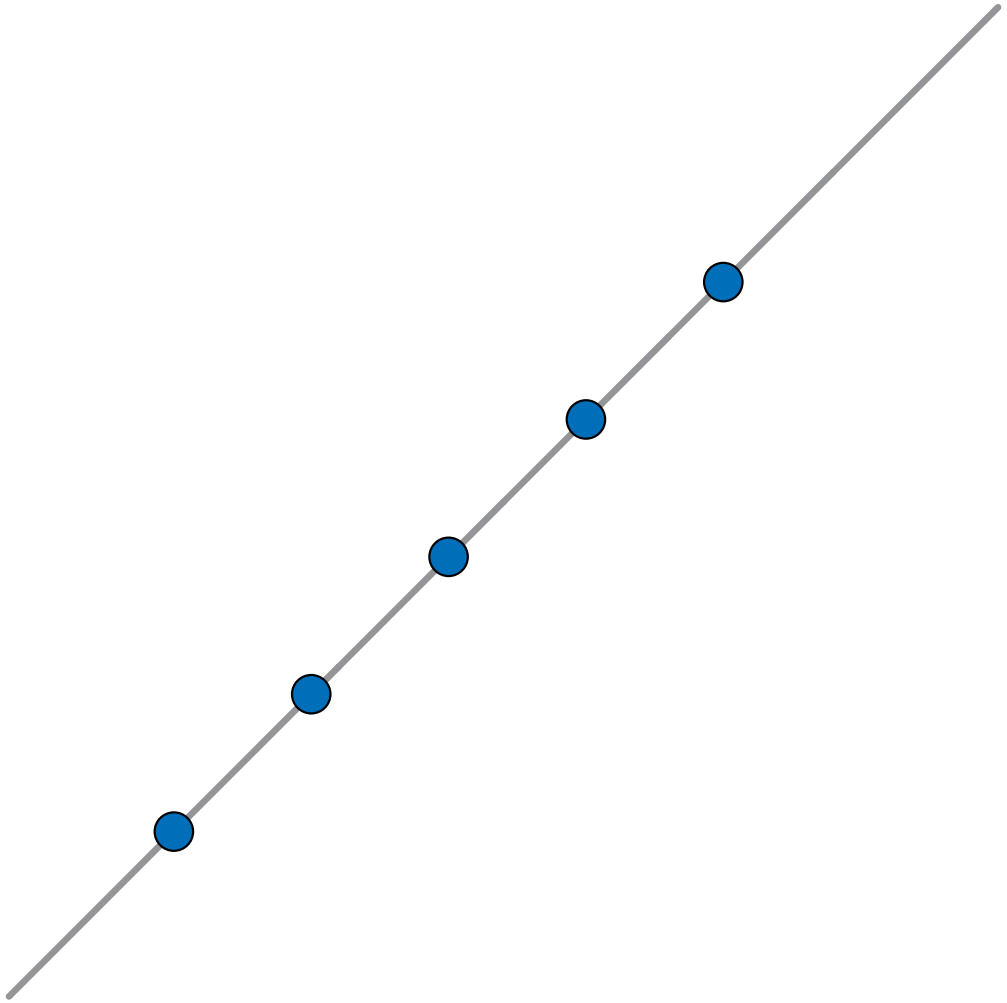
Covariance matrix has a single eigenvalue $\|\vec{c}\|^2$ and a single eigenvector $\vec{\psi}$, which defines principal direction of the data-point vectors

Principal component analysis

Trivial example

3D space

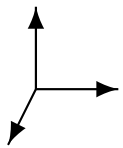


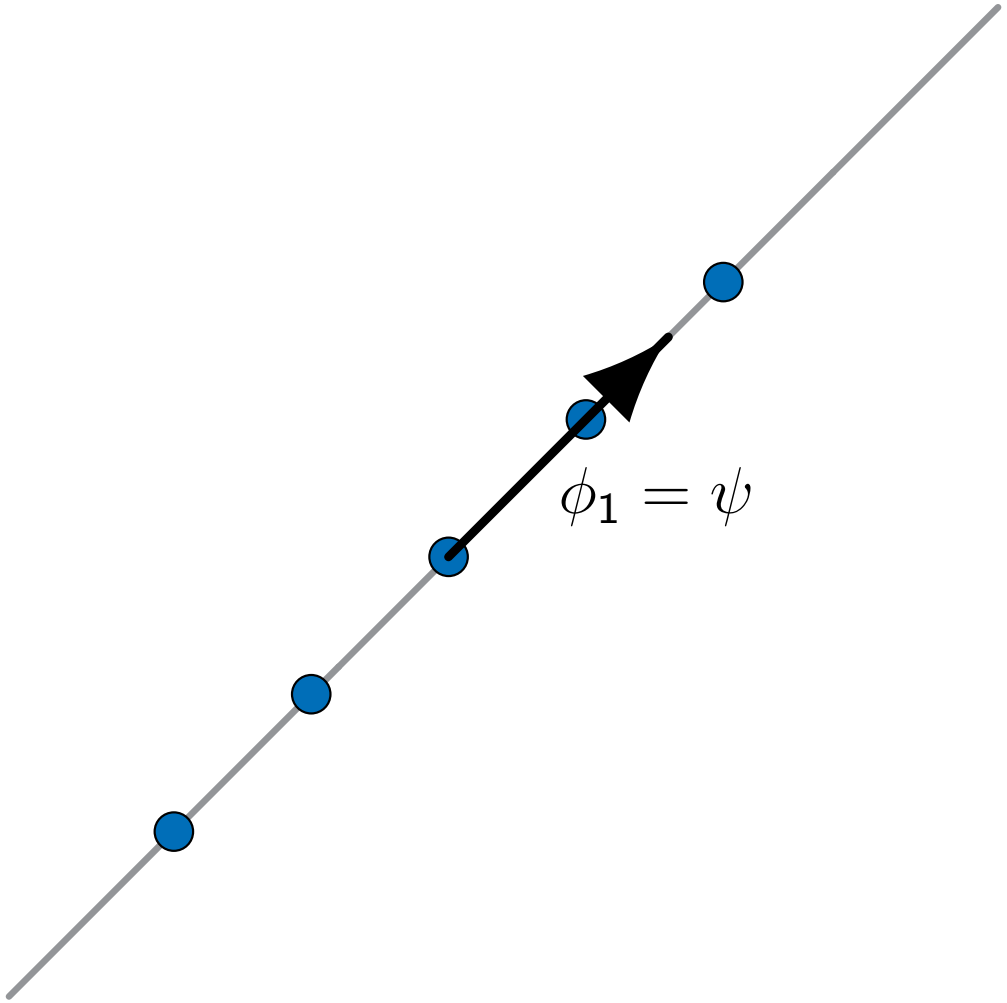


Principal component analysis

Trivial example

3D space

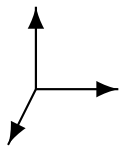


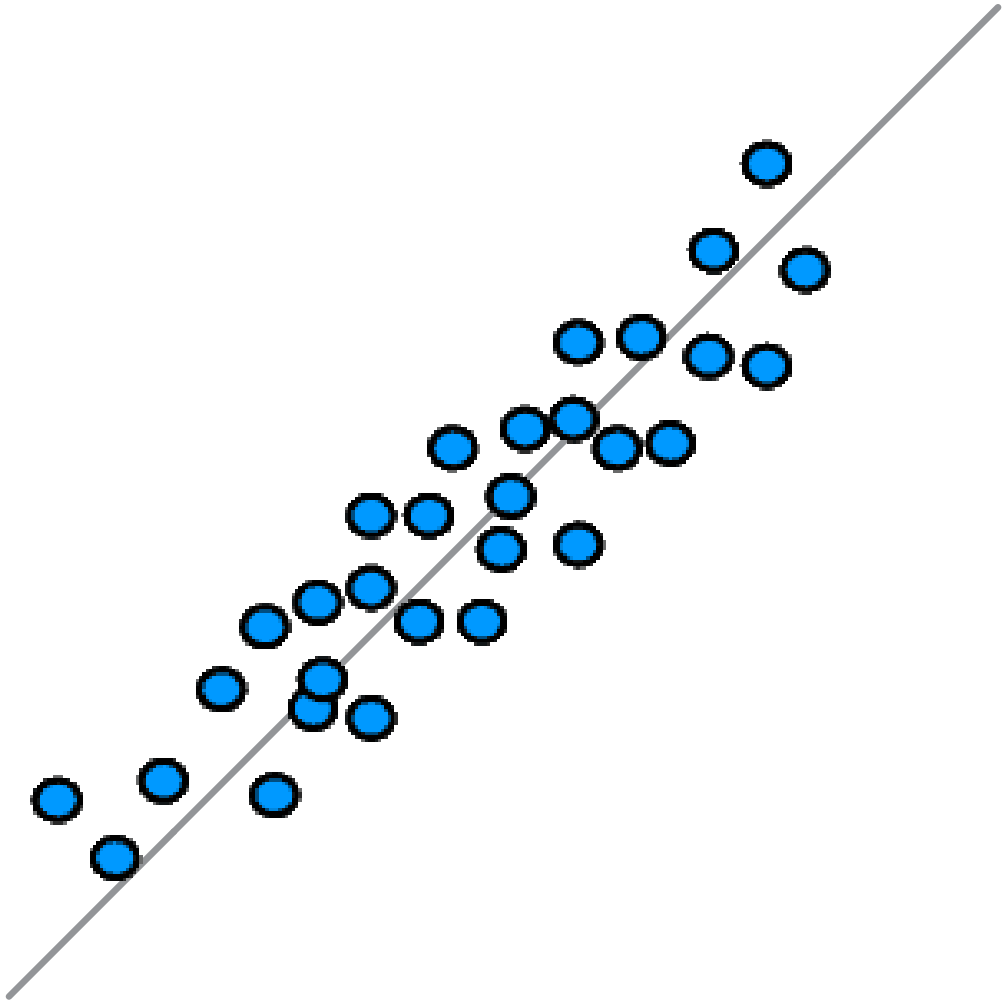


Principal component analysis

Trivial example

3D space

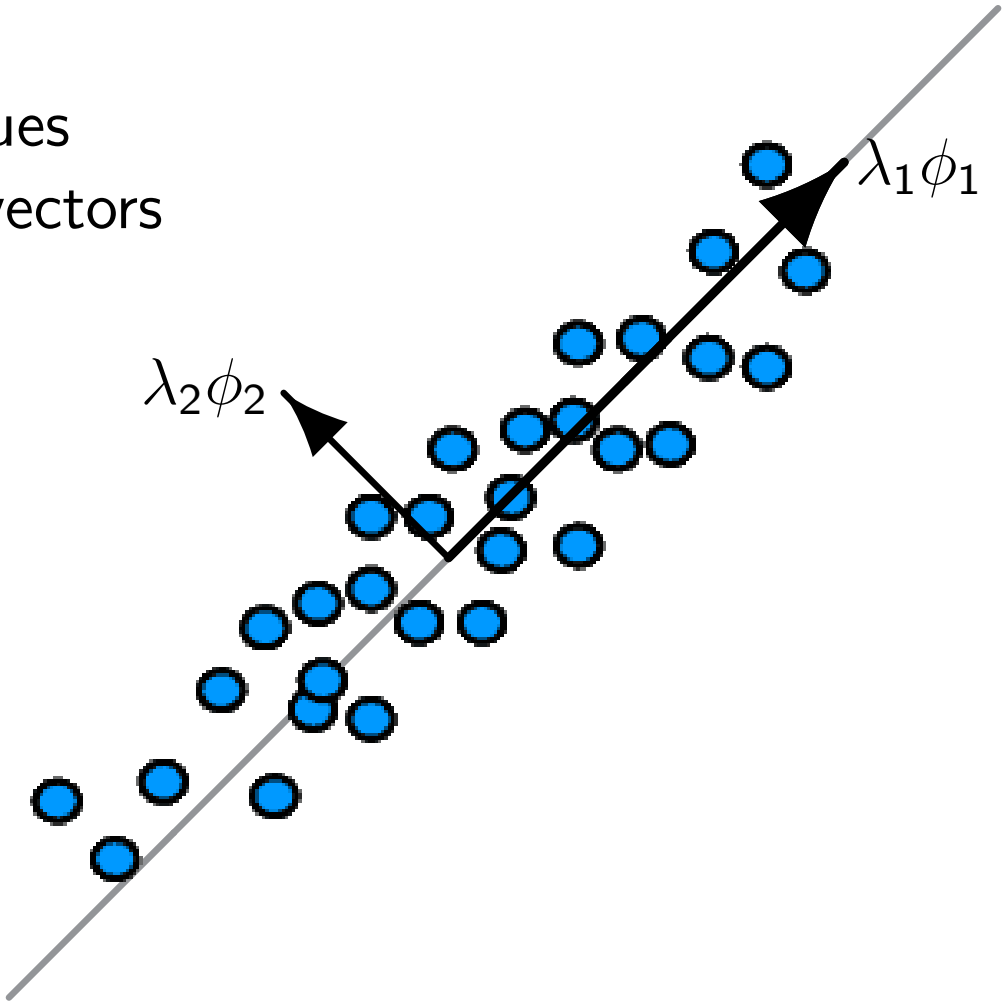




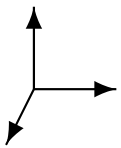
Principal component analysis

Trivial example

Length: eigenvalues
Direction: eigenvectors



3D space



principal components $\Rightarrow$ max var directions

Principal component analysis

PCA algorithm:

1. Centering
2. Covariance
3. SVD (or eigendecomposition)
4. Projection

Alternative method: Multi-Dimensional Scaling (MDS) - preserve distances/inner-products with minimal set of coordinates.

Summary

Preprocessing steps are crucial in preparing data for meaningful analysis.

Linear dimensionality reduction for alleviating the curse of dimensionality.

Principal Component Analysis (PCA) is a standard dimensionality reduction approach:

- Based on projecting data on leading eigenvectors of the covariance matrix.
- Minimizes reconstruction error by the projecion,
- Equivalently, finds a subspace that maximizes captured variance,
- In practice, SVD is used instead of eigendecomposition,