Unsupervised learning

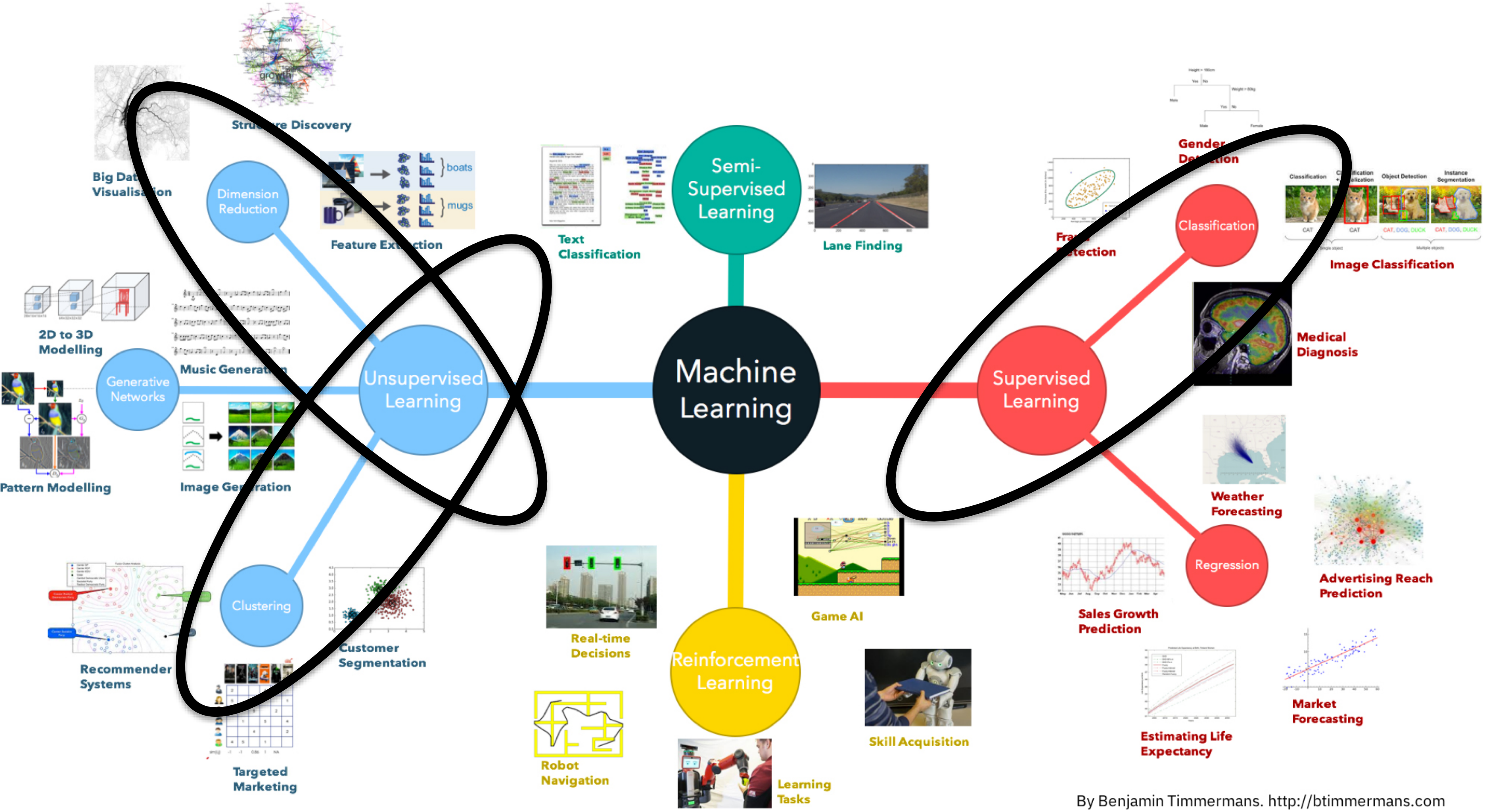
Gosia Migut

TUDelft

Admin stuff

- Next lecture on Thursday!

Machine learning

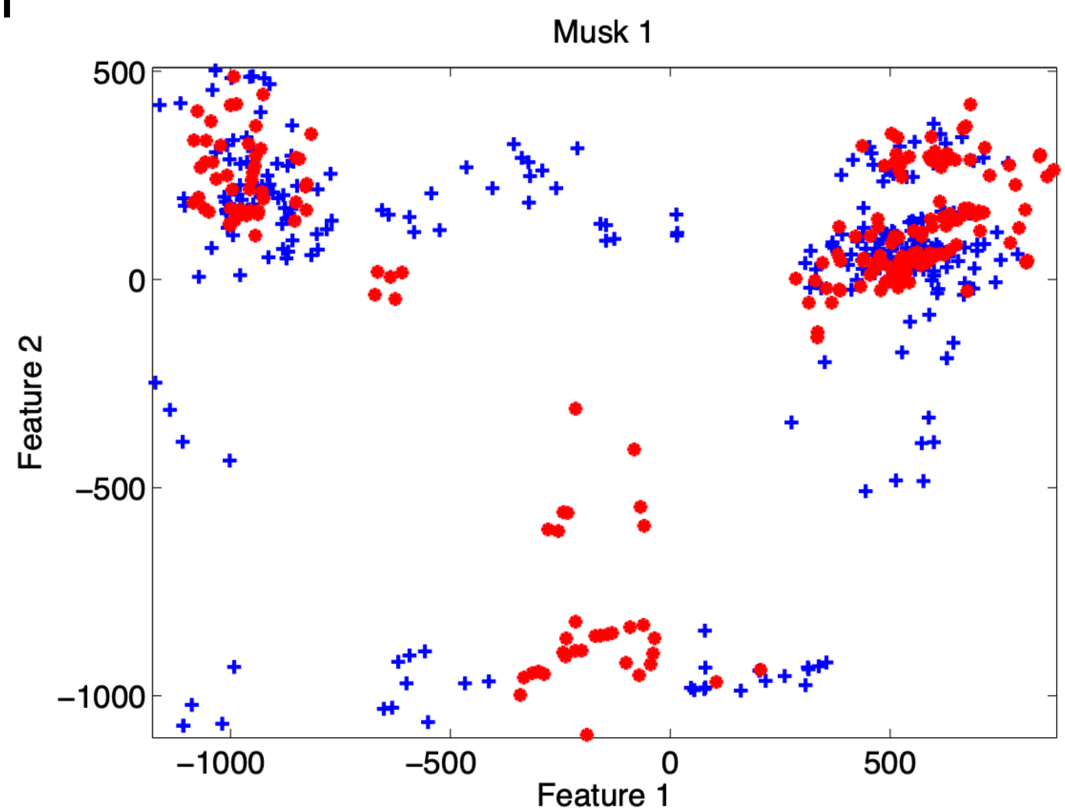


By Benjamin Timmermans. http://btimmermans.com

TU Delft

Recap supervised methods

- Until now only supervised methods
  - Each training example described by a feature vector and a label

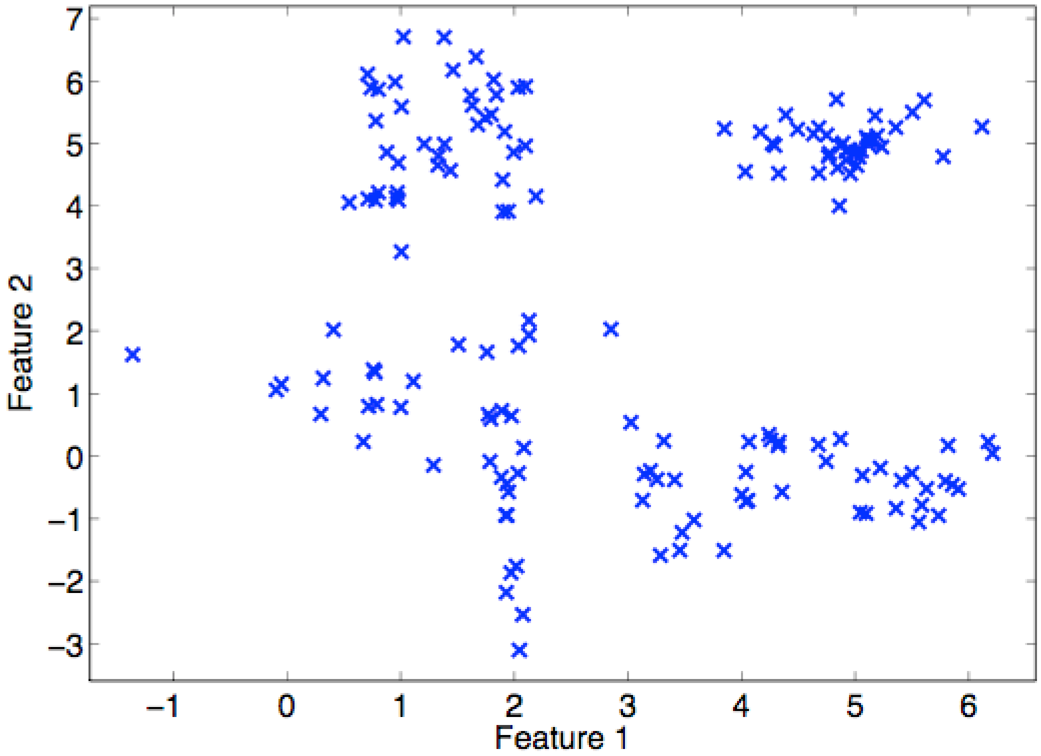


Supervised methods

Method	Generative	Discriminative	Linear	Non-linear	Parametric	Non-parametric
LDA	✓		✓		✓	
QDA	✓			✓	✓	
Nearest mean	✓		✓		✓	
Parzen	✓			✓		✓
K-nn	✓			✓		✓
Naive Bayes	✓		(✓)	✓	✓	✓
Logistic reg.		✓	✓		✓	
SVM		✓	✓	(✓)	✓	
Decision trees		✓		✓		✓
MLP		✓		✓	✓	

✓ discussed in this course
(✓) not discussed in this course

Unlabelled data: what now?



- Unsupervised learning: no labels/targets present

Unsupervised learning

- Dimensionality reduction
  - does not use information about the labels

- Clustering
  - Discover structures in unlabelled data

Dimensionality reduction

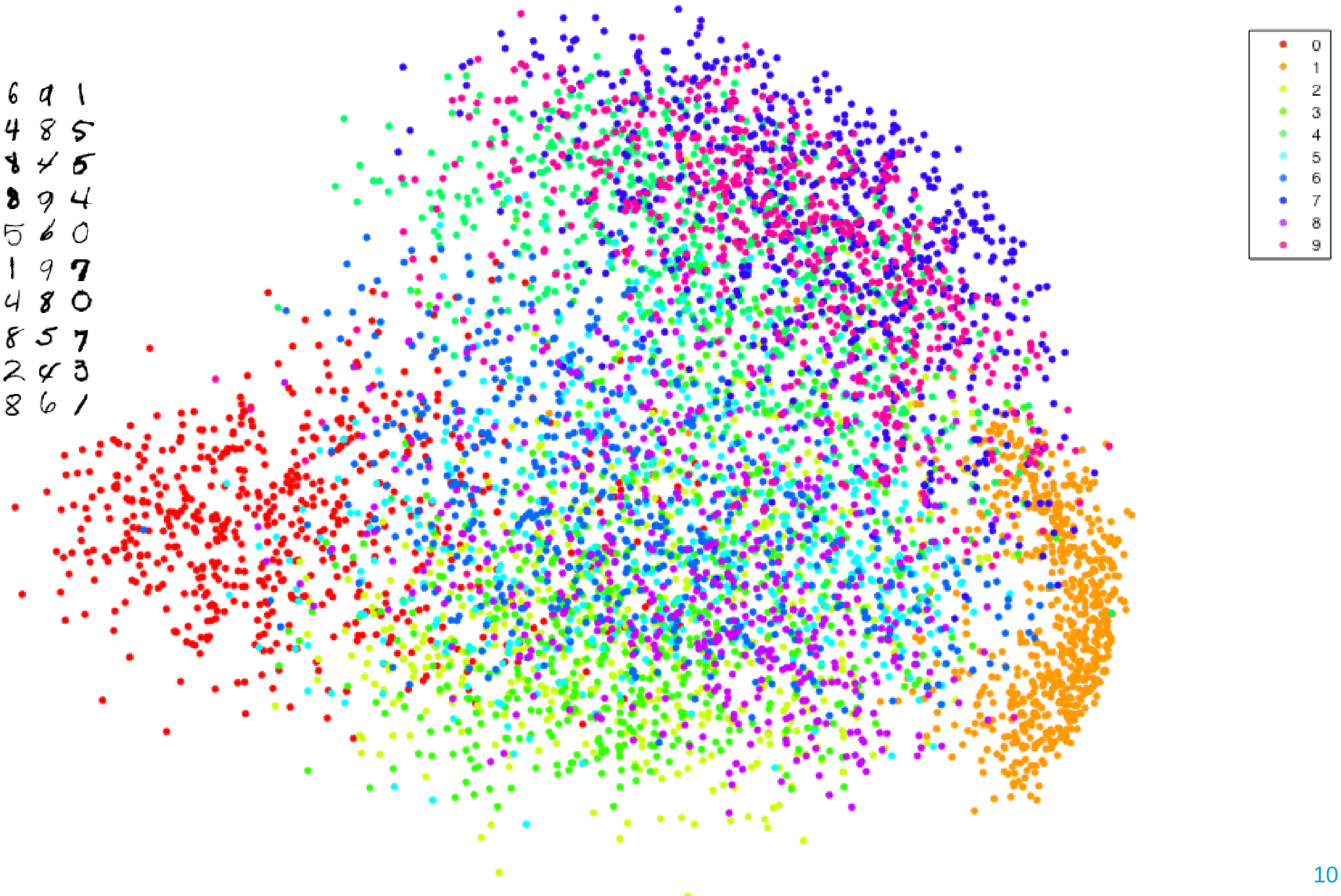
- Many data sets are *high-dimensional*: each instance is described by many features.

- Why do we want to reduce data dimensionality?
- What does it mean to reduce dimensionality?
- How Principal Component Analysis reduces dimensionality?

Dimensionality reduction

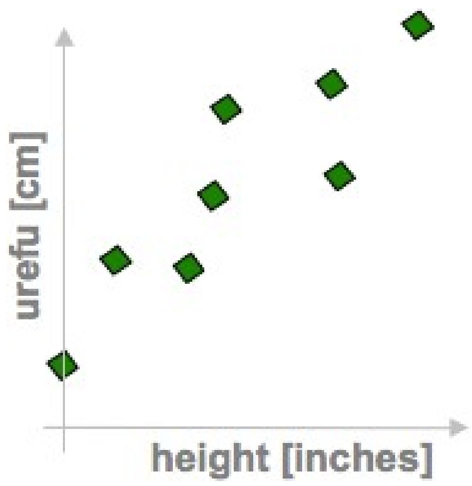
- Why do we want to reduce data dimensionality?
  - Make storage or processing of data easier
  - (Visual) discovery of hidden structure in the data
  - Remove redundant and noisy features
  - Intrinsic dimensionality might be smaller

Dimensionality reduction
Visual discovery of data structure



Dimensionality reduction
Redundant features

- Get a population, predict some property
  - instances represented as {urefu, height} pairs
  - what is the dimensionality of this data?



"height" = "urefu" in Swahili

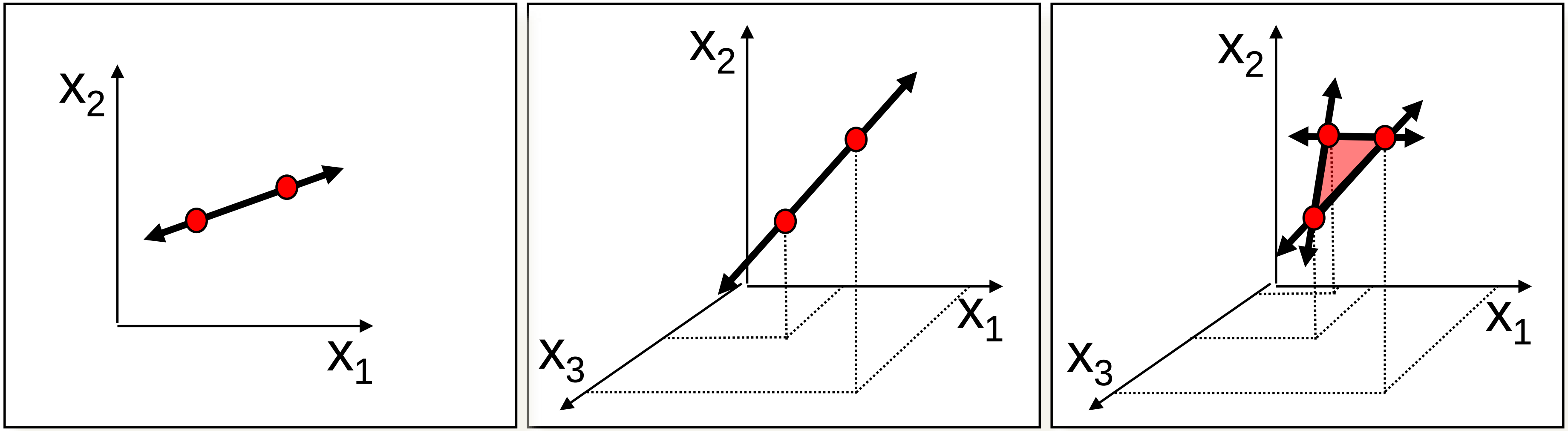
Dimensionality reduction
Redundant features

- Data points over time from different geographic areas over time:
  - X_1: # of traffic accidents
  - X_2: # of burst water pipes
  - X_3: # of snow-plow expenditures
  - X_4: # of forest fires
  - X_5: # of patients with heat stroke

Temperature?

TU Delft

Dimensionality reduction
Intrinsic dimensionality



2 objects, 2 dimensions
➔ 1 dimension

2 objects, 3 dimensions
➔ 1 dimension

3 objects, 3 dimensions
➔ 2 dimension

TU Delft

Dimensionality reduction
Curse of dimensionality

- As dimensionality grows: fewer observations per region
  - 1d: 3 regions, 2d: 3^2 regions, 1000d – hopeless
  - statistics need repetition

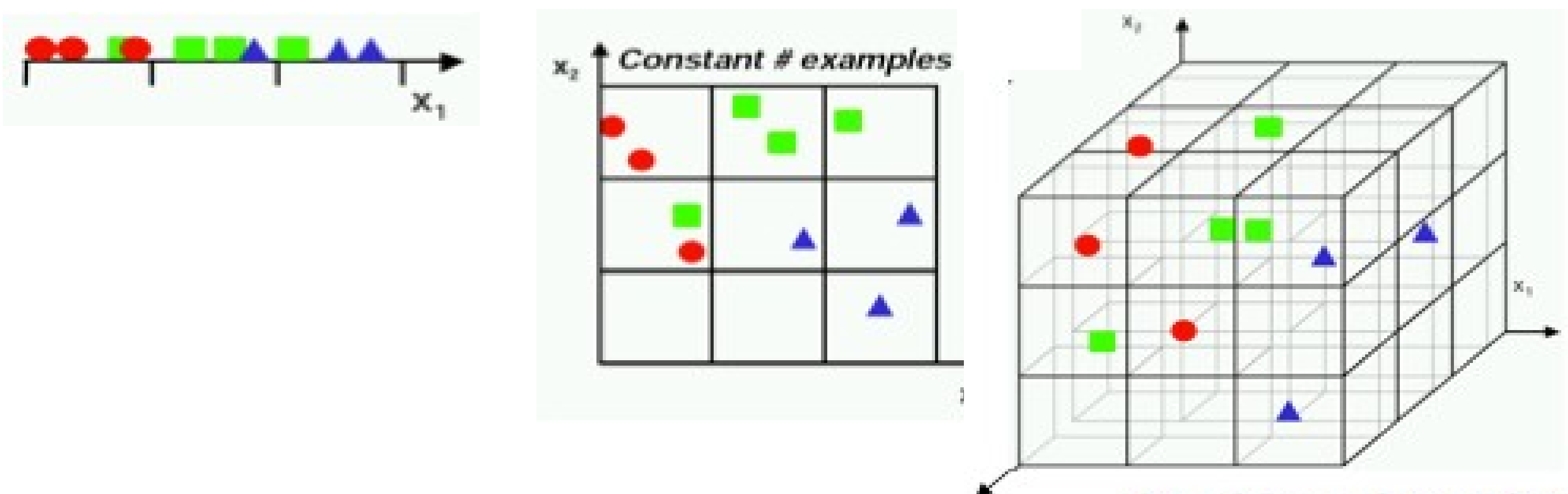


Figure © Ricardo Gutierrez-Osuna

TU Delft

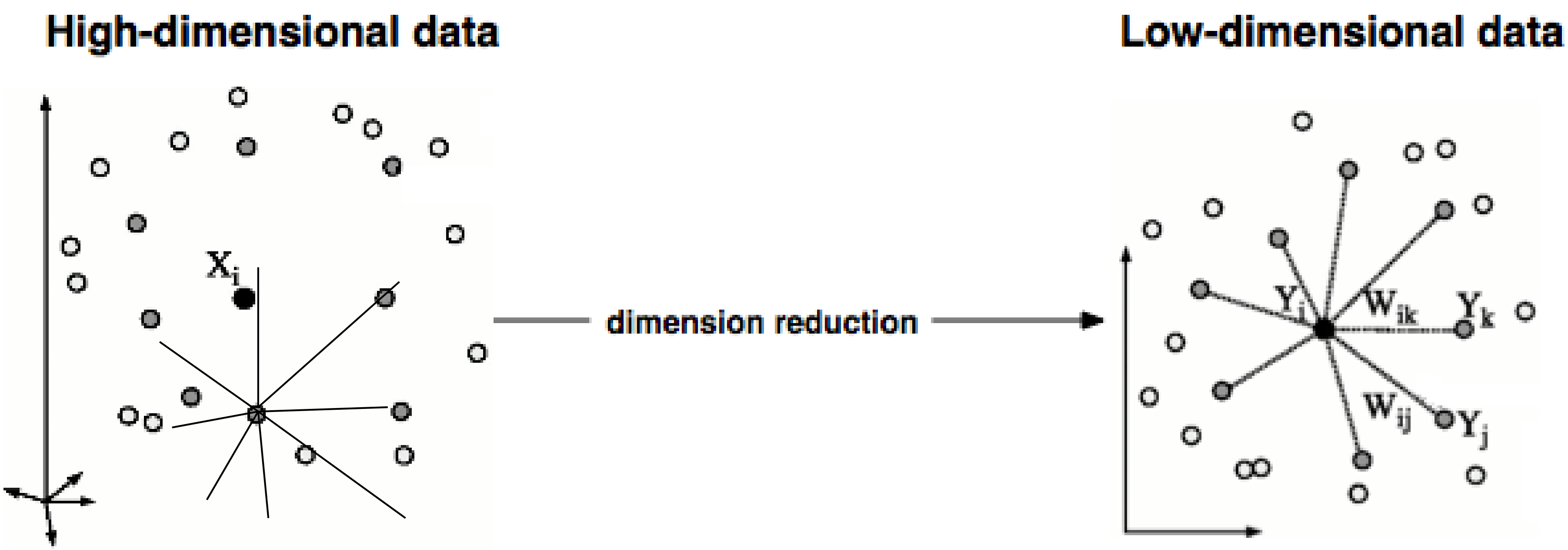
Dimensionality reduction

- Many data sets are *high-dimensional*: each instance is described by many features.

- Why do we want to reduce data dimensionality?
- **What does it mean to reduce dimensionality?**
- How Principal Component Analysis reduces dimensionality?

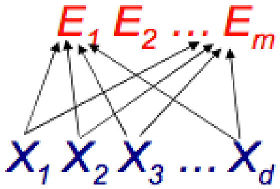
Reducing dimensionality

- Transform high-dimensional data to data of lower dimensionality, whilst *preserving the structure* in the original data as good as possible:



Reducing dimensionality: methods

- Use domain knowledge
- Feature selection
  - pick a subset of the original dimensions $X_1 X_2 X_3 \dots X_{d-1} X_d$
- **Feature extraction**
  - construct a new set of dimensions $E_i = f(X_1 \dots X_d)$

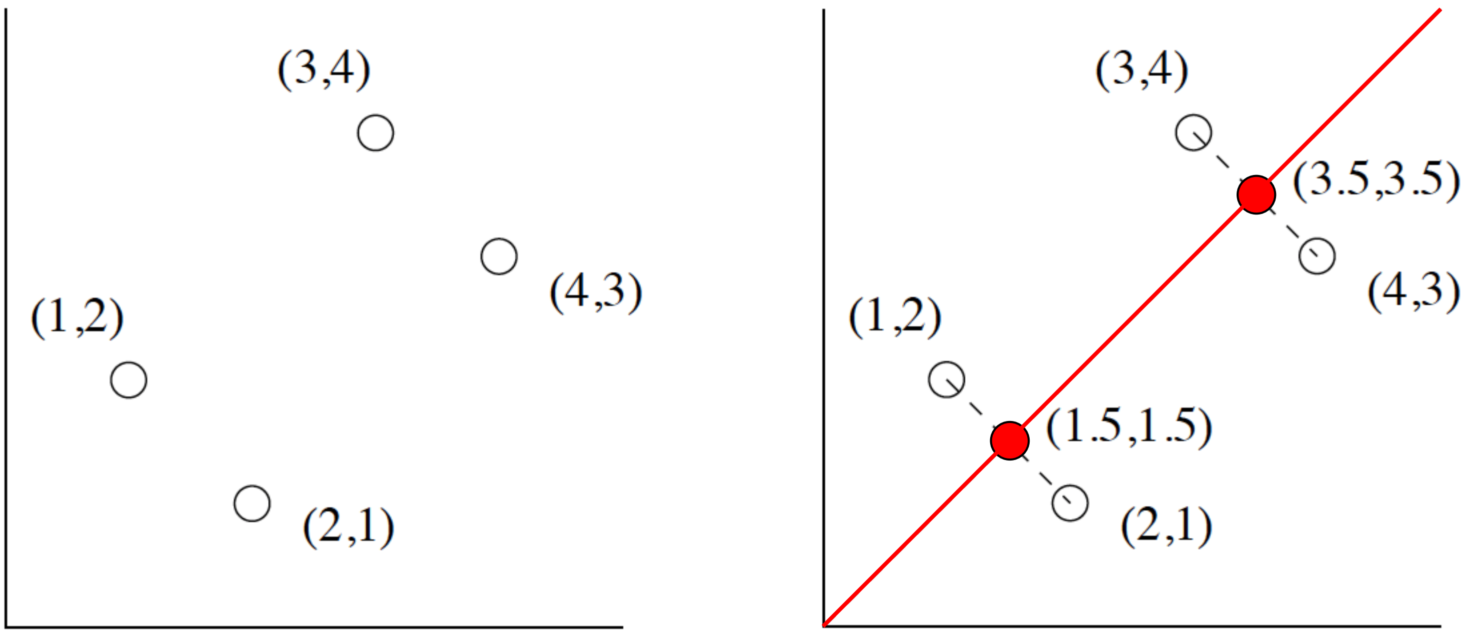


  - (linear) combinations of original

TUDelft

Reducing dimensionality
Feature extraction

- Many important dimensionality reduction techniques are *linear* techniques
- These project the data onto a *linear subspace* of *lower dimensionality (e.g. Principal Components Analysis)*



Dimensionality reduction

- Many data sets are *high-dimensional*: each instance is described by many features.

- Why do we want to reduce data dimensionality?
- What does it mean to reduce dimensionality?
- **How Principal Component Analysis reduces dimensionality?**

TU Delft

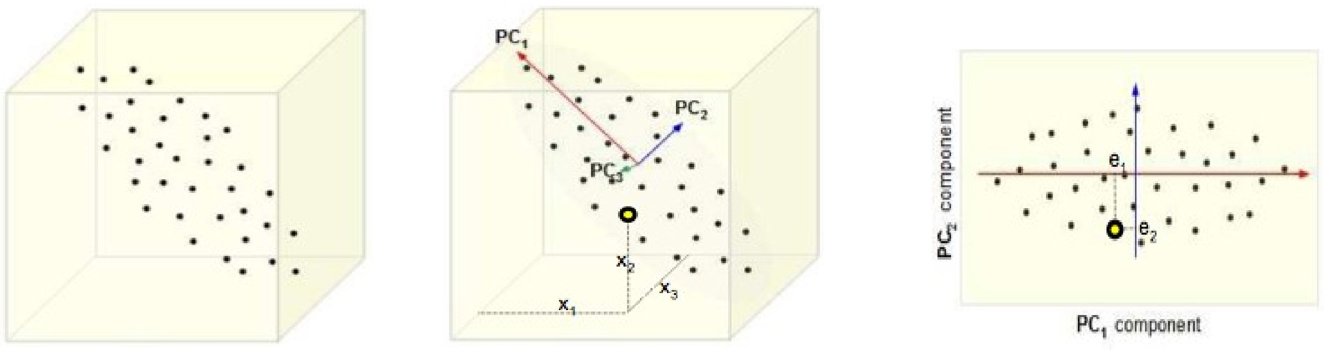
Principal components analysis

Principal Components Analysis

- Principal Components Analysis (PCA) maps the data onto a ***linear subspace***, such that the ***variance*** of the projected data is ***maximized.***

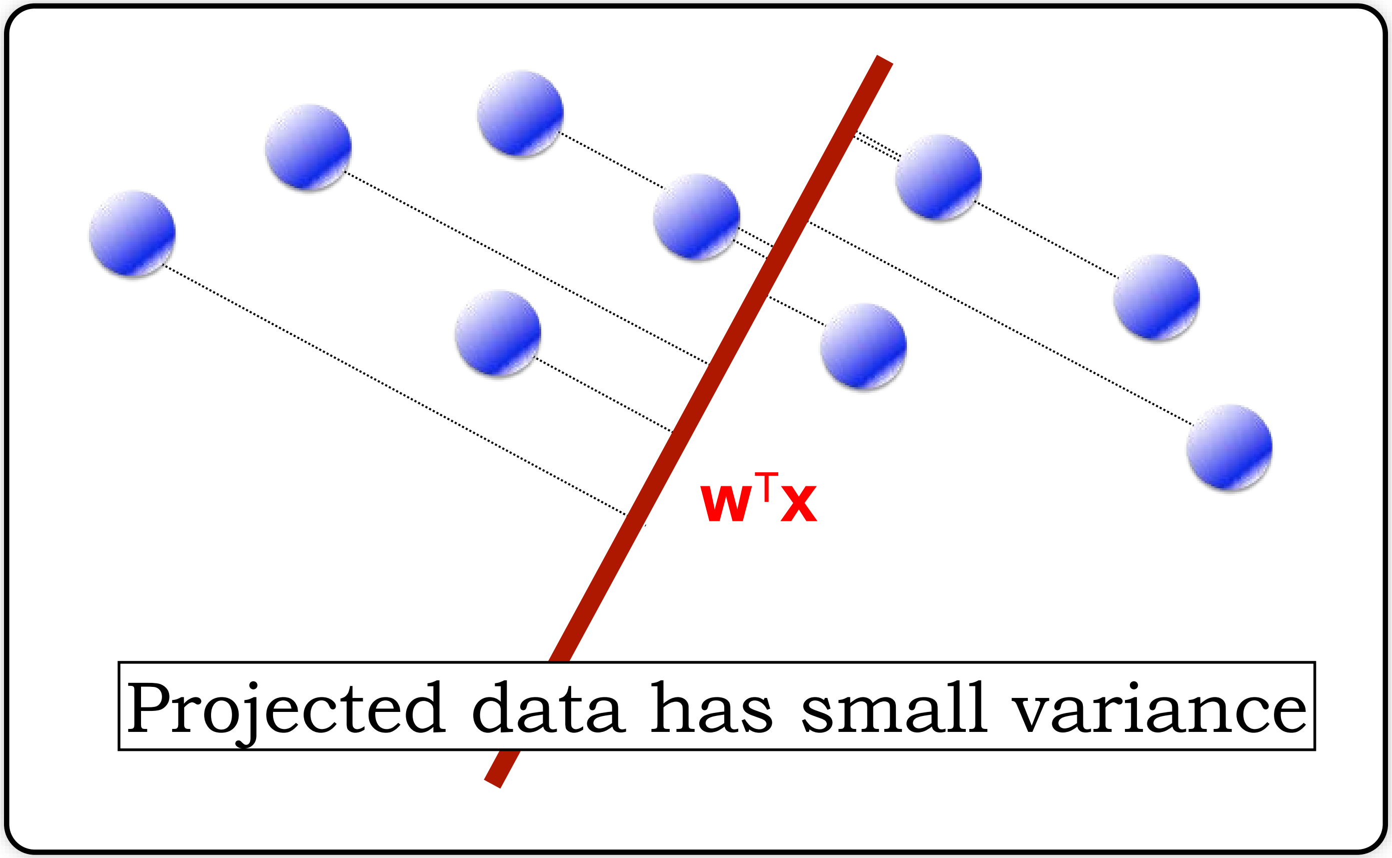
PCA overview

- Defines a set of principal components
  - 1^{st}: direction of the greatest variability in the data
  - 2^{nd}: perpendicular to 1^{st}, greatest variability of what's left
  - ... and so on until d (original dimensionality)
- First *m* components become *m* new dimensions
  - change coordinates of every data point to these dimensions



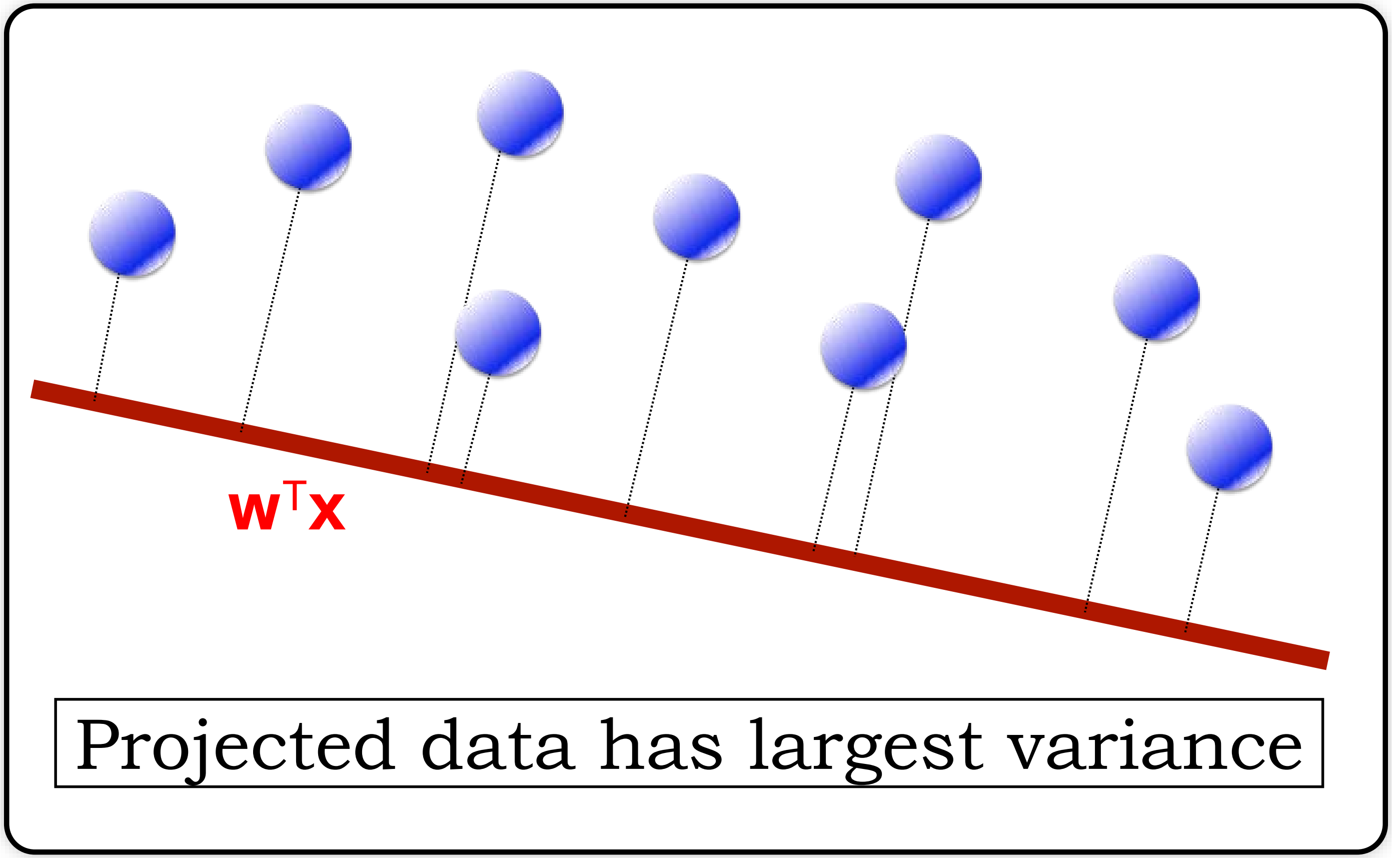
Principal components analysis

- Principal Components Analysis maps the data onto a *linear subspace*, such that the *variance* of the projected data is *maximized*:



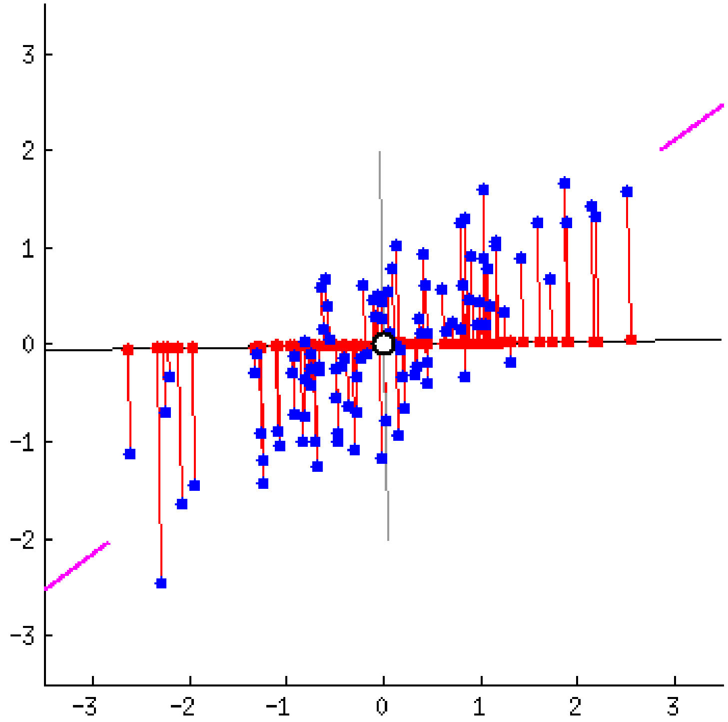
Principal components analysis

- Principal Components Analysis maps the data onto a *linear subspace*, such that the *variance* of the projected data is *maximized*:



Principal components analysis

- Principal Components Analysis maps the data onto a *linear subspace*, such that the *variance* of the projected data is *maximized*:



TU Delft

Principal components analysis

- Principal Components Analysis (PCA) maps the data onto a *linear subspace*, such that the *variance* of the projected data is *maximized*

- Recall the definition of variance:

$$var(\mathbf{x}) = \mathbb{E}[(x - \mathbb{E}[x])^2] = \frac{1}{N}\sum_{n=1}^{N}\left(x_n - \frac{1}{N}\sum_{n=1}^{N} x_n\right)^2$$

Principal components analysis

- Principal Components Analysis (PCA) maps the data onto a *linear subspace*, such that the *variance* of the projected data is maximized

- Recall the definition of variance:

$$var(\mathbf{x}) = \mathbb{E}[(x - \mathbb{E}[x])^2] = \frac{1}{N}\sum_{n=1}^{N}\left(x_n - \frac{1}{N}\sum_{n=1}^{N} x_n\right)^2$$

- So PCA performs maximization:

$$\max_{\|w\|^2=1} var(w^T x)$$

Covariance matrix

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i$$

$$\hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \hat{\mu})(\mathbf{x}_i - \hat{\mu})^T$$

- The *covariance* of two variables is the expectation of their (zero-mean) product:

$$Cov(x,y) = \mathbb{E}\left[(x - \mathbb{E}[x])(y - \mathbb{E}[y])\right]$$

- The *covariance matrix* is the matrix with all pairwise covariances:

$$\mathrm{M} = \begin{bmatrix} \mathbb{E}[(X_1-\mu_1)(X_1-\mu_1)] & \mathbb{E}[(X_1-\mu_1)(X_2-\mu_2)] & \cdots & \mathbb{E}[(X_1-\mu_1)(X_D-\mu_D)] \\ \mathbb{E}[(X_2-\mu_2)(X_1-\mu_1)] & \mathbb{E}[(X_2-\mu_2)(X_2-\mu_2)] & \cdots & \mathbb{E}[(X_2-\mu_2)(X_D-\mu_D)] \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{E}[(X_D-\mu_D)(X_1-\mu_1)] & \mathbb{E}[(X_D-\mu_D)(X_2-\mu_2)] & \cdots & \mathbb{E}[(X_D-\mu_D)(X_D-\mu_D)] \end{bmatrix}$$

- If data is *zero-mean*, the covariance matrix is simply: $M = \frac{1}{n}XX^T$

TUDelft

Principal components analysis

- *Principal components* are given by the eigenvectors of the covariance matrix

$$Me = \lambda e$$

- First principal component is given by the eigenvector with the corresponding highest eigenvalue

Eigenvalues & eigenvectors: Definition

- M square matrix, λ constant, **e** a non-zero column vector
- λ is an eigenvalue of M and **e** is the corresponding eigenvector of M if

$$Me = \lambda e$$

- Avoiding ambiguity regarding length: eigenvector to be *unit vector*

- λ and **e** form eigenpairs

- Watch: 3blue1brown: **Eigenvectors and eigenvalues | Essence of linear algebra, chapter 14**

Eigenvalues & eigenvectors: Example

- Let *M* be matrix $\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix}$
- One of eigenvectors of M is $\begin{bmatrix} 1/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix}$
- Corresponding eigenvalue is 7, since

$$\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} 1/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix} = 7 \begin{bmatrix} 1/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix}$$

- Eigenvector is indeed unit vector

$$(1/\sqrt{5})^2 + (2/\sqrt{5})^2 = 1/5 + 4/5 = 1$$

How to find eigenpairs?

- Pivotal condensation
- Power iteration

How to find eigenpairs? Pivotal condensation

- Restate definition eigenpair $M\mathbf{e} = \lambda\mathbf{e}$ as

$$(M - \lambda I)\mathbf{e} = \mathbf{0}$$

- For this to hold the determinant of $(M - \lambda I)$ must be 0

- Determinant of $(M - \lambda I)$ is an *n*-th degree polynomial from which we can get the *n* values for λ that are eigenvalues of *M*

Eigenpairs: Pivotal condensation (example)

- Set *M* to $\left[\begin{array}{cc} 3 & 2 \\ 2 & 6 \end{array}\right]$
- Then $M - \lambda I$ is $\left[\begin{array}{cc} 3-\lambda & 2 \\ 2 & 6-\lambda \end{array}\right]$
- Determinant is $(3-\lambda)(6-\lambda) - 4$.
- Setting to zero, solving equation $\lambda^2 - 9\lambda + 14 = 0$
- Gives solutions $\lambda = 7 \text{ and } \lambda = 2$ being principal eigenvalues
- Let **e** be vector of unknowns $\left[\begin{array}{c} x \\ y \end{array}\right]$
- Solve $\left[\begin{array}{cc} 3 & 2 \\ 2 & 6 \end{array}\right]\left[\begin{array}{c} x \\ y \end{array}\right] = 7\left[\begin{array}{c} x \\ y \end{array}\right]$

Eigenpairs: Pivotal condensation (example)

- Two equations: $\left[\begin{array}{rcl} 3x+2y & = & 7x \\ 2x+6y & = & 7y \end{array}\right]$
- Both saying the same thing $y = 2x$
- Possible eigenvector: $\left[\begin{array}{c} 1 \\ 2 \end{array}\right]$
- Make unit vector (divide by length): $\left[\begin{array}{c} 1/\sqrt{5} \\ 2/\sqrt{5} \end{array}\right]$

- Second eigenvalue: repeat with $\lambda = 2$
- Equation becomes: $x = -2y$
- Second eigenvector: $\left[\begin{array}{r} 2/\sqrt{5} \\ -1/\sqrt{5} \end{array}\right]$

How to find eigenpairs? Power iteration

- Start with any unit vector $\mathbf{x_0}$ (of appropriate length)
- Compute $M\,\mathbf{x}_k$ until it converges:

$$\mathbf{x}_{k+1} := \frac{M\mathbf{x}_k}{\|M\mathbf{x}_k\|}$$

$\|A\|$ frobenius norm; the square root of the sum of the absolute squares of elements of N

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}$$

- Limiting vector is the *principal eigenvector* (eigenvector with largest eigenvalue)
- When converged, compute eigenvalue $\lambda_1 = \mathbf{x}^T M \mathbf{x}$
- To find second eigenpair create new matrix
- Use power iteration on *M** to compute its principal eigenvector, etc.

$$M^* = M - \lambda_1 \mathbf{x}\mathbf{x}^T$$

TUDelft

Find eigenpairs with power iteration Example

- Let $$M = \begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix}$$

- Start with $\mathbf{x}_0$ being vector with 1s

- Multiply $M\,\mathbf{x}_0$: $$\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 8 \end{bmatrix}$$

- Frobenius norm equals $\sqrt{5^2 + 8^2} = \sqrt{89} = 9.434$

- Obtain $\mathbf{x}_1$: $$\mathbf{x}_1 = \begin{bmatrix} 0.530 \\ 0.848 \end{bmatrix}$$

Find eigenpairs with power iteration Example

- Next iteration: $\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} 0.530 \\ 0.848 \end{bmatrix} = \begin{bmatrix} 3.286 \\ 6.148 \end{bmatrix}$

- Frobenius norm equals 6.971 so x_2 becomes

$$\mathbf{x}_2 = \begin{bmatrix} 0.471 \\ 0.882 \end{bmatrix}$$

- Repeat, converges to $\boldsymbol{x} = \begin{bmatrix} \mathbf{0.447} \\ \mathbf{0.894} \end{bmatrix}$
- Principal eigenvalue

$$\lambda = \mathbf{x}^T M \mathbf{x} = \begin{bmatrix} 0.447 & 0.894 \end{bmatrix} \begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} 0.447 \\ 0.894 \end{bmatrix} = 6.993$$

Find eigenpairs with power iteration Example

- To find second eigenpair create new matrix

$$M^* = M - \lambda_1 \mathbf{x}\mathbf{x}^T$$

- Use power iteration on *M** to compute its principal eigenvector, etc.

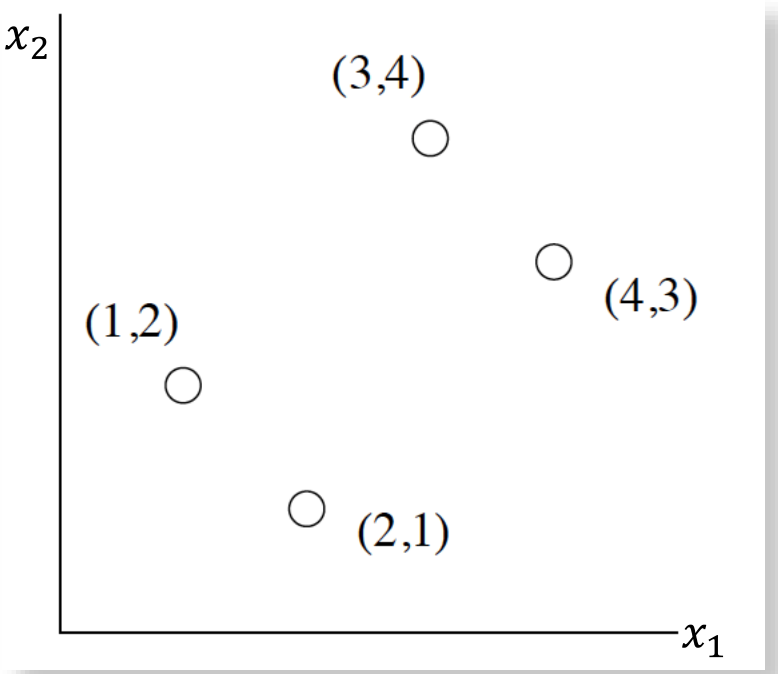
TUDelft

Principal components analysis

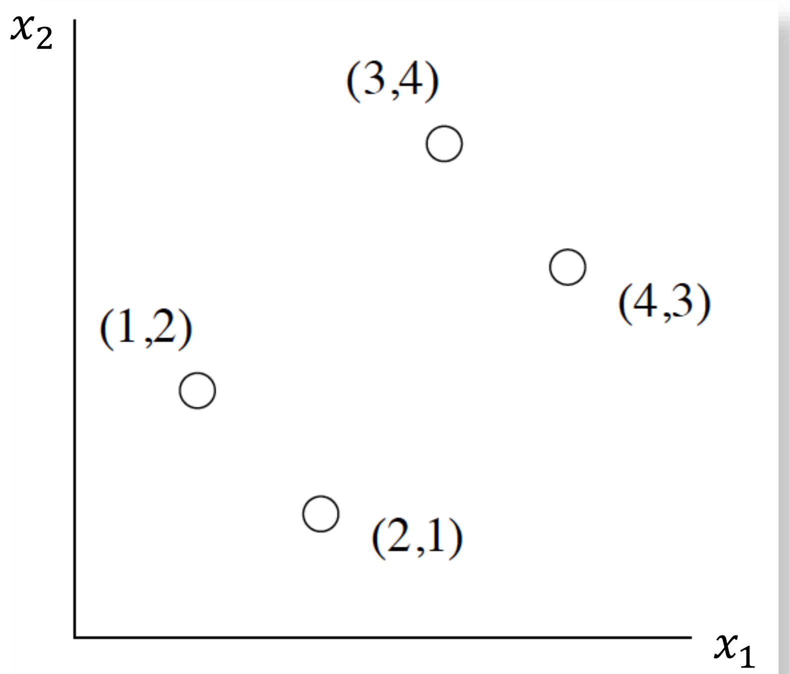
- *Principal components* are given by the eigenvectors of the covariance matrix

$$Me = \lambda e \qquad\qquad M = \frac{1}{n} X X^T$$

- Perform PCA for:



Principal component analysis
Example from the book (mistakes highlighted!)



$$X = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \end{bmatrix}$$

Not zero-mean

$$M = \boxed{XX^T} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 4 \\ 4 & 3 \end{bmatrix} = \begin{bmatrix} 30 & 28 \\ 28 & 30 \end{bmatrix}$$

not a covariance matrix; M should be $\frac{1}{n} XX^T$, where X is zero-mean

Principal component analysis (example)

- Assuming M is a covariance matrix $M = \begin{bmatrix} 30 & 28 \\ 28 & 30 \end{bmatrix}$ (*X* not zero-mean data)
- Find eigenvalues det $(M - \lambda I)$ = 0 $(30-\lambda)(30-\lambda) - 28 \times 28 = 0$
- Solution $\lambda = 58 \text{ and } \lambda = 2$
- Solve $\begin{bmatrix} 30 & 28 \\ 28 & 30 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 58 \begin{bmatrix} x \\ y \end{bmatrix}$
- Two equations telling same thing $\begin{bmatrix} 30x + 28y & = & 58x \\ 28x + 30y & = & 58y \end{bmatrix}$ $x = y$
- Unit eigenvector corresponding to eigenvalue 58: $\begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$
- Similarly for eigenvalue 2

$$\begin{bmatrix} 30 & 28 \\ 28 & 30 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 2 \begin{bmatrix} x \\ y \end{bmatrix} \quad \begin{bmatrix} 30x + 28y & = & 2x \\ 28x + 30y & = & 2y \end{bmatrix} \quad x = -y \quad \begin{bmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$$

Example from the book correct covariance matrix calculation

- Given $X = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \end{bmatrix}$
- Mean of each dimension:

$$\mu_1 = \frac{1}{n}\sum_{i=1}^{n} x_{1i} = \frac{1}{4}(1+2+3+4) = 2.5$$

$$\mu_2 = \frac{1}{n}\sum_{i=1}^{n} x_{2i} = \frac{1}{4}(2+1+4+3) = 2.5$$

- Zero-mean data:

$$X = \begin{bmatrix} x_{11}-\mu_1 & x_{12}-\mu_1 & x_{13}-\mu_1 & x_{14}-\mu_1 \\ x_{21}-\mu_2 & x_{22}-\mu_2 & x_{23}-\mu_2 & x_{24}-\mu_2 \end{bmatrix} = \begin{bmatrix} -1.5 & -0.5 & 0.5 & 1.5 \\ -0.5 & -1.5 & 1.5 & 0.5 \end{bmatrix}$$

- Covariance matrix: $M = \frac{1}{n}XX^T = \frac{1}{4}\begin{bmatrix} -1.5 & -0.5 & 0.5 & 1.5 \\ -0.5 & -1.5 & 1.5 & 0.5 \end{bmatrix}\begin{bmatrix} -1.5 & -0.5 \\ -0.5 & -1.5 \\ 0.5 & 1.5 \\ 1.5 & 0.5 \end{bmatrix}$

Principal component analysis (example)

- Matrix of eigenvectors for XX^T becomes

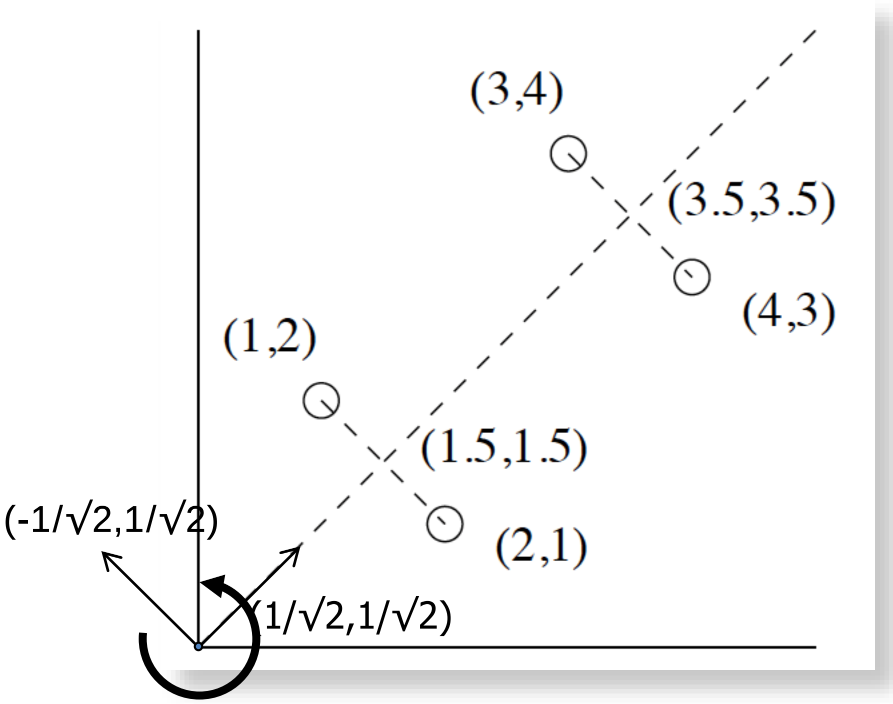
$$E = \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}$$

$$X^T E = \begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 4 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 3/\sqrt{2} & 1/\sqrt{2} \\ 3/\sqrt{2} & -1/\sqrt{2} \\ 7/\sqrt{2} & 1/\sqrt{2} \\ 7/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$$

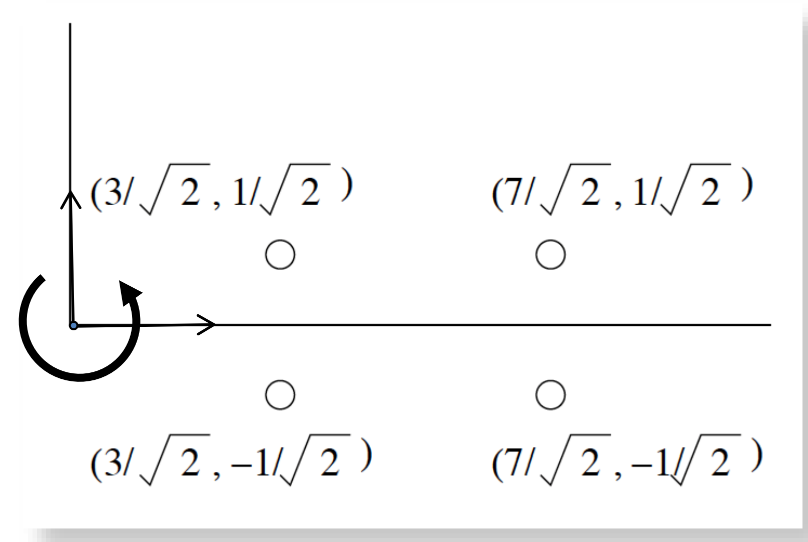
- Any matrix of orthonormal vectors represents a rotation of the axes of a Euclidean space. Matrix E can be viewed as a rotation (in this case 45 degrees counterclockwise)

Principal component analysis (example)

- First point [1,2] transformed into $[3/\sqrt{2}, 1/\sqrt{2}]$



Original points, eigenvectors, projections



New coordinate system

TUDelft

PCA in a nutshell

- X matrix whose rows represent (zero-mean) points in Euclidean space

- Compute covariance XX^T and its eigenpairs

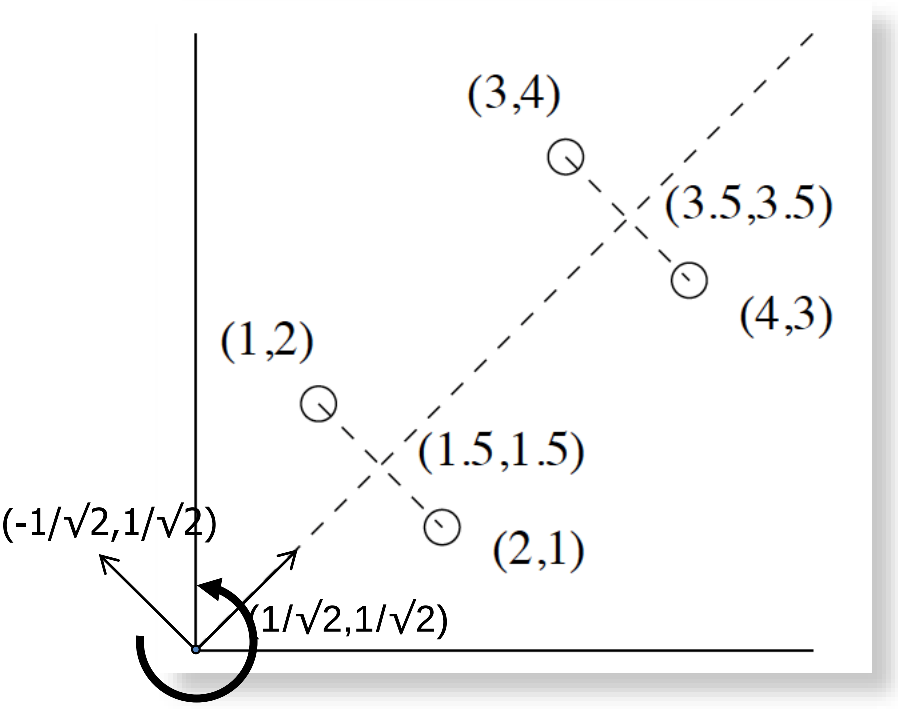
- E matrix whose columns are the eigenvectors, ordered as largest eigenvalues first

- X^TE : points of X transformed into new coordinate space
  - First axis (largest eigenvalue) most significant
  - Second axis (second eigenpair), next most siginificant

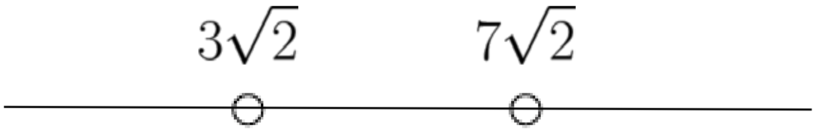
- Let E_k be first k columns of E

- Then X^TE_k is *k-dimensional* representation of X

Principal component analysis (example)



Original points, eigenvectors, projections



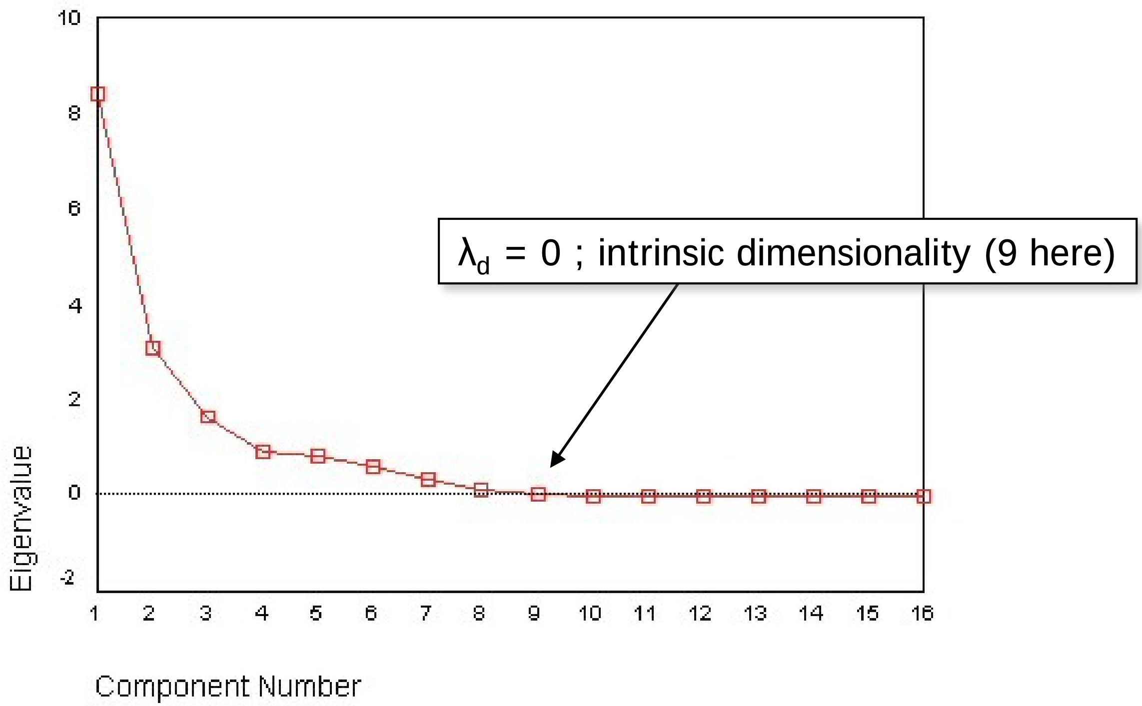
XE_1

New coordinate system

1 dimensional !

PCA scree plot

- *Scree plot* of eigenvalues shows amount of variance retained by the eigenvectors (*principal components, PCs*):

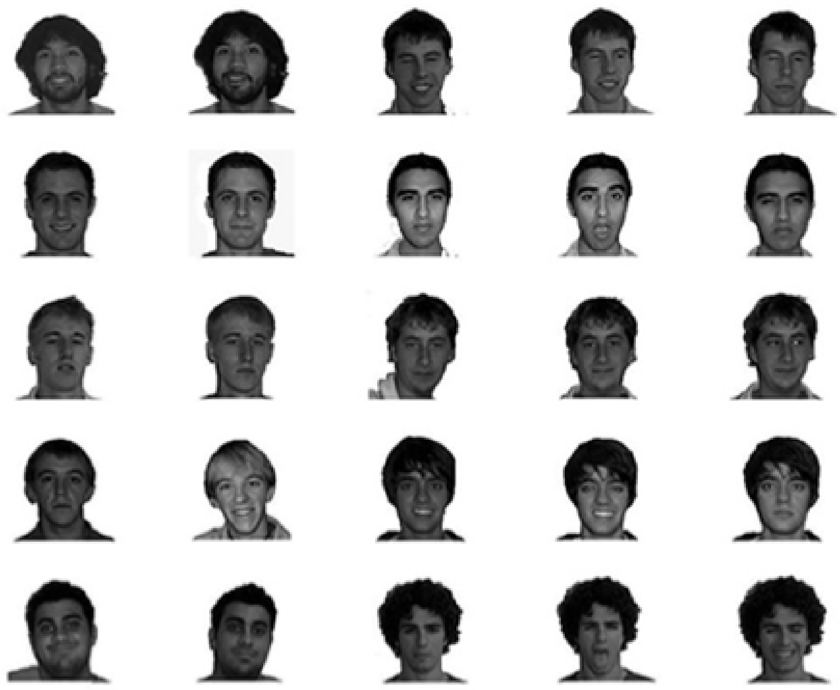


- First *K* PCs explain $\dfrac{\sum_{d=1}^{K} \lambda_d}{\sum_{d'=1}^{D} \lambda_{d'}} \times 100\%$ of variance

TU Delft

Eigenfaces

- Suppose we are applying PCA on the following set of face images:

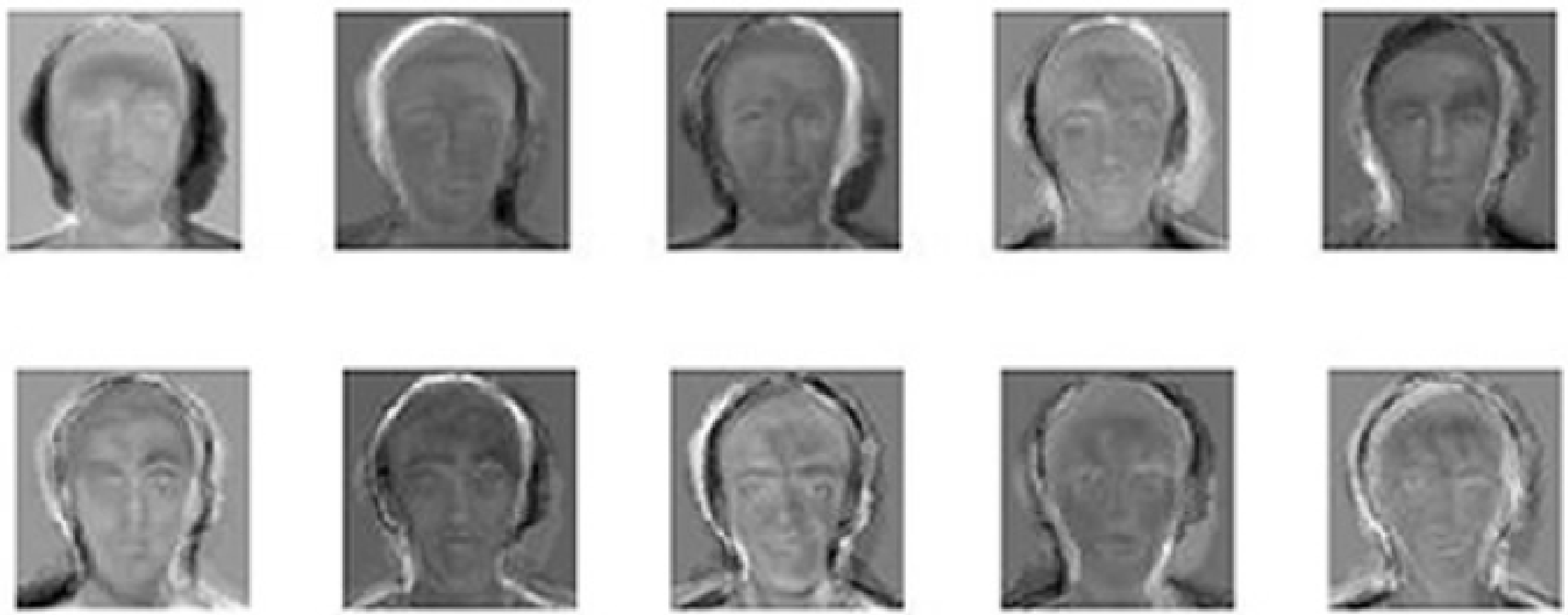


- Image is matrix; *but* represented as a row vector !
- Eigenvectors also row vector, so eigenvector is also an image !

TUDelft

Eigenfaces

- Example of first eigenvectors of set of face images (faces were aligned):



Principal components analysis

- Since we have projected onto a subspace, we can reconstruct the data in the original data space by performing the inverse of the projection:

$$\hat{\mathbf{x}} = \mathbf{w}\mathbf{w}^{\mathrm{T}}\mathbf{x}$$

- Example reconstructions of face images and digits (using 30D PCA subspace):

original

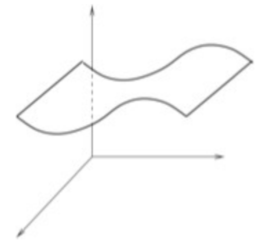
reconstructed

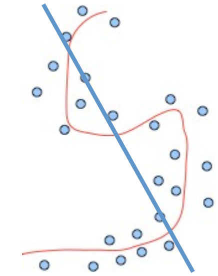
original

reconstructed

TU Delft

PCA: practical issues

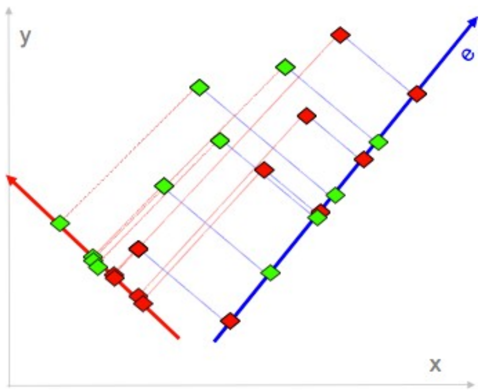




- Covariance extremely sensitive to large values
  - Multiply some dimensions by 1000
    - Dominates covariance
    - Becomes a principal component
  - Normalize each dimension to zero mean and unit variance: $x' = \frac{x - \mu}{\sigma}$
- PCA assumes underlying subspace is linear
  - 1d: straight line, 2d: plane
  - transform to handle non-linear spaces (manifolds)

PCA and classification

- PCA is unsupervised
  - maximizes overall variance of the data along a small set of directions
  - does not know anything about class labels
  - can pick direction that makes it hard to separate classes
- Discriminative approach
  - look for a dimension that makes it easy to separate classes

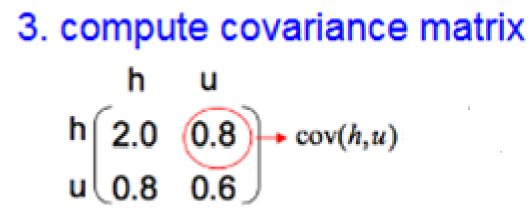


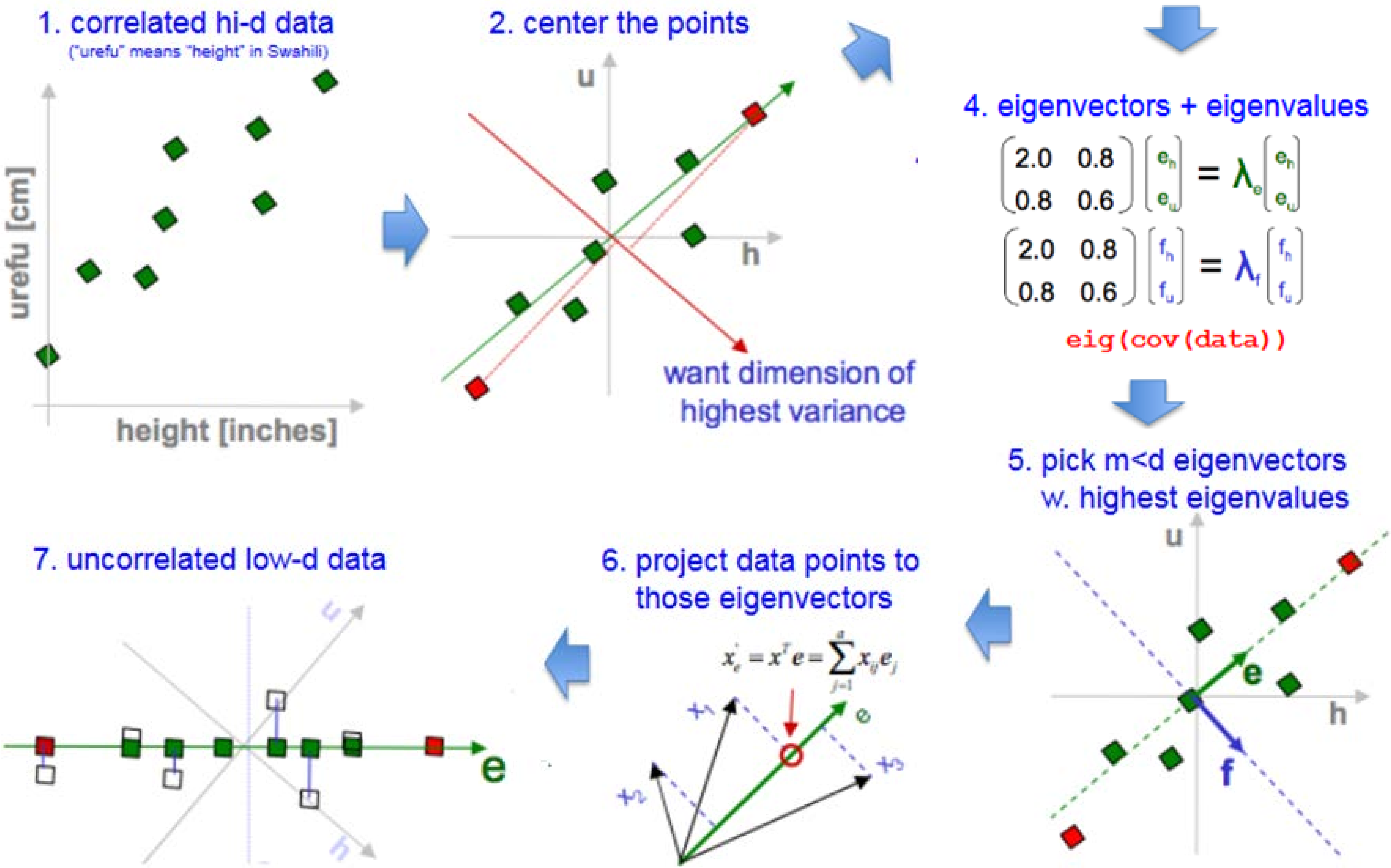
TU Delft

Principal Components Analysis

- Pros
  - reflects our intuitions about the data
  - dramatic reduction in size of data
    - faster processing (as long as reduction is fast), smaller storage
- Cons
  - too expensive for many applications (Twitter, web)
  - understand assumptions behind the methods (linearity etc.)

PCA in a nutshell





Recap

- Dimensionality reduction builds a condensed data representation
- This removes redundant or noisy features, and identifies correlations
- Principal components analysis projects data onto the principal eigenvectors of the covariance matrix: maximizes variance of the projection

PCA demo

- http://setosa.io/ev/principal-component-analysis/