http://poloclub.gatech.edu/cse6242

CSE6242: Data & Visual Analytics

Principal Component Analysis

Duen Horng (Polo) Chau

Associate Professor, College of Computing

Associate Director, MS Analytics

Georgia Tech

Mahdi Roozbahani

Lecturer, Computational Science & Engineering, Georgia Tech

Founder of Filio, a visual asset management platform

Partly based on materials by

Professors Guy Lebanon, Jeffrey Heer, John Stasko, Christos Faloutsos

Outline

- Overview
- Principle Component Analysis: Main Idea
- The PCA Algorithm
- PCA and SVD
- Summary

Motivating Example: Data Visualization

53 blood and urine samples (features) from 65 people

- Matrix format (65x53)

Instances

	H-WBC	H-RBC	H-Hgb	H-Hct	H-MCV	H-MCH	H-MCHC
A1	8.0000	4.8200	14.1000	41.0000	85.0000	29.0000	34.0000
A2	7.3000	5.0200	14.7000	43.0000	86.0000	29.0000	34.0000
A3	4.3000	4.4800	14.1000	41.0000	91.0000	32.0000	35.0000
A4	7.5000	4.4700	14.9000	45.0000	101.0000	33.0000	33.0000
A5	7.3000	5.5200	15.4000	46.0000	84.0000	28.0000	33.0000
A6	6.9000	4.8600	16.0000	47.0000	97.0000	33.0000	34.0000
A7	7.8000	4.6800	14.7000	43.0000	92.0000	31.0000	34.0000
A8	8.6000	4.8200	15.8000	42.0000	88.0000	33.0000	37.0000
A9	5.1000	4.7100	14.0000	43.0000	92.0000	30.0000	32.0000

Features

Difficult to see the correlations of different features

Motivating Example: Data Visualization

Is there a representation better than the coordinate axes?

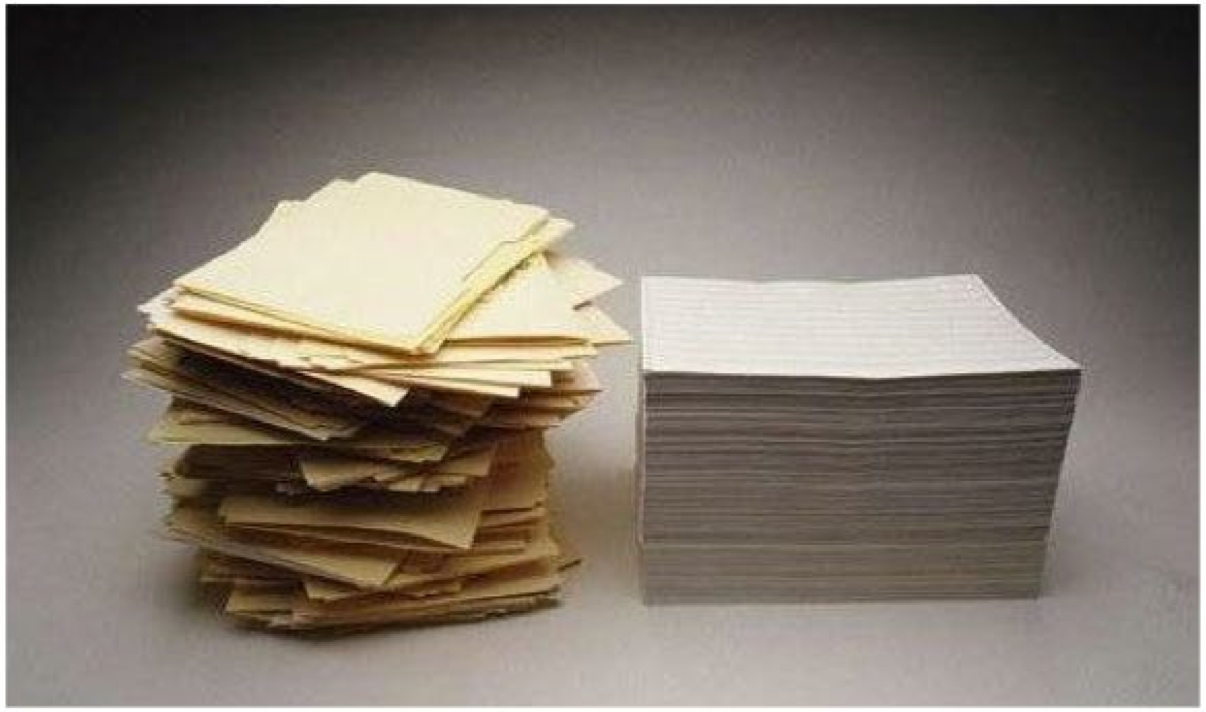
Is it really necessary to show all the 53 dimensions?

- ... what if there are strong correlations between the features?

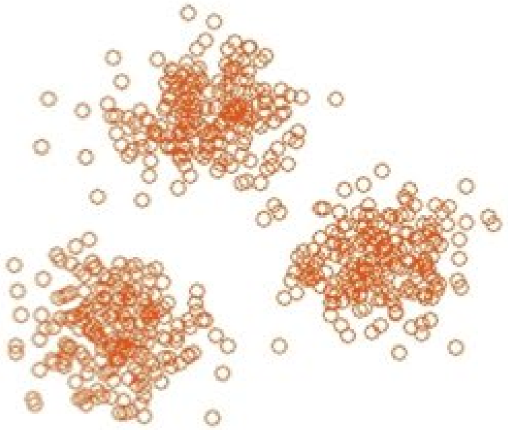
How could we find
the *smallest* subspace of the 53-D space that
keeps the *most information* about the original data?

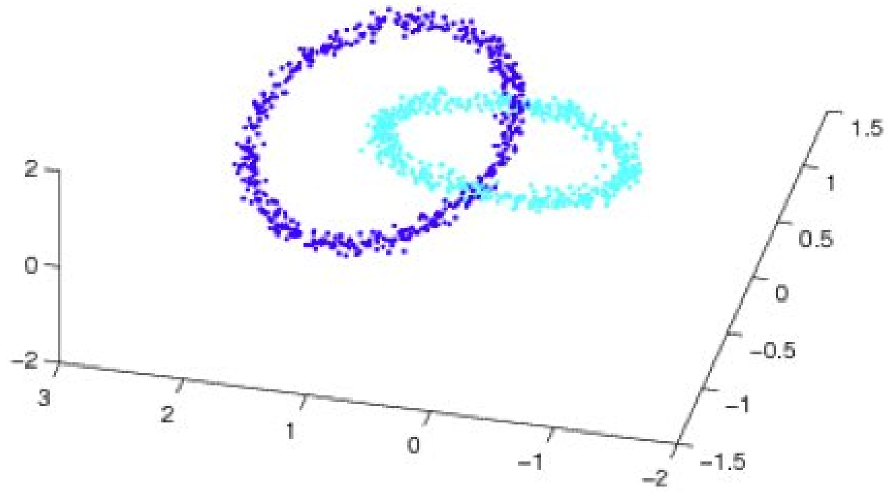
A Solution: **Dimension Reduction**

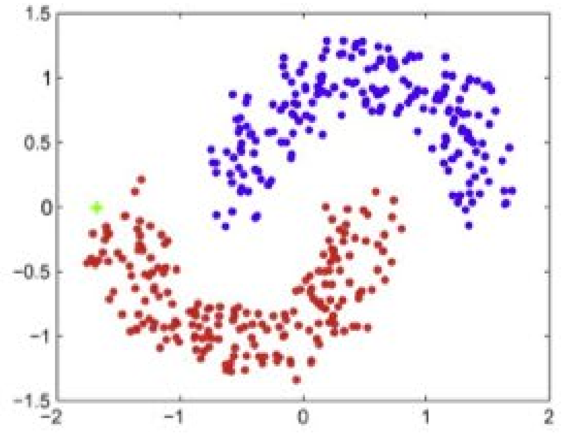
Another Example: Dimension Reduction for Text



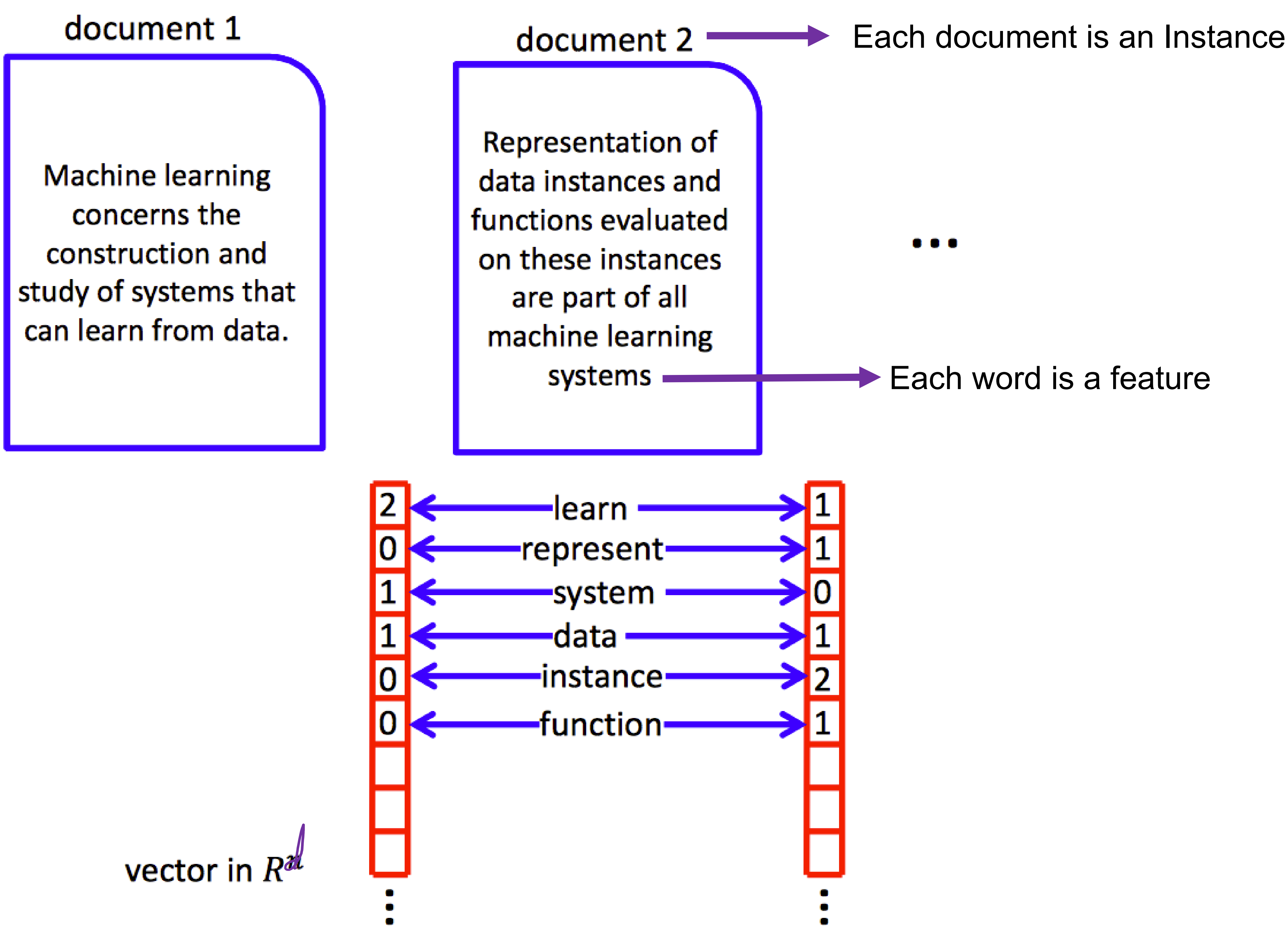
What are the relations between data points?







Bag-of-Words Representations



Term-Document Data Matrix – Bag-of-words

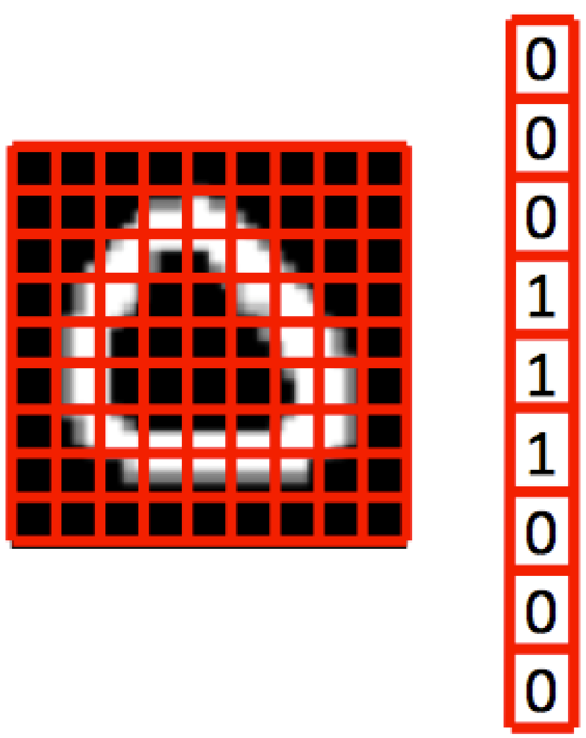
	database	SQL	index	regression	likelihood	linear
d1	24	21	9	0	0	3
d2	32	10	5	0	3	0
d3	12	16	5	0	0	0
d4	6	7	2	0	0	0
d5	43	31	20	0	3	0
d6	2	0	0	18	7	16
d7	0	0	1	32	12	0
d8	3	0	0	22	4	2
d9	1	0	0	34	27	25
d10	6	0	0	17	4	23

▪ ▪ ▪ Many more features

Solution:
Dimension Reduction

What is Dimension Reduction?

- The process of reducing the number of random variables under consideration
  - One can combine, transform or select variables
  - One can use linear or nonlinear operations



vector in R^d

Original data point

Reduced representation

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \quad f(x): R^d \mapsto R^k \quad z = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_k \end{pmatrix}$$

$$k \ll d$$

Applications of Dimension Reduction

- The dimension-reduced data can be used for
  - Visualizing, exploring and understanding the data
  - Aggregating weak signals in the data
  - Cleaning the data
  - Speeding up subsequent learning task
  - Building simpler model later

- Key questions of a dimensionality reduction algorithm
  - What is the criterion for carrying out the reduction process?
  - What are the algorithm steps?

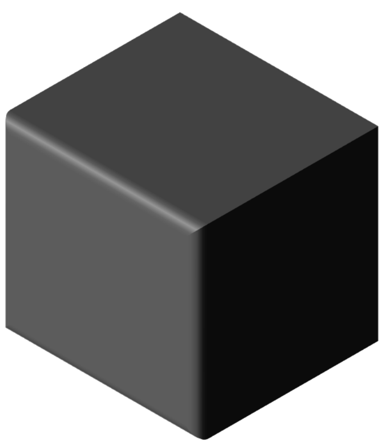
Outline

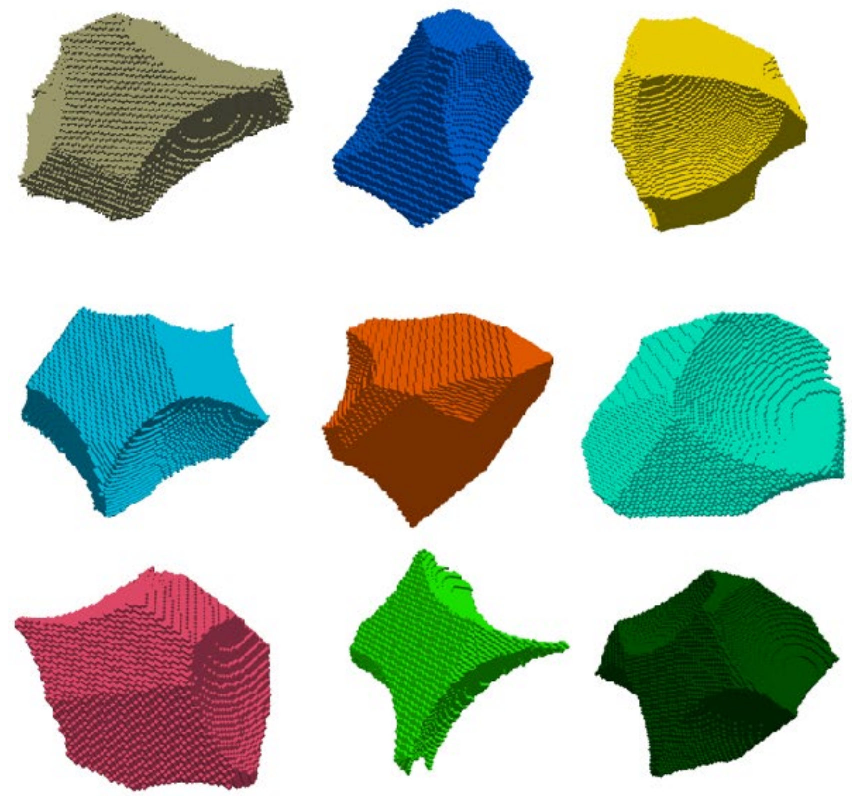
- Overview
- Principle Component Analysis: Main Idea
- The PCA Algorithm
- PCA and SVD
- Summary

Mahdi’s example

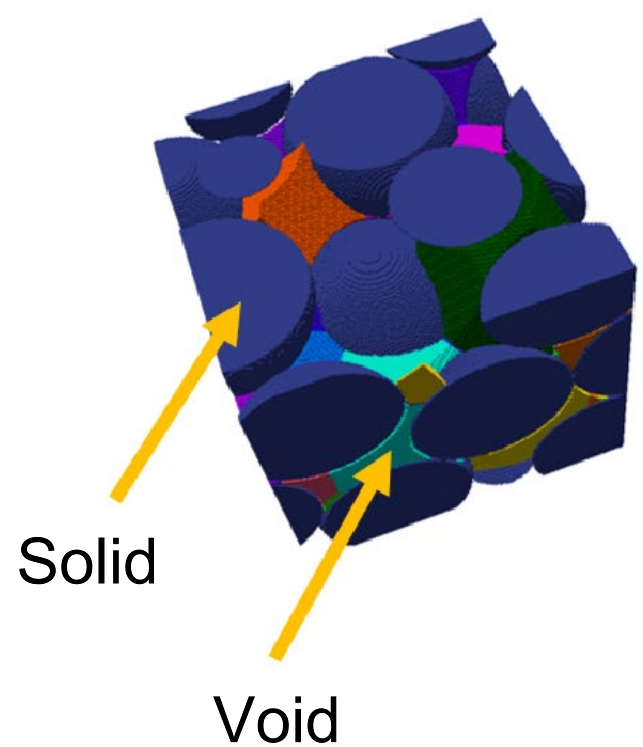
Pixel in 2D

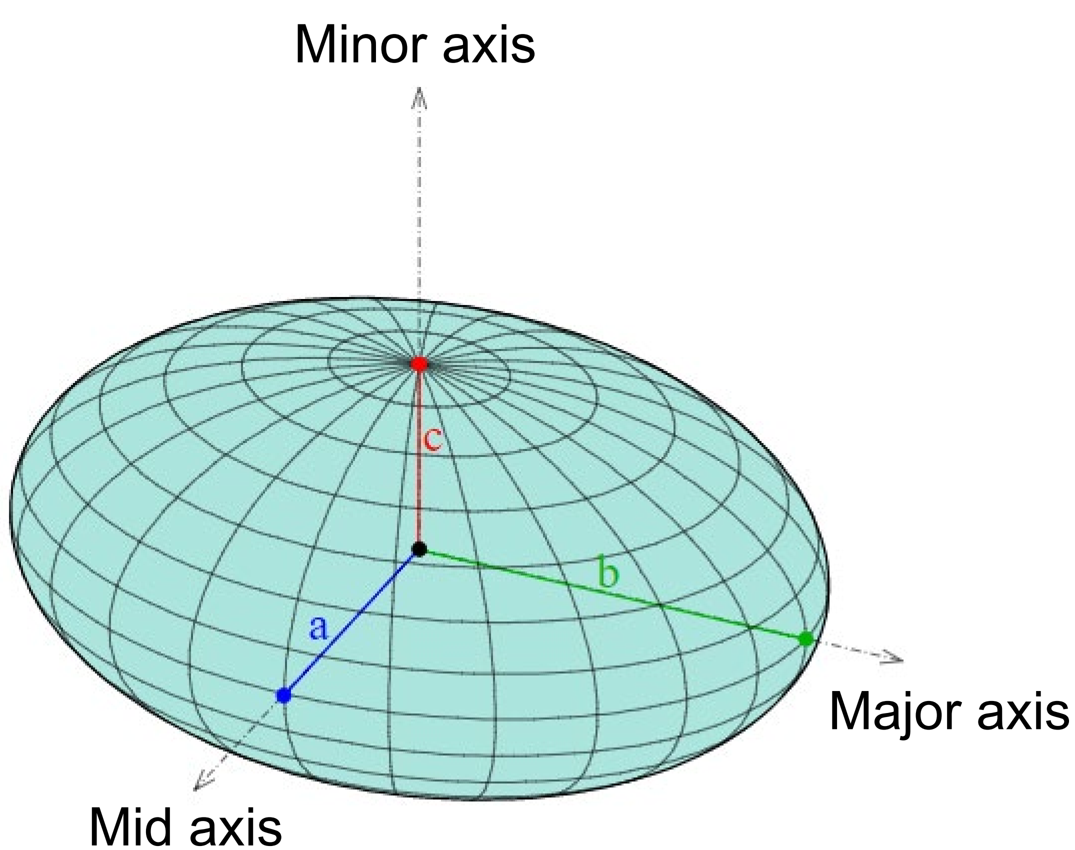
Voxel in 3D

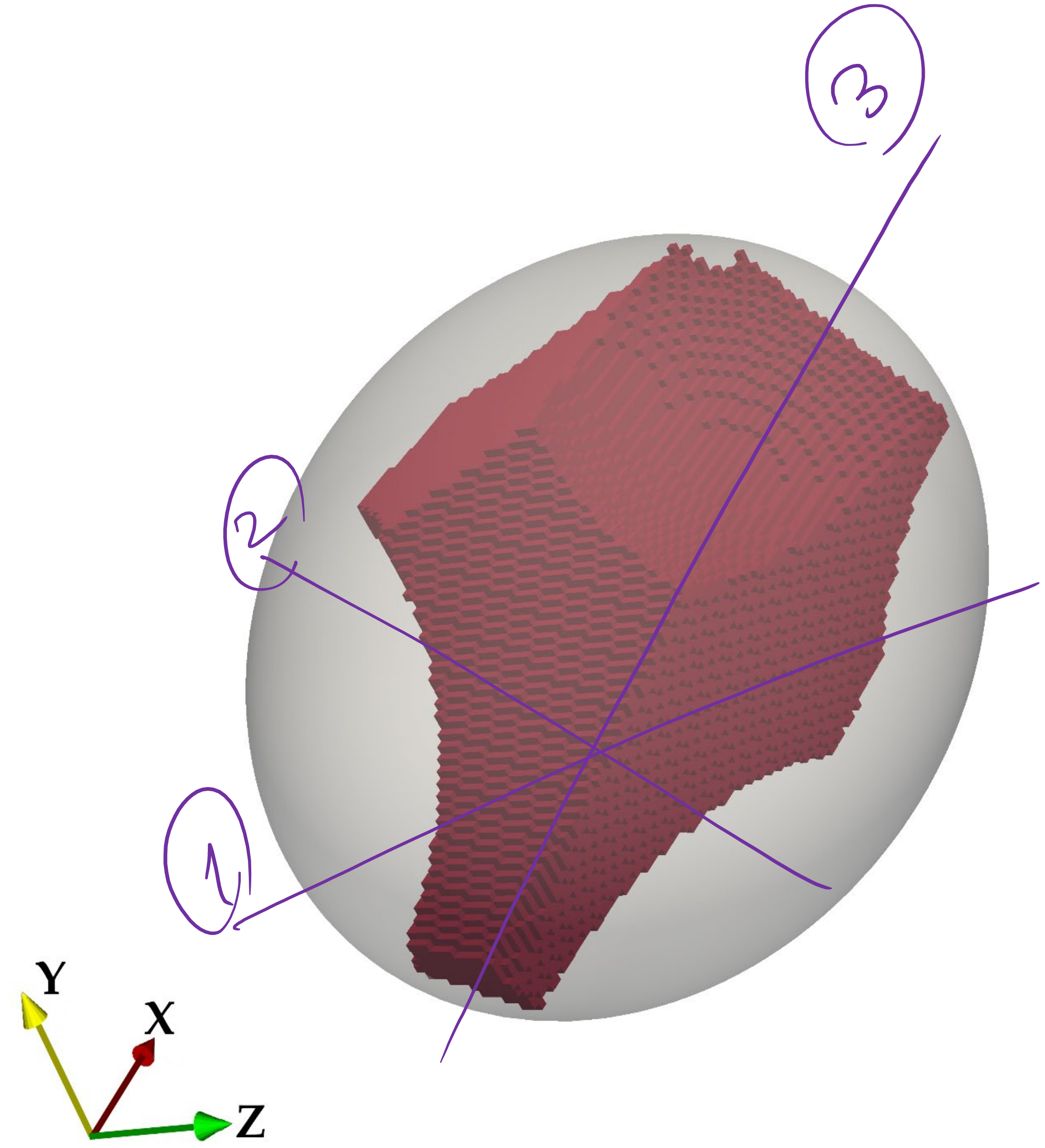


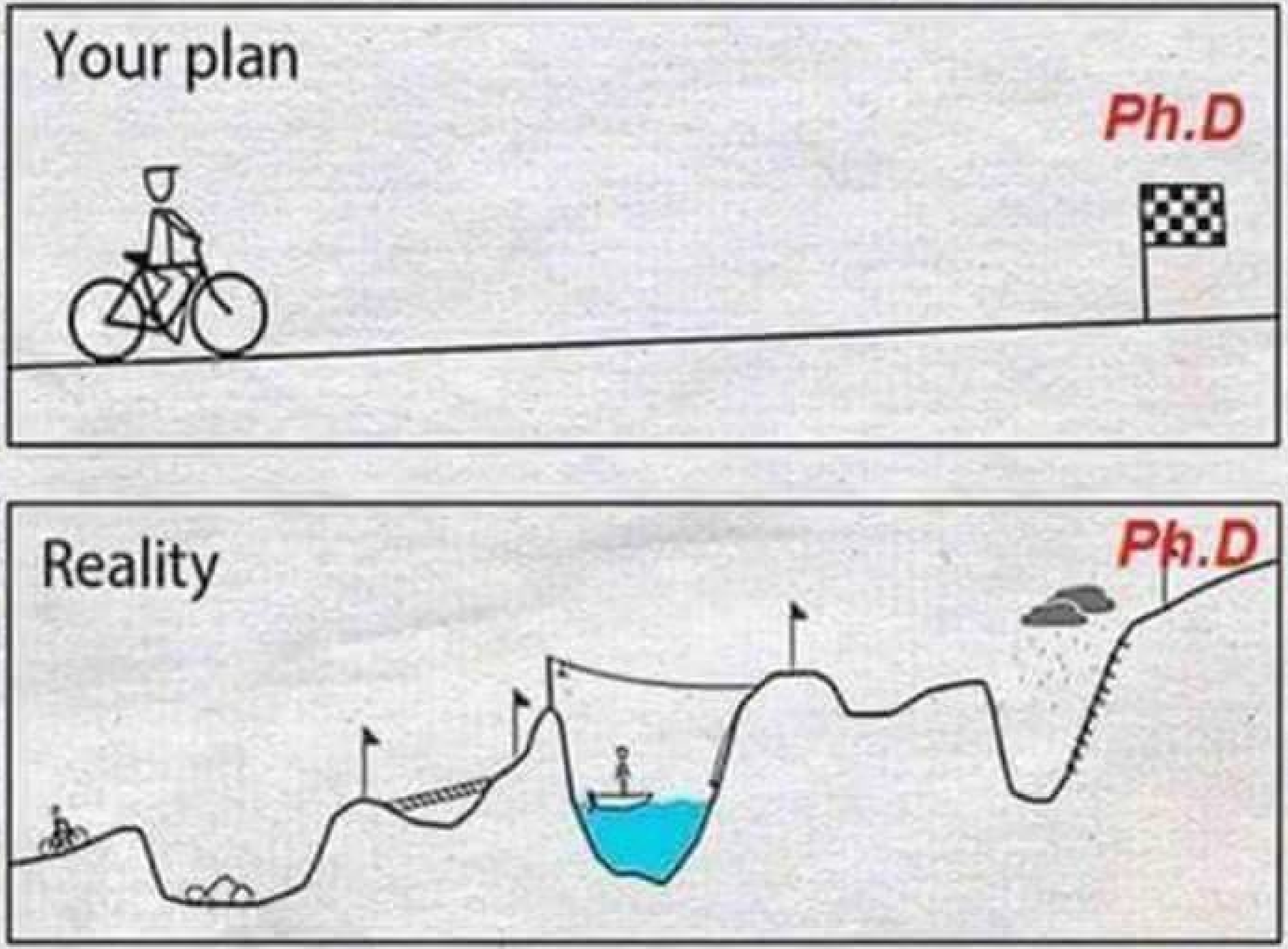


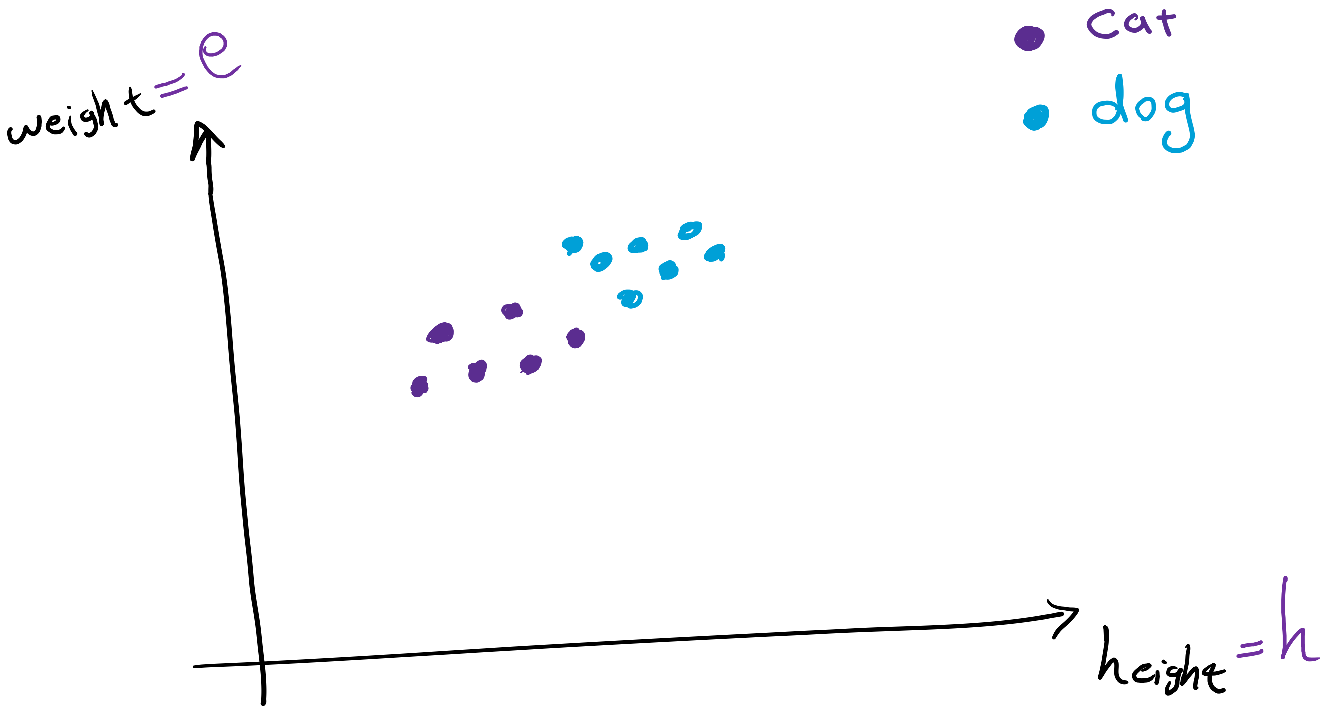
Segmented Voids



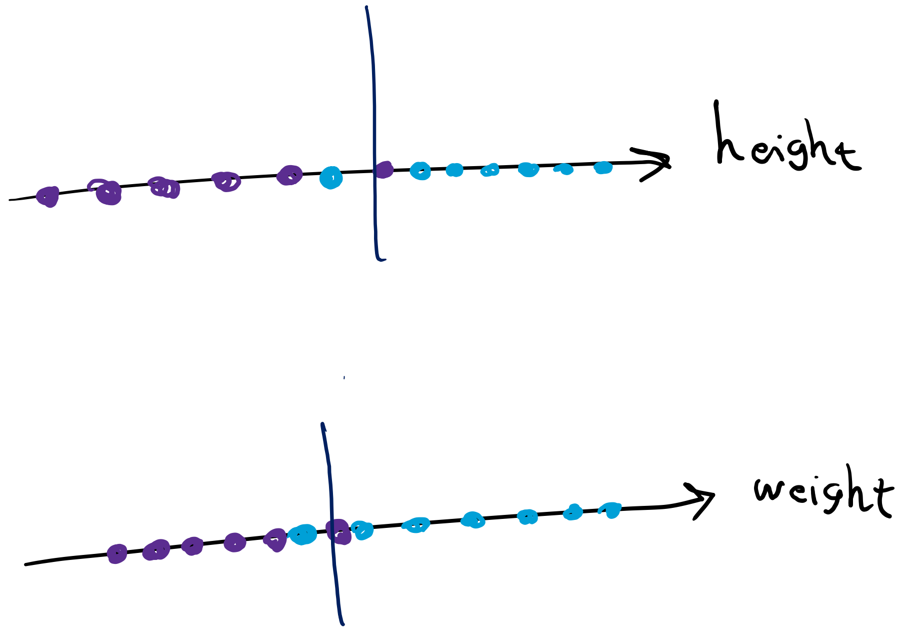

Minor axis
c
b
a
Major axis
Mid axis


3
2
1
Y
X
Z

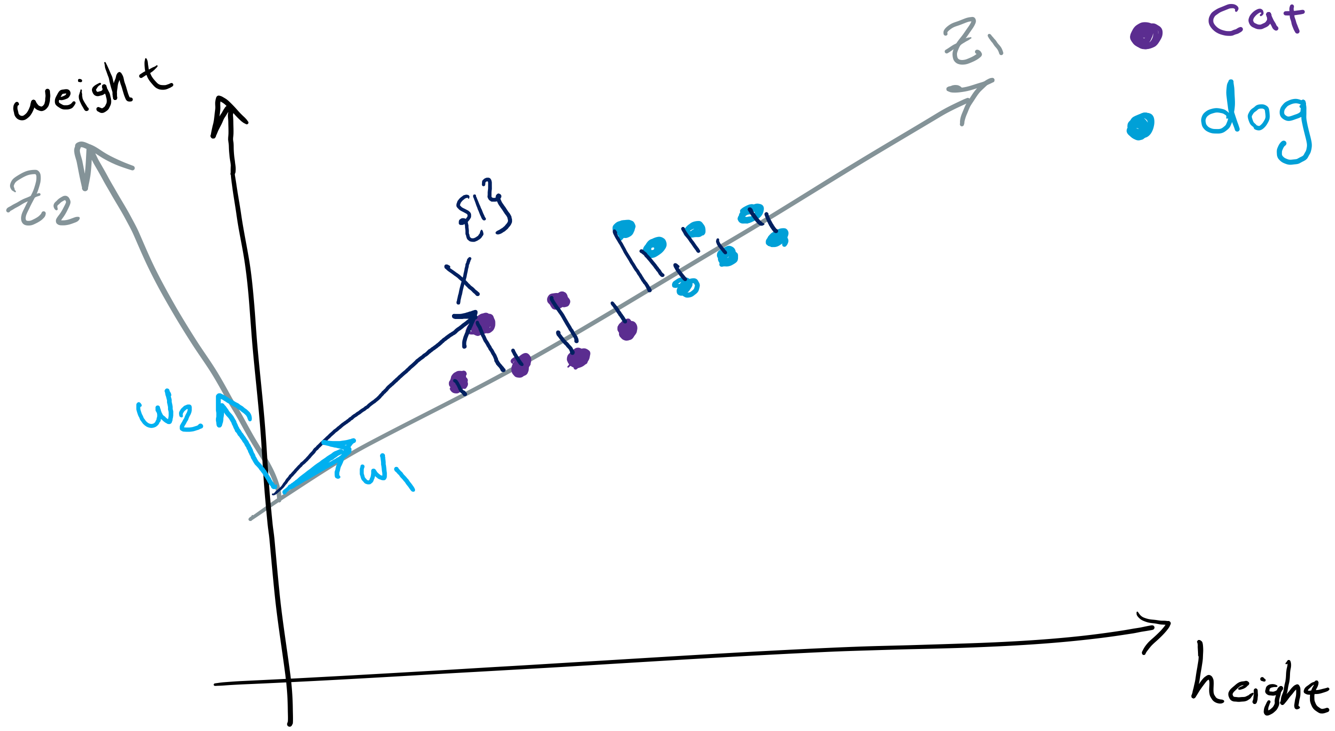

Your plan
Ph.D
Reality
Ph.D

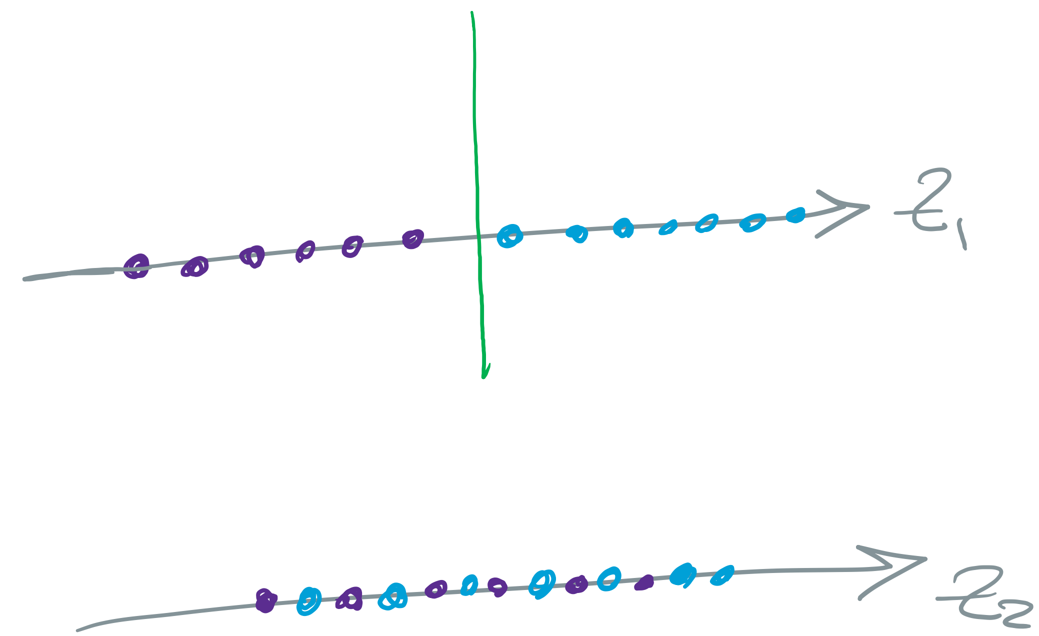


weight = e
height = h
cat
dog

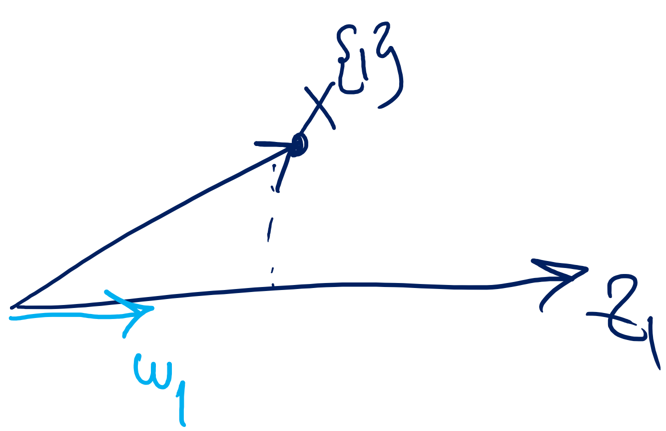


height
weight



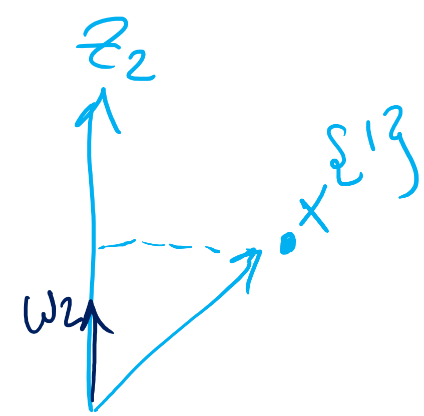


$$z_2^{\{1\}} = X^{\{1\}} \cdot w_2$$

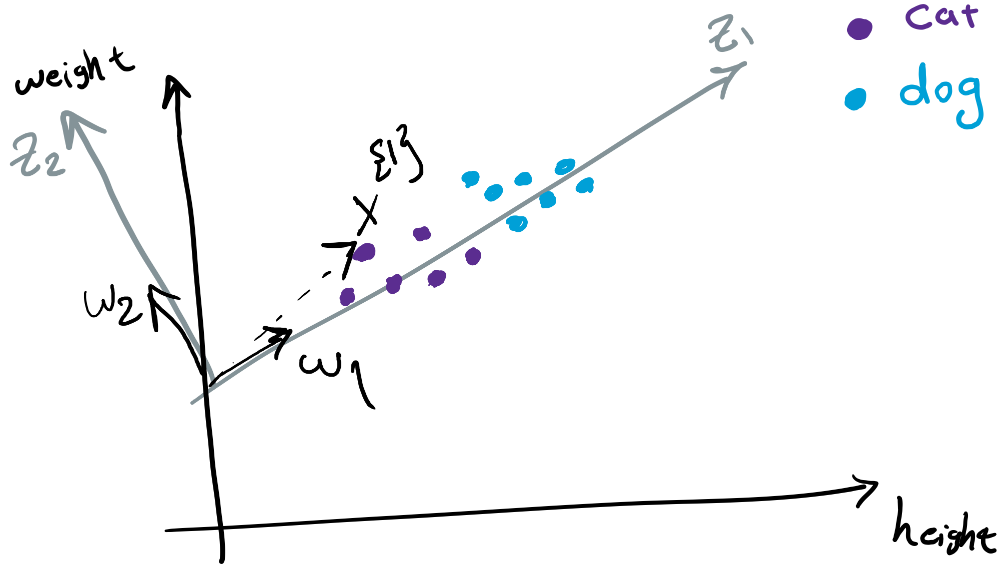


$\cancel{X^{\{1\}} \cdot z_1}$

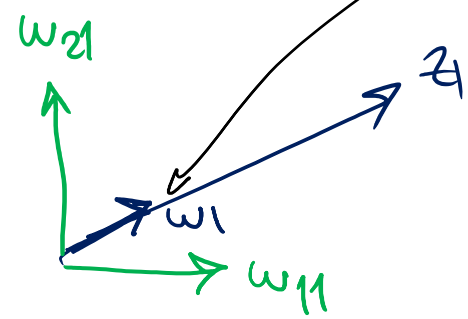
$$z_1^{\{1\}} = X^{\{1\}} \cdot w_1$$



$$||w|| = 1$$



$$X = \begin{array}{cc} h & w \\ \hline \end{array}\begin{bmatrix} & \\ & \end{bmatrix} \qquad Z = \begin{array}{cc} z_1 & z_2 \\ \hline \end{array}\begin{bmatrix} & \\ & \end{bmatrix}$$



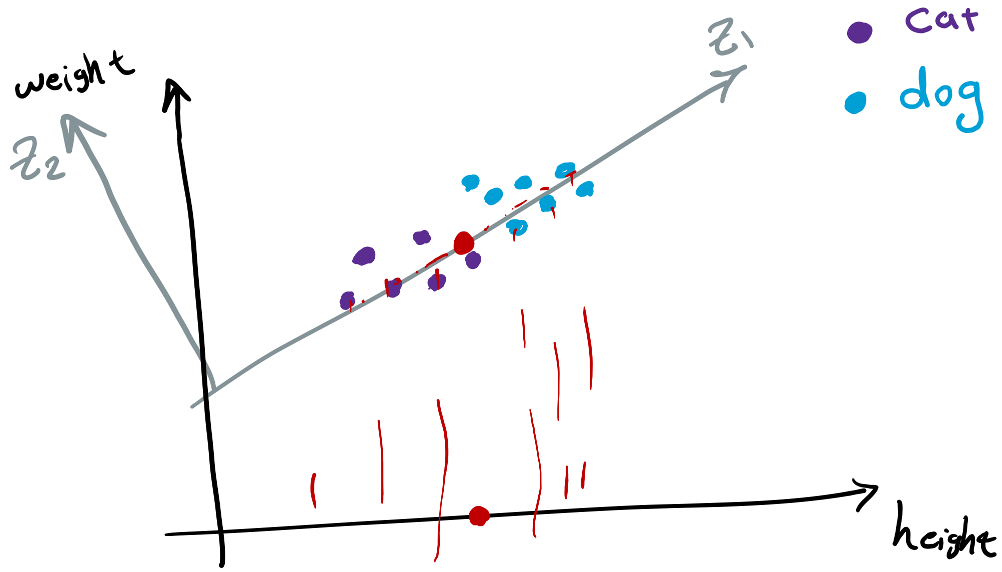
$$W = \begin{array}{cc} w_1 & w_2 \\ \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} \end{array}$$

$$X^{\{1\}} \longrightarrow z^{\{1\}}$$

$$[h \quad e] \qquad \cdot \; [z_1 \quad z_2]$$

$$z_1 = X^{\{1\}} \cdot w_1 = [h \quad e]\begin{bmatrix} w_{11} \\ w_{21} \end{bmatrix} = h w_{11} + e w_{21}$$

$$z_2 = X^{\{1\}} \cdot w_2 = [h \quad e]\begin{bmatrix} w_{12} \\ w_{22} \end{bmatrix} = h w_{12} + e w_{22}$$

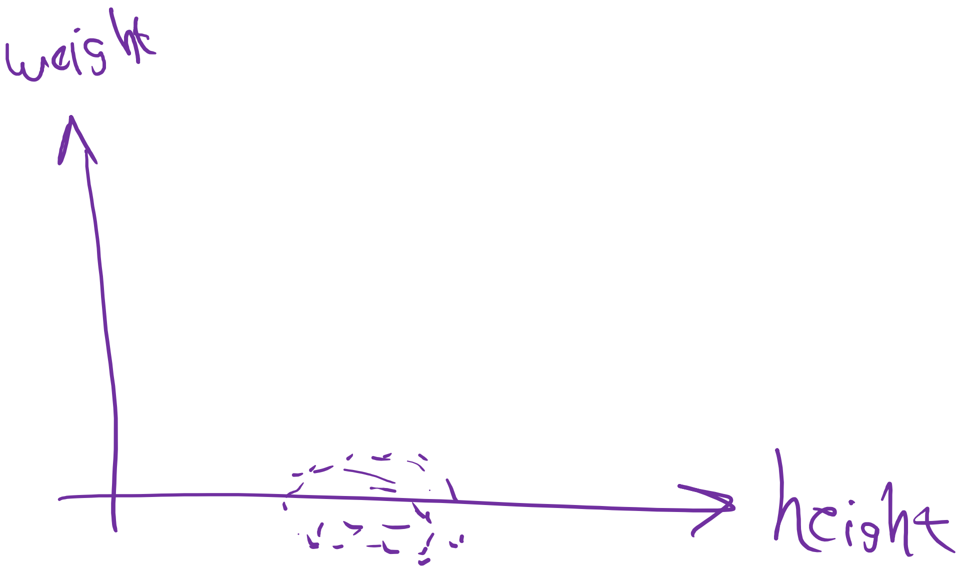


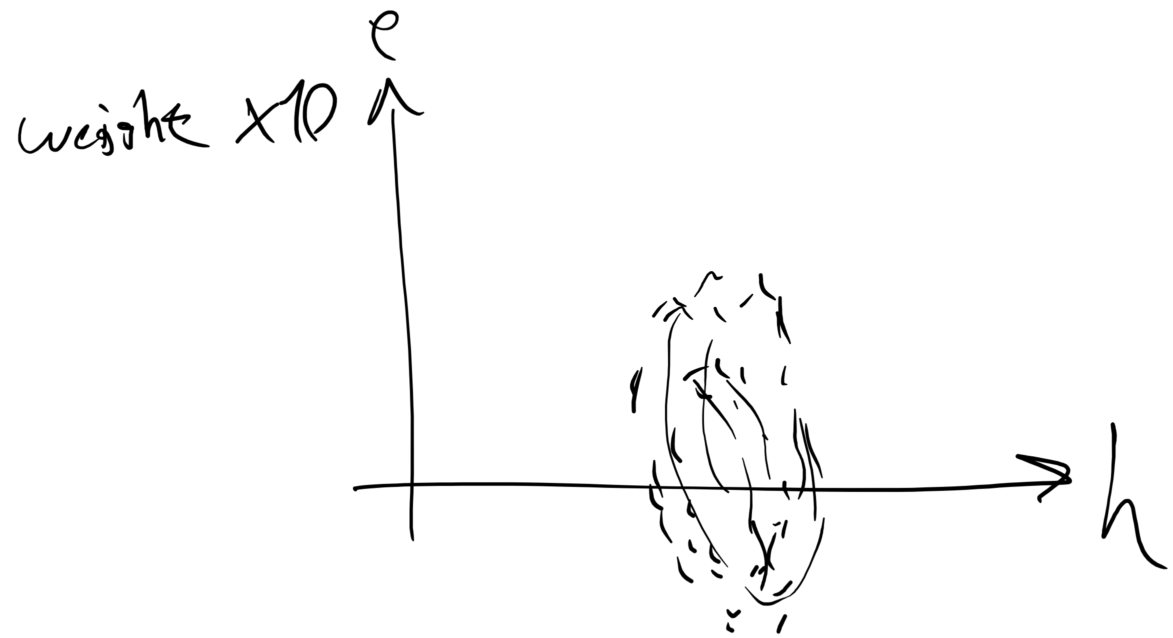
$$Var(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2$$

$$\text{Max } Var(z) = \frac{1}{N}\sum_{i=1}^{N}(z_i - \mu_z)^2$$

$$= \frac{1}{N}\sum_{i=1}^{N}(x_i w - \mu w)^2$$

$$\|w\| = 1$$


weight
height


weight x10
h

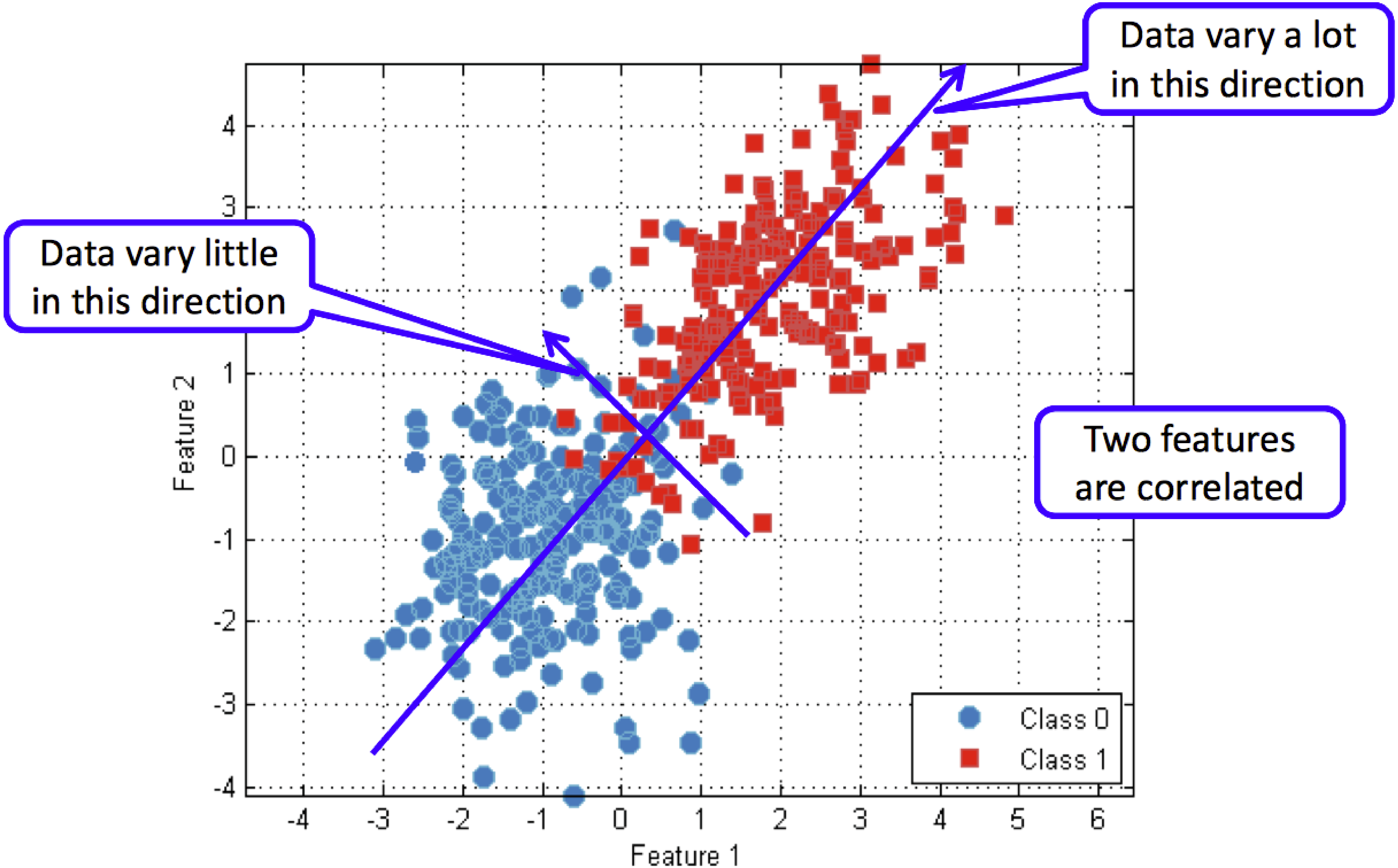
PCA: Dimension Reduction by Capturing Variation

- There are many criteria (geometric based, information theory based, etc.)

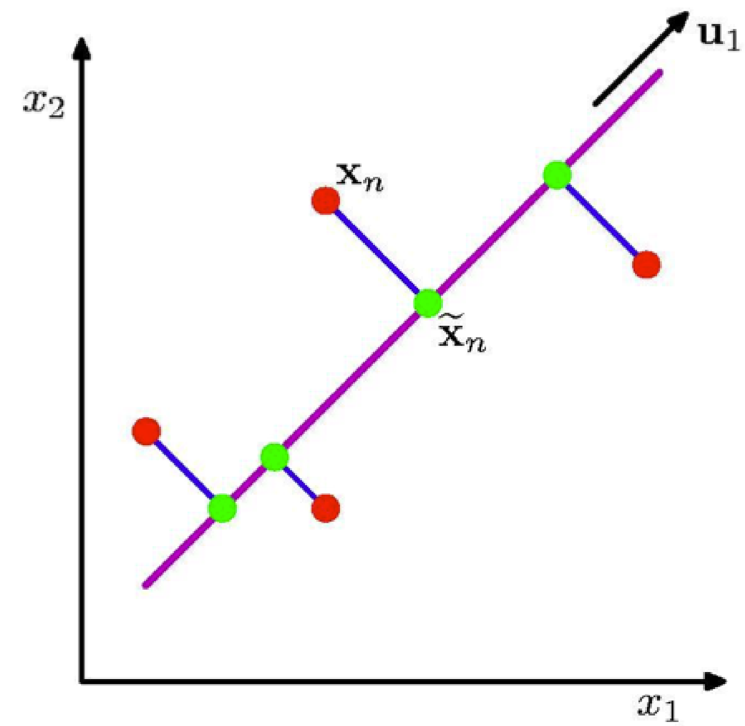
- One criterion: want to capture variation in data
  - variations are "signals" or information in the data
  - need to normalize each variables first

- In the process, also discover variables or dimensions highly correlated
  - represent highly related phenomena
  - combine them to form a stronger signal
  - lead to simpler presentation

Capturing Variation in Data



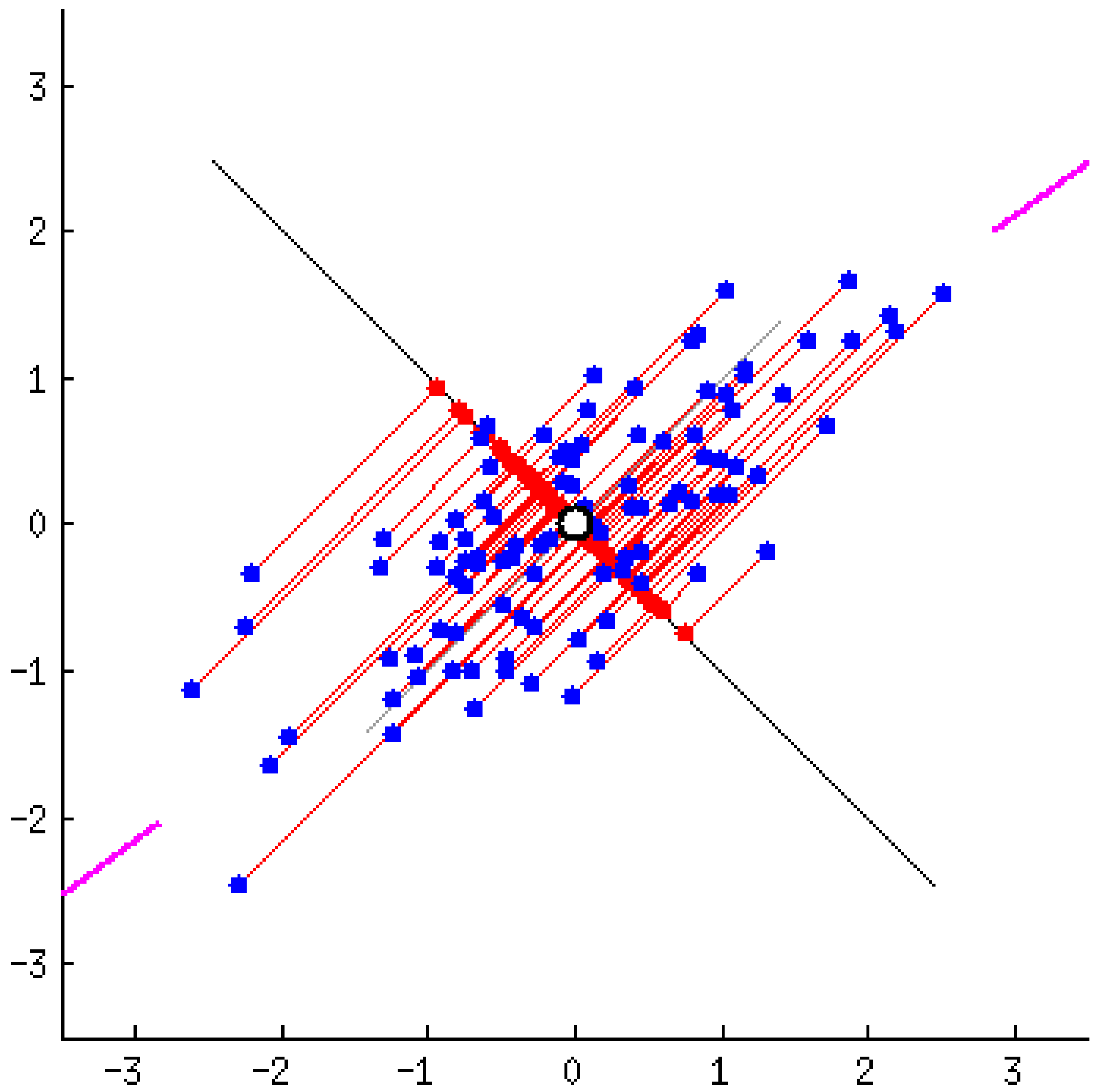
Two Equivalent Perspectives of PCA



PCA:

Orthogonal projection of the data onto a lower-dimension linear space that...

- ❑maximizes variance of projected data (purple line)
- ❑minimizes mean squared distance between
  - data point and
  - projections (sum of blue lines)


3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3

Outline

- Overview
- Principle Component Analysis: Main Idea
- The PCA Algorithm
- PCA and SVD
- Summary

What is variance equation?

$$Var(x) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2$$

Formulating the Problem

- Given n data points, $\{x_1, x_2, \ldots, x_n\} \in R^d$ with their mean $\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$
- Find a direction $w \in R^d$ where

$$\|w\| = \sqrt{\sum_{j \in d} \omega_j^2} = 1$$

We constrain the norm of w to be equal to one to avoid having very large variance in each new dimension.

- Given n data points, $\{x_1, x_2, \dots, x_n\} \in R^d$ with their mean μ

$$\|w\| = \sqrt{\sum_{j \in d} \omega_j^2} = 1 \qquad\qquad \mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

- Such that the variance (or variation) of the data along direction w is maximized

$$\max_{||w||=1} \underbrace{\frac{1}{n}\sum_{i=1}^{n}(x_i w - \mu w)^2}_{\text{variance in new feature space}}$$

An Optimization Problem

- Manipulate the objective with linear algebra

$$\frac{1}{n}\sum_{i=1}^{n}(x_i w - \mu w)^2 \quad = \frac{1}{n}\sum_{i=1}^{n}((x_i - \mu) w)^2 \ =$$

$$= \frac{1}{n}\sum_{i=1}^{n}\Big((\underbrace{x_i - \mu}_{A})\underbrace{w}_{B}\Big)^T((x_i - \mu)w) = \frac{1}{n}\sum_{i=1}^{n} w^T (x_i - \mu)^T \, (x_i - \mu) w$$

$$(AB)^T = B^T A^T$$

$$w^T \boxed{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^T \, (x_i - \mu)\right)} w \qquad = w^T C w$$

Covariance matrix

Equivalence to The Eigenvalue Problem

- Claim:

$$\max_{||w||=1} w^T C w$$

$w^T w$

$g(w) = 1 - ||w||$

- Form lagrangian function of the optimization problem

$$L(w, \lambda) = w^{\mathsf{T}} C w + \lambda(1 - w^t w)$$

- If w is a maximum of the original optimization problem, then there exists a λ, where (w, λ) is a stationary point of $L(w, \lambda)$

- This implies that

$Ax = \delta x$

$$\frac{\partial L}{\partial w} = 0 = 2Cw - 2\lambda w \quad \Rightarrow Cw = \lambda w$$

Eigen-Value Problem

- Eigen-value problem

d: dimension

  - Given a symmetric matrix $C \in R^{d\times d}$

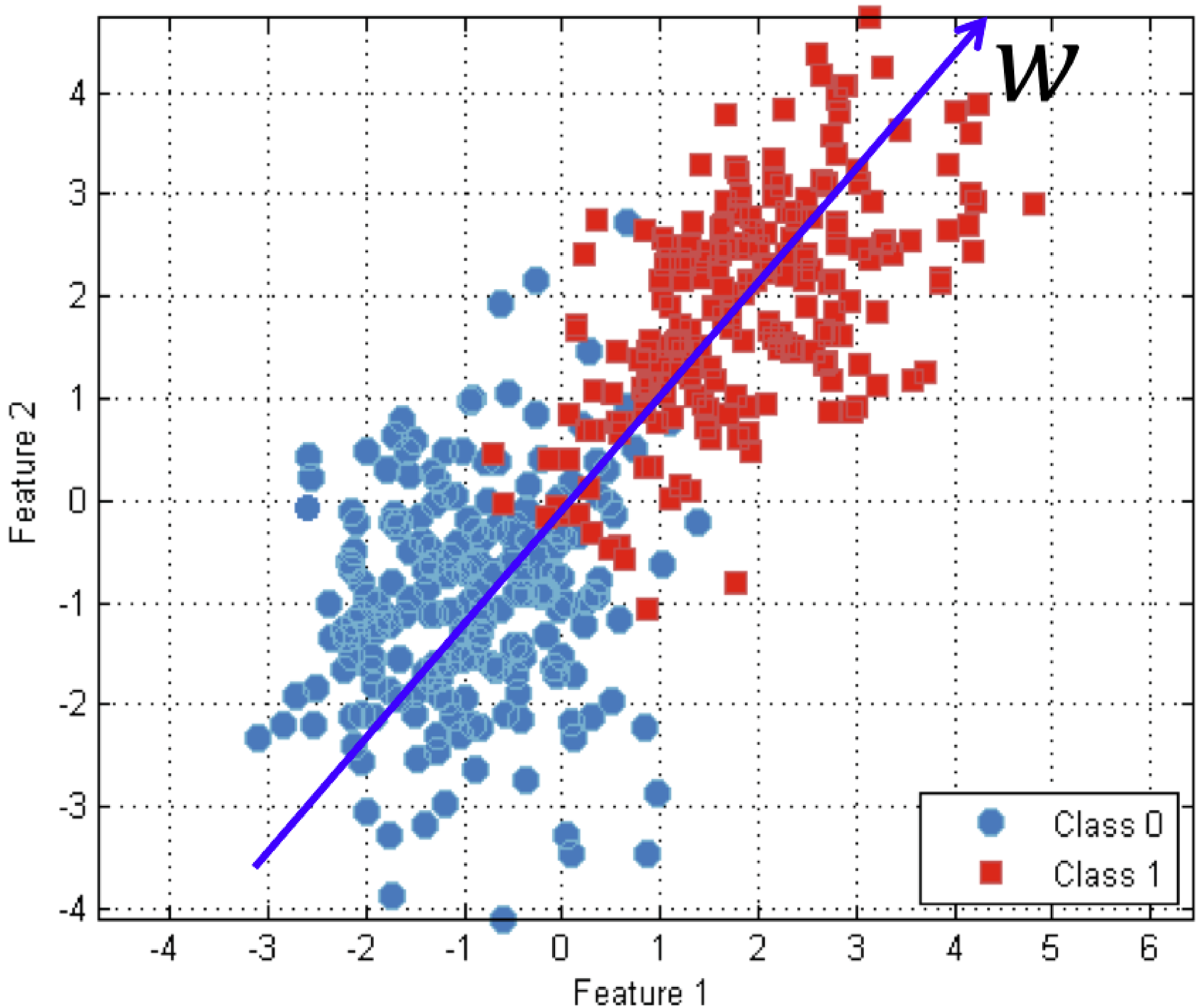
C is also a positive semidefinite matrix

  - Find a vector $w \in R^d$ and $\|w\| = 1$
  - Such that

$$Cw = \lambda w$$

- There will be multiple solution of $w_1, w_2, \dots, w_d$ for its corresponding $\lambda_1, \lambda_2, \dots, \lambda_d$
  - They are ortho-normal: $w_i^T w_i = 1 \qquad w_i^T w_j = 0$

Principal Direction of the Data



Variance in the Principal Direction

- Principal direction w satisfies

$$Cw = \lambda w = w\lambda$$

- Variance in principal direction is

$$w^T C w \rightarrow = \mathrm{Var}(z) = \frac{1}{N}\sum_{i=1}^{N}(x_i w - \mu_w)^2$$

$$= w^T w \lambda \rightarrow$$

$$= \lambda$$

eigen-value

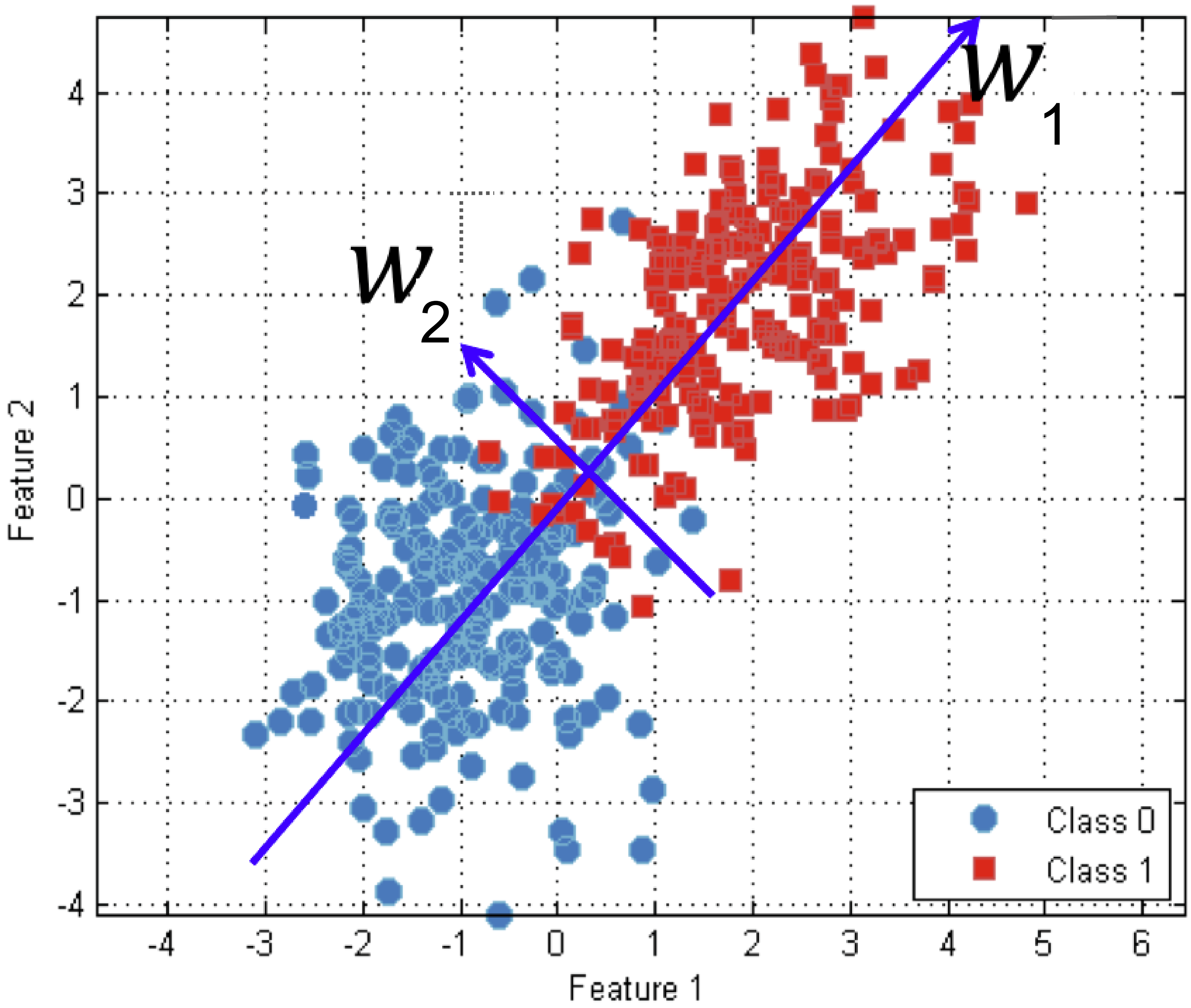
$$\begin{array}{cccc} w_1 & w_2 & \cdots & w_d \end{array}$$

$$X w_1 = z_1$$

Multiple Principal Directions

- Directions $w_1, w_2, \ldots$ which has
  - the largest variances
  - but are orthogonal to each other
- Take the eigenvectors $w_1, w_2, \ldots$ of C corresponding to
  - the largest eigenvalue λ_1,
  - the second largest eigenvalue λ_2
  - …

Extra Principal Directions



Relations Between Principal Components

Principal component #1: points in the direction of the **largest variance**.

Each subsequent principal component

- is **orthogonal** to the previous ones, and
- points in the directions of the **largest variance of the residual subspace**

The PCA Algorithm

- Given n data points, $\{x_1, x_2, \dots, x_n\} \in R^d$ with mean

- Step 1: Estimate the mean and covariance matrix from data

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i \quad and \quad C = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^T (x_i - \mu)$$

Principal directions

- Step 2: Take the eigenvectors w_1, w_2 , ... of C corresponding to the largest eigenvalue λ_1, the second largest eigenvalue λ_2 ...

- Step 3: Compute reduced representation

$$z_i = \left(\frac{(x_i - \mu_1)}{\sigma_1} w_1 \quad \frac{(x_i - \mu_2)}{\sigma_2} w_2 \quad \dots \right)$$

$z \Rightarrow \mathrm{n} \times k$

$k \ll d$

Normalizing by standard deviation

Outline

- Overview
- Principle Component Analysis: Main Idea
- The PCA Algorithm
- PCA and SVD
- Summary

Singular Value Decomposition

$$X_{n \times d}$$

n: instances
d: dimensions
X is a centered matrix

$$X = U\Sigma V^T$$

$$U_{n \times n} \rightarrow unitary\ matrix \rightarrow U \times U^T = I$$

$$\Sigma_{n \times d} \rightarrow diagonal\ matrix$$

$$V_{d \times d} \rightarrow unitary\ matrix \rightarrow V \times V^T = I$$

Principle direction

$$X = \begin{bmatrix} u_{1\times1} & \cdots & \cdots & \cdots & u_{1\times n} \\ \vdots & \ddots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \ddots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \ddots & \vdots \\ u_{1\times1} & \cdots & \cdots & \cdots & u_{n\times n} \end{bmatrix} \times \begin{bmatrix} \sum_{1\times1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sum_{d\times d} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} \boxed{v_{1\times1} \quad \cdots \quad \cdots \quad \cdots \quad v_{1\times d}} \\ \vdots & \ddots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \ddots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \ddots & \vdots \\ v_{d\times1} & \cdots & \cdots & \cdots & v_{d\times d} \end{bmatrix}$$

U Σ V^T

nxn nxk

$$d < n$$

Matrix compression: K dimensions out of d

$U\Sigma = n \times k$

According to PCA ➔ $Cw = \lambda w = w\lambda$

$$\text{Covariance } C_{d\times d} = \frac{1}{n}\sum_{i=1}^{n}\left(x^i-\mu\right)^T \underbrace{\left(x^i-\mu\right)}^{\textbf{Centering X}} = \frac{X^T X}{n}$$

$$\left.\begin{aligned} X &= U\Sigma V^T \\ C &= \frac{X^T X}{n} \end{aligned}\right\} C = \frac{V\Sigma^T U^T U\Sigma V^T}{n} = \frac{V\Sigma^2 V^T}{n}$$

$$C = \frac{V\Sigma^2 V^T}{n} = V\frac{\Sigma^2}{n}V^T$$

$CV = V\frac{\Sigma^2}{n}$

$CW = W\Lambda$

$$CV = V\frac{\Sigma^2}{n}V^TV = V\frac{\Sigma^2}{n}$$

According to Eigen-decomposition definition ➔ $CV = V\Lambda$

V is the eigen vectors of covariance (Principal directions)

$\lambda_i = \frac{\sigma_i^2}{n}$ ➔ The eigenvalues of covariance matrix

Let's project the data (X) on principal directions:

XW

$$XV = U\Sigma V^TV = U\Sigma$$

$\boldsymbol{XV}$ is independent linear combinations of the original data

Projection of one instance (x) on the first principal direction using k dimensions

$$\mathrm{p}_1 = [u_{1\times 1}\Sigma_{1\times 1}, u_{1\times 2}\Sigma_{2\times 2}, \dots, u_{1\times k}\Sigma_{k\times k}]$$

$$\mathrm{p}_2 = [u_{2\times 1}\Sigma_{1\times 1}, u_{2\times 2}\Sigma_{2\times 2}, \dots, u_{2\times k}\Sigma_{k\times k}]$$

$U \Rightarrow n \times k$

$\Sigma \Rightarrow k \times k$

Upper left corner

Eigen values $\lambda = \frac{\Sigma^\wedge 2}{m}$

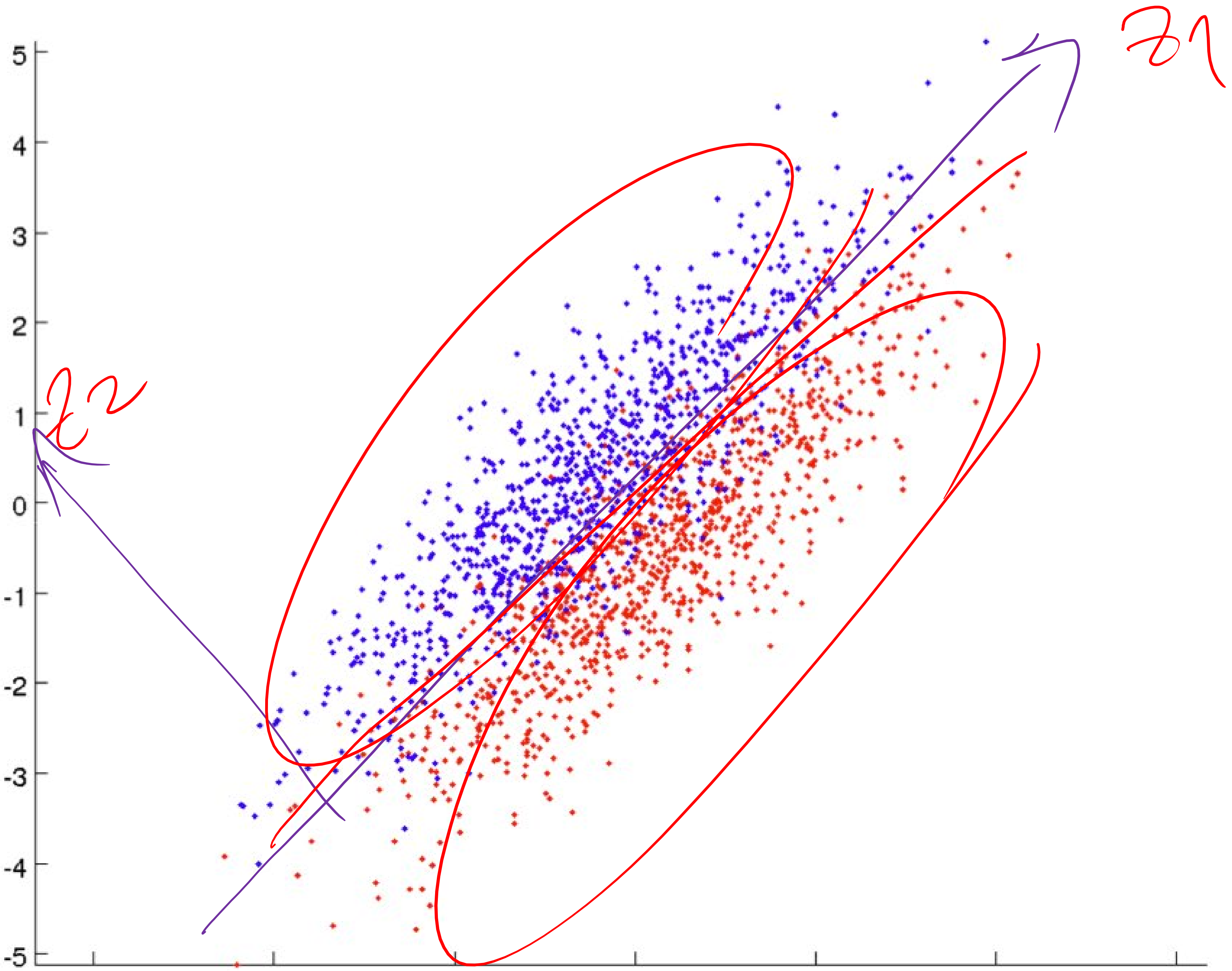
Eigenvectors (principal directions) V

$$X = \underline{U\Sigma} V^T$$

Principal components (Scores) or projections on principal directions

In fact, using the SVD to perform PCA makes much better sense numerically than forming the covariance matrix to begin with, since the formation of $X^T X$ can cause loss of precision.

Are Principal Components Good for Classification?



Why PCA potentially works in classification?

the dimension with the largest variance corresponds to the dimension with the largest entropy and thus encodes the most information (Information Theory). The smallest eigenvectors will often simply represent noise components, whereas the largest eigenvectors often correspond to the principal components that define the data.

Outline

- Overview
- Principle Component Analysis: Main Idea
- The PCA Algorithm
- PCA and SVD
- Summary

Summary

- PCA
  - Finds orthonormal basis for data
  - Sorts dimensions in order of “importance”
  - Discard low significance dimensions
- Uses
  - Get concise low-dimensional representations
  - Remove noise
- Not magic
  - Doesn’t know class labels
  - Can only capture linear variations

Image compression using PCA

$$X = \underbrace{U}\,\underbrace{\Sigma}_{n\times d}\,\underbrace{V^T_{d\times d}}$$

$$V_{n\times 10}$$

$$X_{n\times d} = U_{n\times n}\,\Sigma_{n\times 10}\,V^T_{10\times d}$$