Machine Learning Fundamentals – DTSC102

Lecture 7
ML Diagnostics + PCA

Course Instructor: Dr.-Ing. Maggie Mashaly
maggie.ezzat@guc.edu.eg
C3.220

Contents

Regularization

- So far we agreed that more data is always better…
- Are more features always better too?

Let's weight the options:

YES because:

- Better fitting accuracy, i.e.: better prediction

No because:

- There is a strong risk of overfitting, i.e.: learning something expressive rather than something general

So it is a trade-off that we need to balance, how to??

Regularization

Occam's Razor

- A principle attributed to the 14th century English logician William of Ockham, states that:

"All other things being equal, the simplest solution is the best"

In other words:

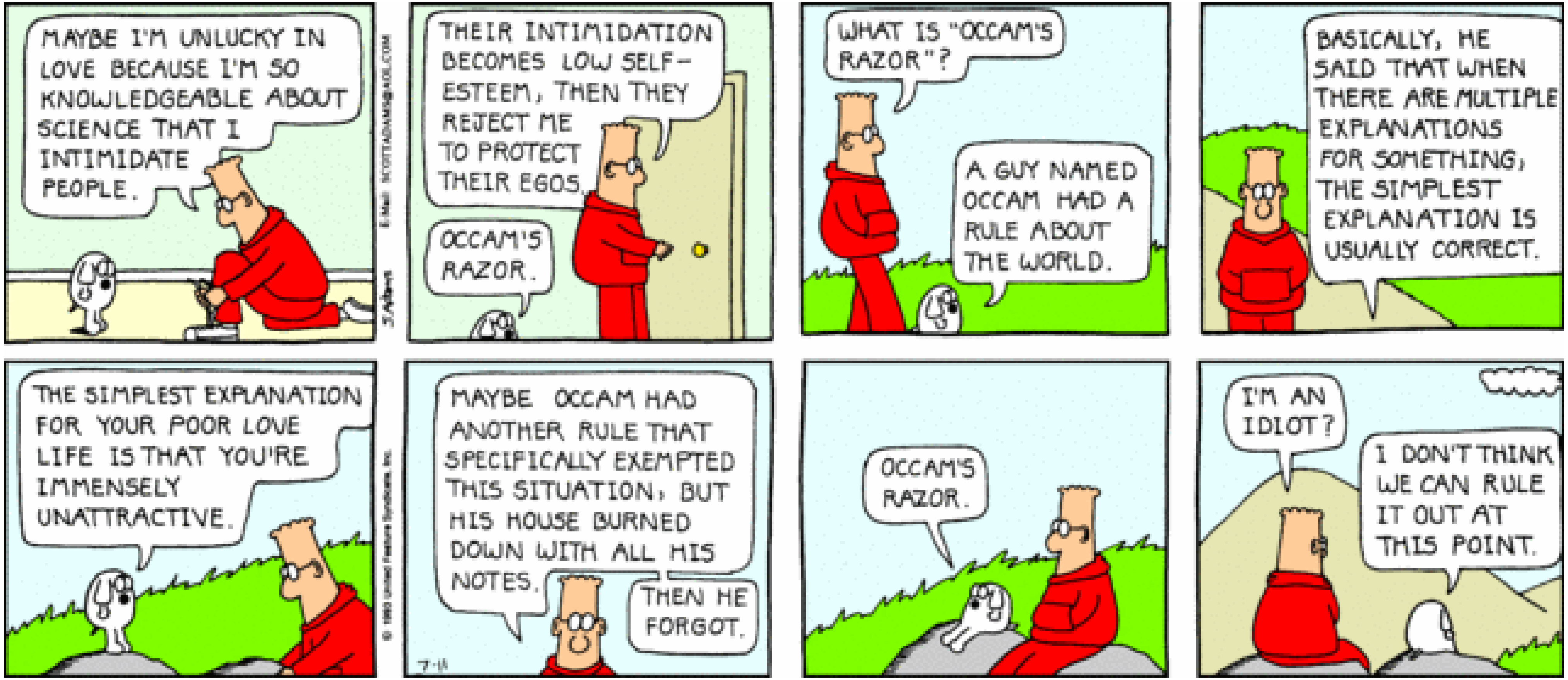
➢ When multiple competing hypothesis are equal in other aspects, select the hypothesis that introduces the fewest assumptions and postulates the fewest entities

In fewer words:

➢ Prefer the simplest hypothesis that fits the data

Regularization

Occam's Razor



Dr.- Ing. Maggie Mashaly

Regularization

- Prefer the simplest hypothesis that fits the data: Regularization
- The idea is to add a “Penalty Term” that increases with the complexity of the hypothesis to the optimization problem

Thus:

- Complex Hypothesis incur High penalty and thus are rejected

Unless:

- Complex Hypothesis incurs a big decrease in the error function then it will be accepted

Regularization

Keep all Features but reduce the parameters of some features

How

Add a regularization term to the cost function

$$J(\theta_0, \theta_1, .., \theta_m) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2 + \boldsymbol{\frac{\lambda}{2m} \sum_{j=1}^{n} \theta_j^2}$$

Regularization Term

What is the best value for λ

- Very small value of λ will cause over fitting in complicated Hypothesis
- Very large value of λ will cause under fitting

Regularization with Gradient Descent

Repeat until convergence:

{

$$\theta_0 = \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^i \right)$$

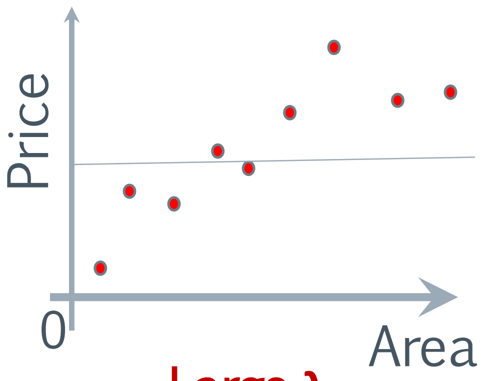
$$\theta_j = \theta_j - \frac{\alpha}{m} \left(\left(\sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^i \right) . x_j^{(i)} \right) + \lambda \theta_j \right) \}$$

$$\theta_j = \theta_j \left(1 - \frac{\alpha\lambda}{m} \right) - \frac{\alpha}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^i \right) . x_j^{(i)}$$

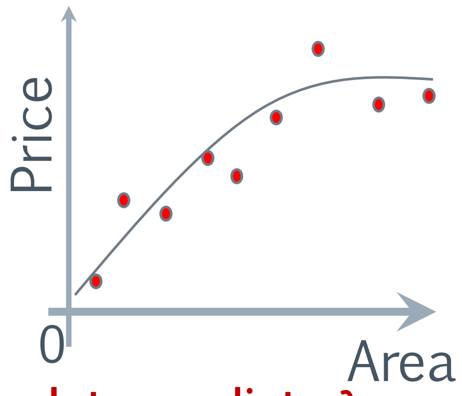
Regularization and Bias/Variance

For linear regression with regularization

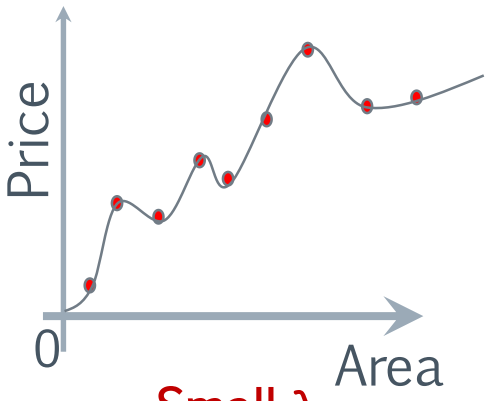
$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2 + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$$



Large λ
High Bias
(Underfit)



Intermediate λ
“Just Right”



Small λ
High Variance
(Overfit)

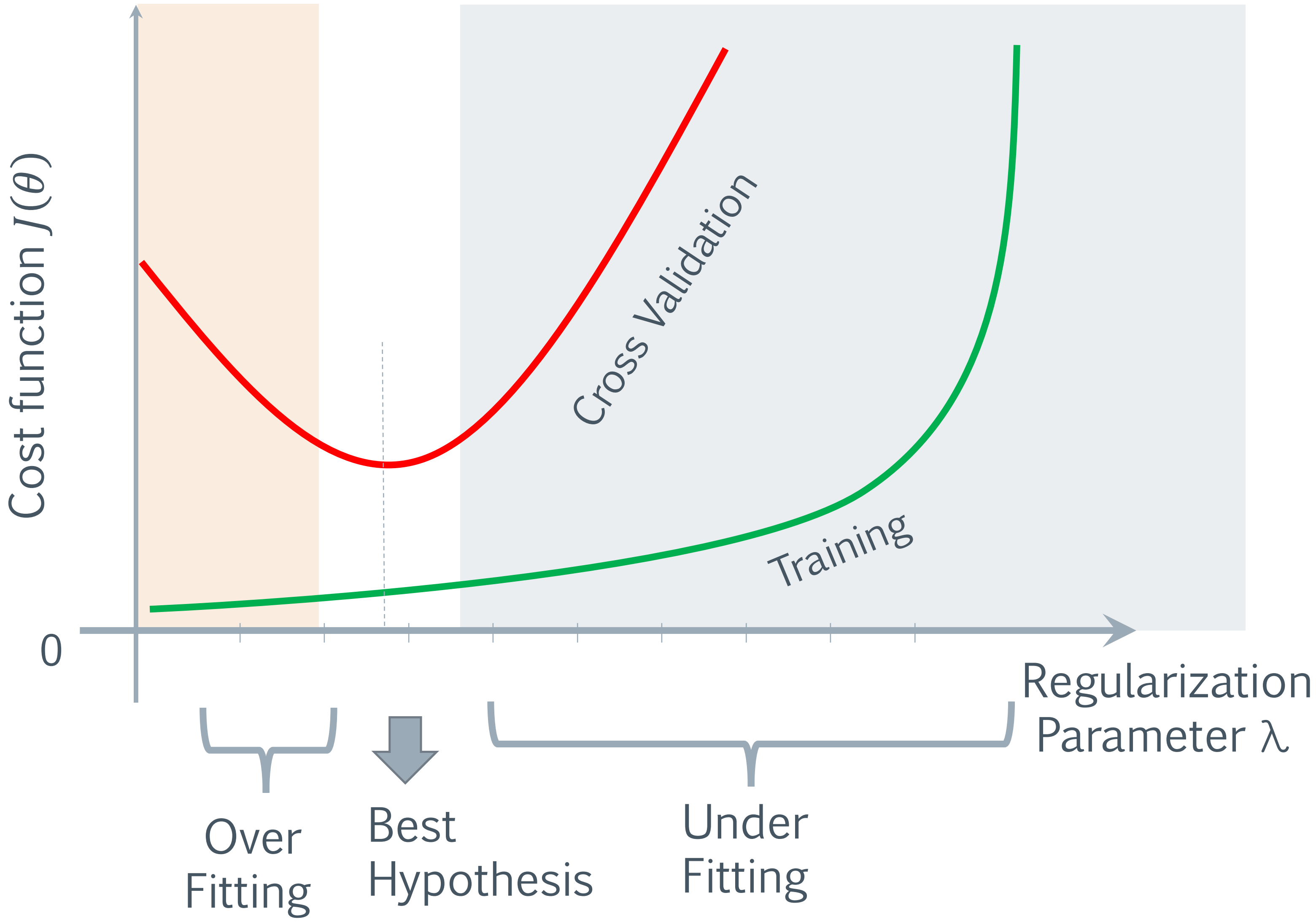
- For large λ, $\theta_1 \approx 0, \theta_2 \approx 0, \ldots$ so $h_\theta(x) = \theta_0$
- For small λ, regularization term is almost 0

Regularization and Bias/Variance

How to choose Regularization Factor **λ**

- Create a list of lambdas
  (i.e. λ∈{0,0.01,0.02,0.04,0.08,0.16,0.32,0.64,1.28,2.56,5.12,10.24});
- Create a set of models with different degrees or any other variants.
- Iterate through the **λs** and for each λ go through all the models to learn θ.
- Compute the cross validation error using the learned θ (computed with λ) on the $J_{cv}(\theta)$ **without** regularization or λ = 0: $J_{cv}(\theta) = \frac{1}{2m_{cv}} \sum_{i=1}^{m_{cv}} \left(h_\theta\left(x_{cv}^{(i)}\right) - y_{cv}^{(i)}\right)^2$
- Select the best combo that produces the lowest error on the cross validation set.
- Using the best combo θ and λ, apply it on $J_{test}(\theta)$ to see if it has a good generalization of the problem.
- For large λ, $\theta_1 \approx 0, \theta_2 \approx 0, \dots$ so $h_\theta(x) = \theta_0$
- For small λ, regularization term is almost 0

Regularization and Bias/Variance

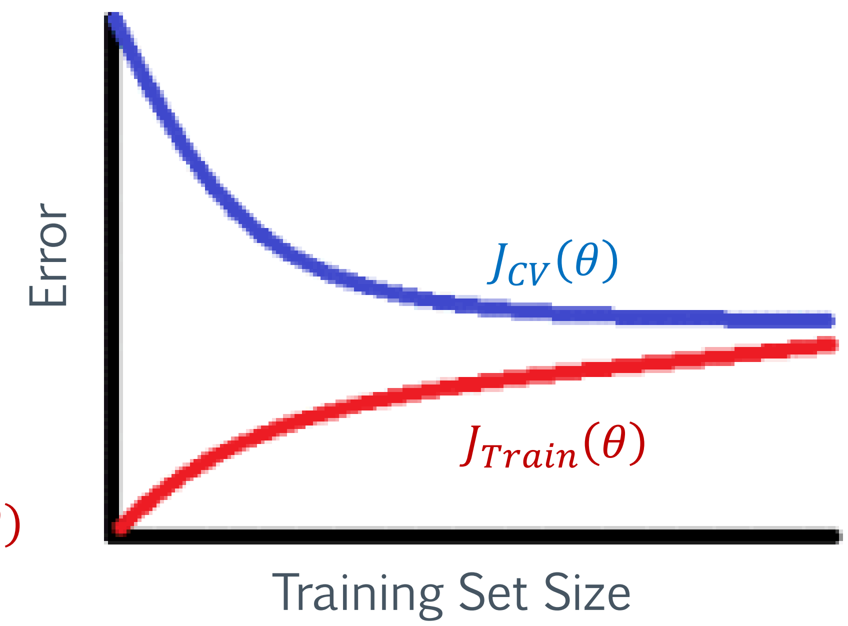


Learning Curves

- Representing the error in $J_{Train}(\theta)$ and $J_{CV}(\theta)$ while varying the training set size m

- When m is **small**:
  - ➢ Few training examples, easy to fit them all so $J_{Train}(\theta)$ will be low
  - ➢ Hypothesis doesn't generalize well to other data, so $J_{CV}(\theta)$ will be high

- When m is **large**:
  - ➢ Many training examples, harder to fit them all so $J_{Train}(\theta)$ will be high
  - ➢ Hypothesis generalize well to other data, so $J_{CV}(\theta)$ will be low

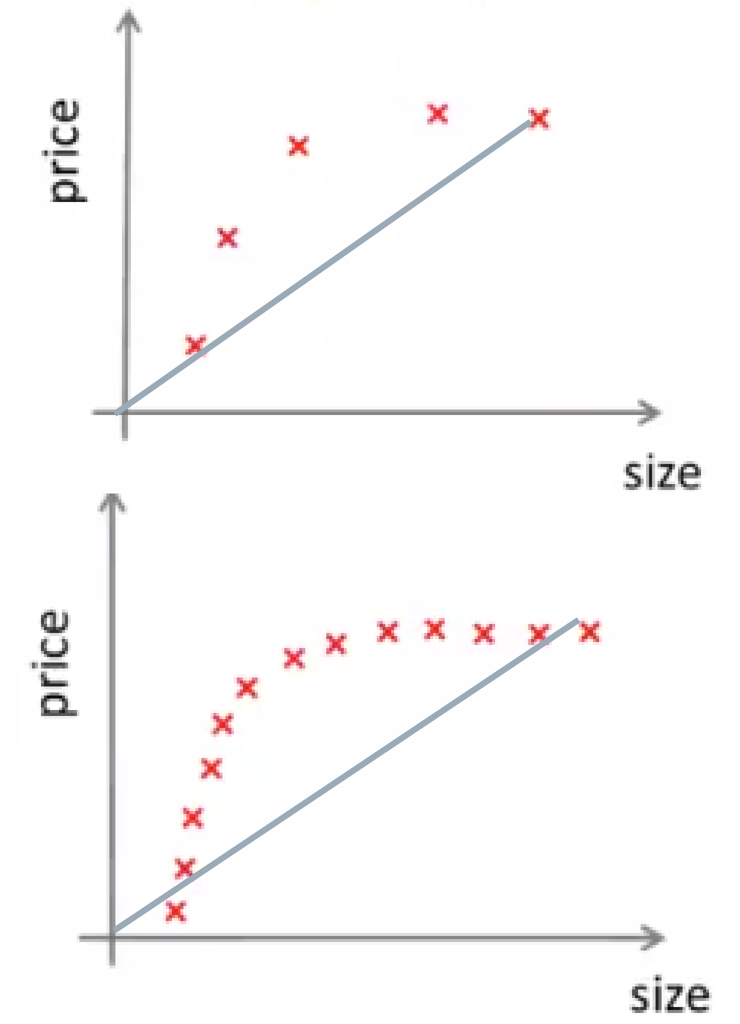


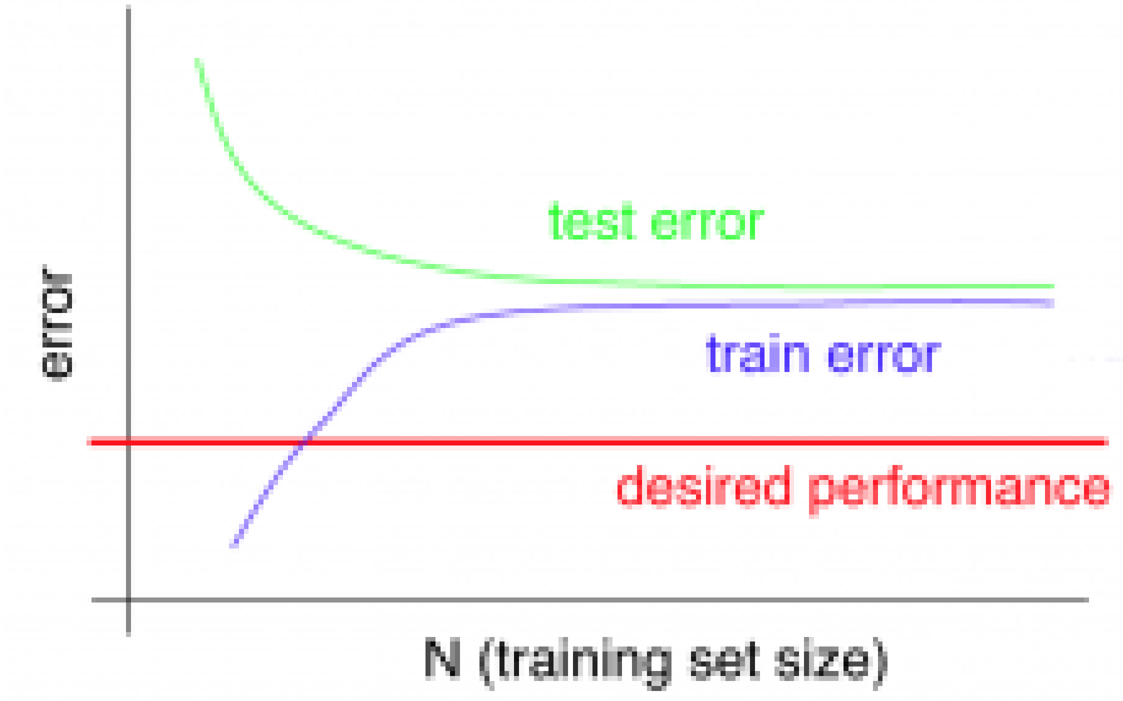
- Error in both functions will plateau after a certain m

Dr.- Ing. Maggie Mashaly

Learning Curves – High Bias

Example: $h_\theta(x) = \theta_0 + \theta_1 x$

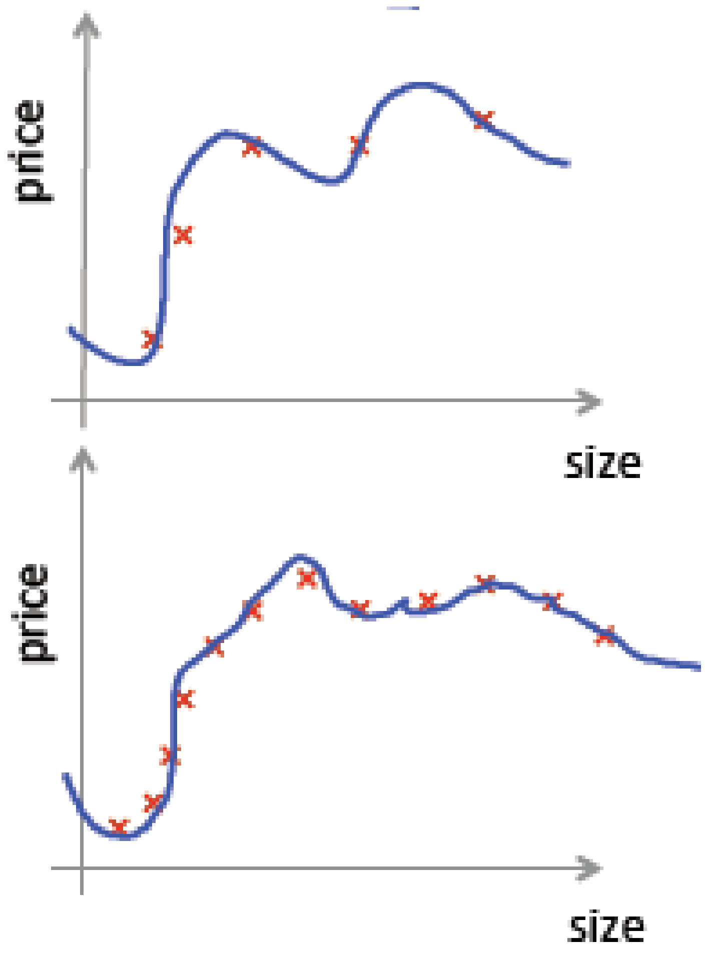




- Results in high error values for both $J_{Train}(\theta)$ and $J_{CV}(\theta)$
- If a learning algorithm is suffering from high bias, getting more training data will not (by itself) help much.

Learning Curves – High Variance

Example: $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{100} x^{100}$



- Results in high error values for both $J_{Train}(\theta)$ and $J_{CV}(\theta)$
- If a learning algorithm is suffering from high variance, getting more training data is likely to help

Machine Learning Fundamentals – DTSC102

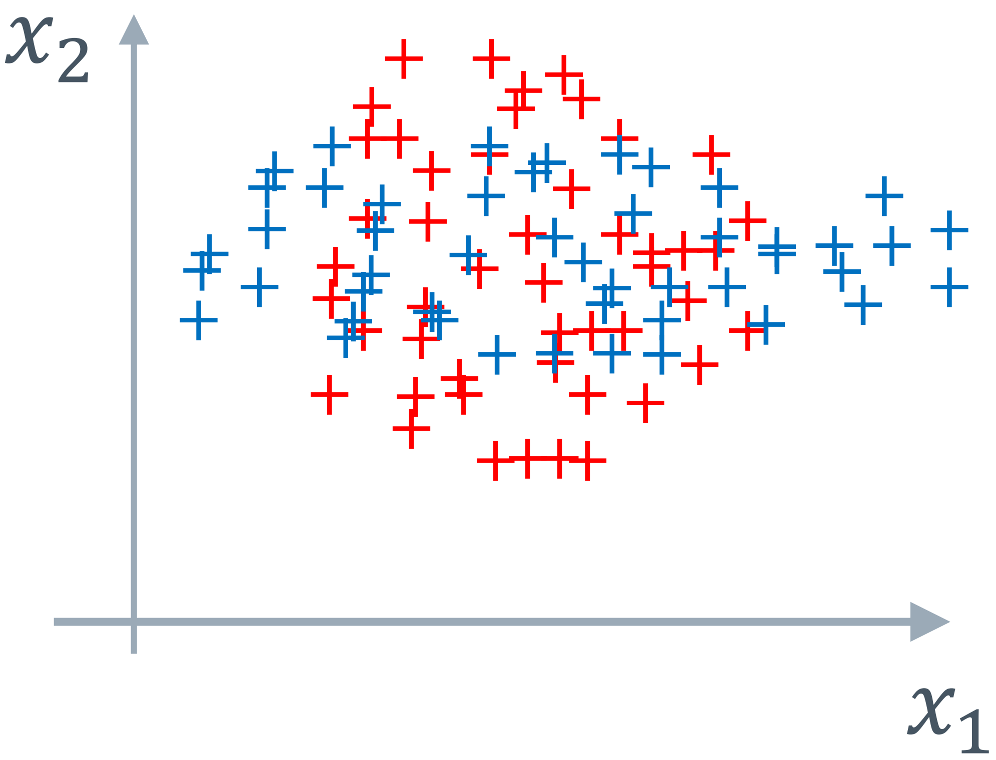
Lecture 7
Principle Component Analysis - PCA

Course Instructor: Dr.-Ing. Maggie Mashaly
maggie.ezzat@guc.edu.eg
C3.220

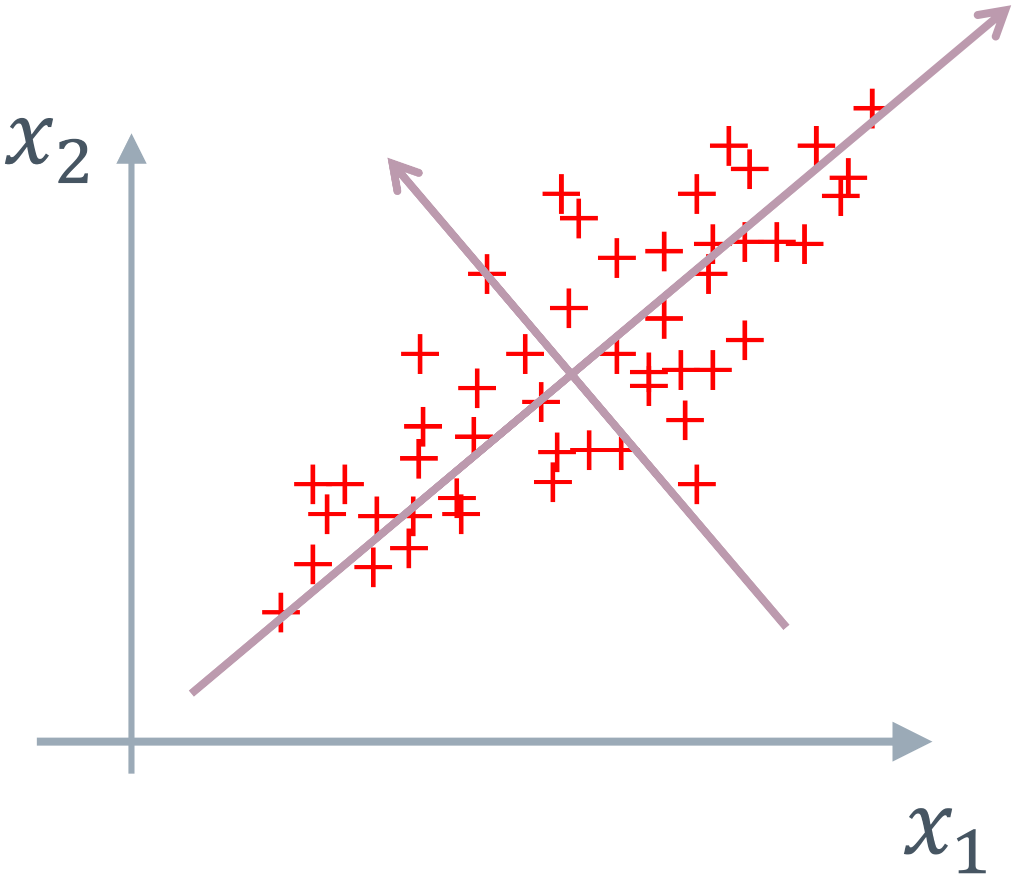
Contents

Dr.- Ing. Maggie Mashaly

Correlation



Uncorrelated



Correlated

Main Questions

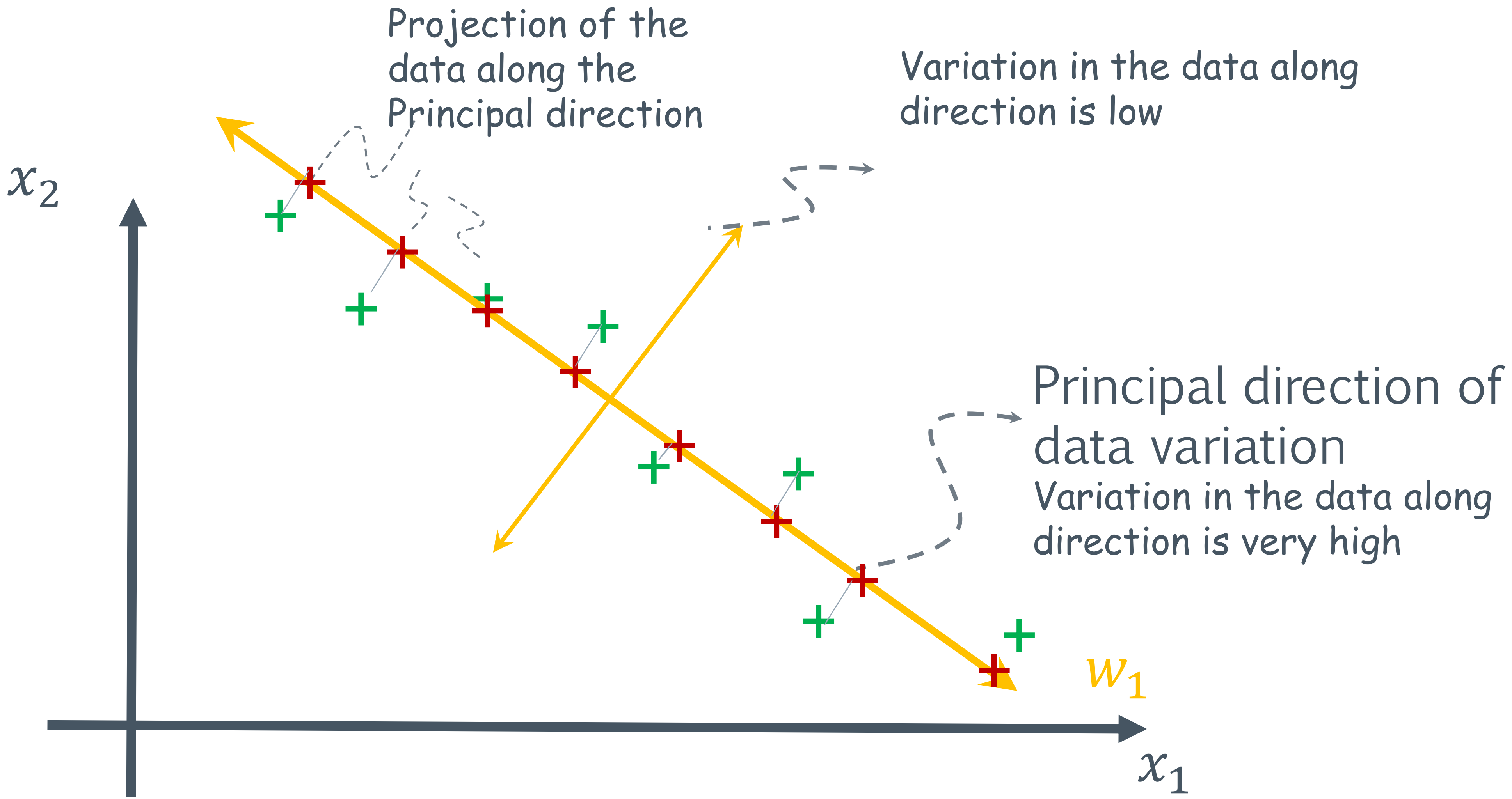
- How to remove correlation?
- What are the dimensions with the most information?
- How can we provide a better scaling or normalization?

Principal Component Analysis

To identify the principal directions in which the data varies

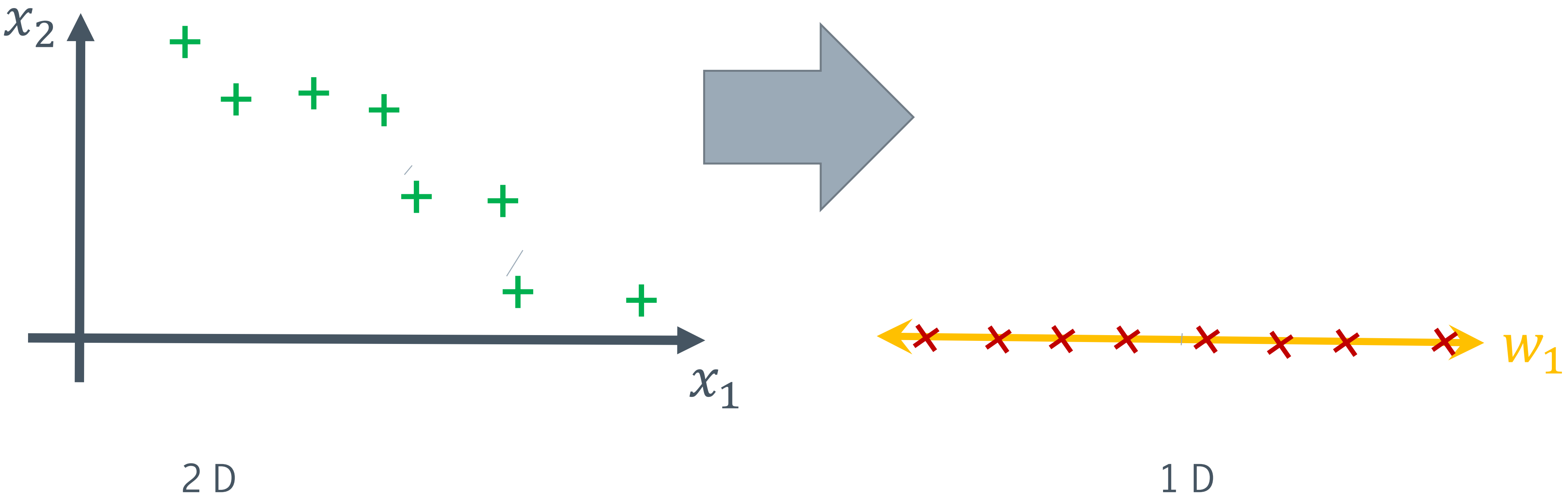
- ❑Better data representation
  - ➢Reduce the number of features
  - ➢Reduce the relation between features
- ❑Eliminate non principal data
- ❑Can be used for compression applications

Principal Component Analysis
2D Example



Dimensionality Reduction

We are interested to find a linear transformation from an m dimension to n dimension (reduce the number of features from m to n



Linear Transformation

$$x = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & x_3^{(1)} & \dots x_m^{(1)} \\ x_1^{(2)} & x_2^{(2)} & x_3^{(2)} & \dots x_m^{(2)} \\ x_1^{(3)} & x_2^{(3)} & x_3^{(3)} & \dots x_m^{(3)} \\ & \dots \dots \dots \dots \dots . & & \\ x_1^{(n)} & x_2^{(n)} & x_3^{(n)} & \dots x_m^{(n)} \end{bmatrix}$$

Data Matrix

$$W = \begin{bmatrix} w_{1,1} & w_{1,2} & w_{1,3} \dots w_{1,k} \\ w_{2,1} & w_{2,2} & w_{2,3} \dots w_{2,k} \\ w_{3,1} & w_{3,2} & w_{3,3} \dots w_{3,k} \\ & \dots \dots \dots \dots \dots . & \\ w_{m,1} & w_{m,2} & w_{m,2} \dots w_{m,k} \end{bmatrix}$$

Transformation matrix

Transform into nxk

nxk nxm mxk

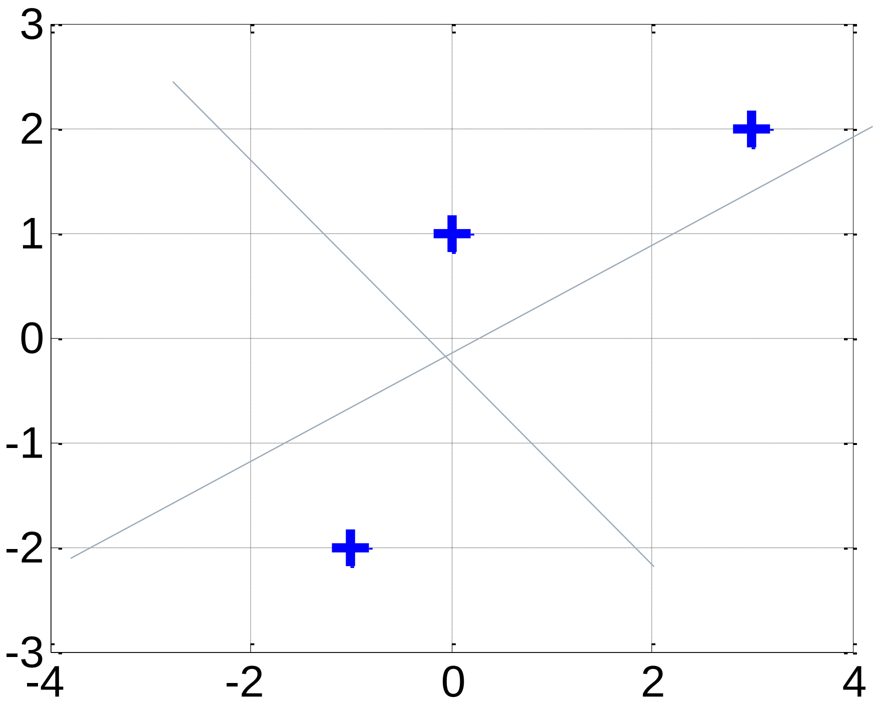
$$Z = xW$$

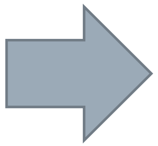
K features
N data points

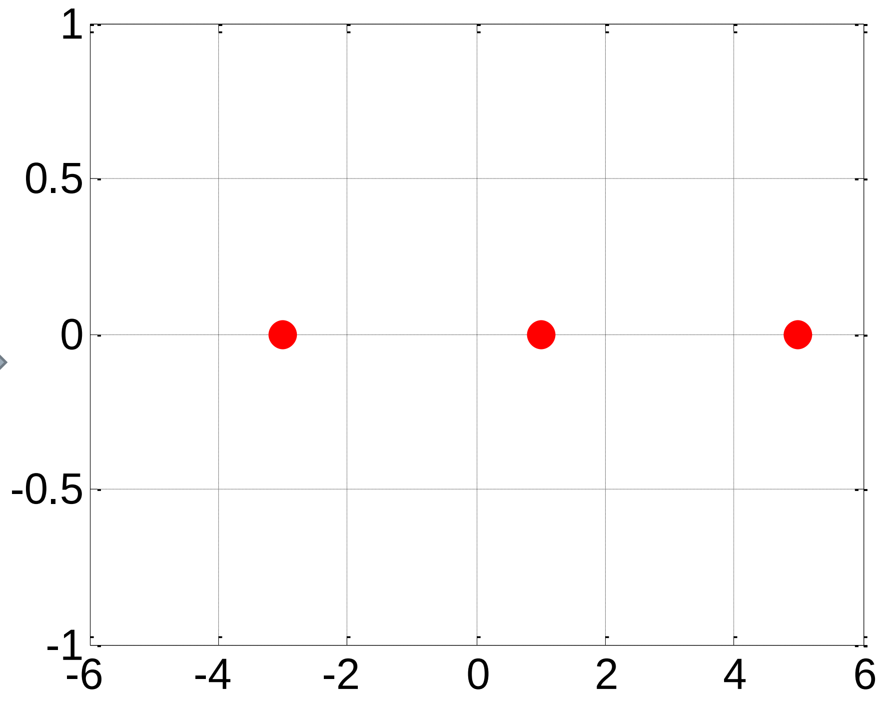
M features
N data points

Example

$$x = \begin{bmatrix} 0 & 1 \\ 3 & 2 \\ -1 & -2 \end{bmatrix} \qquad w = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad Z = \begin{bmatrix} 1 \\ 5 \\ -3 \end{bmatrix}$$







Dr.- Ing. Maggie Mashaly

Finding the right Transformation

- Let us first forget about reducing the dimensions and focus on reducing the relation between features
- Relation between features is described by the covariance matrix
- $Cov(x, x) = x^T x$
- The best transformation would give no relation between any two features
  - (this means that the covariance matrix an identity matrix)
  - $Cov(z, z) = z^T z = I$

Finding the right Transformation

Correlated M features

nxm

x

$Z = xW$

nxk

z

$z^T z = I$

Uncorrelated K features

[kxm] [mxn] [nxm] [mxk]

$$z^T z = w^T x^T x w = I$$

How to Choose K?

- Iteratively increase k from 1 to n and calculate the error
- Pick the k that ensures an error that is smaller that predefined value
- Luckily the eigen values returned for the case of the covariance matrix provides the contribution of each feature and hence it provides the same value as the calculating the error