

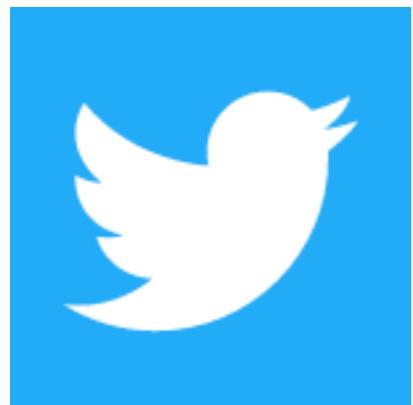
# Introduction to Machine Learning

Jiří Materna





@mlcollegecom



@mlcollegecom



#mlcollege

# About me

- Ph.D. in Natural Language Processing and Artificial Intelligence at Masaryk University
- 10 years at [seznam.cz](http://seznam.cz) (last 8 years as Head Of Research)
- Founder and co-organizer of ML Prague
- Author of the ML Guru blog
- Mentor at StartupYard and Startup AI Incubator
- ML Freelancer and consultant

# Outline

## Day 1

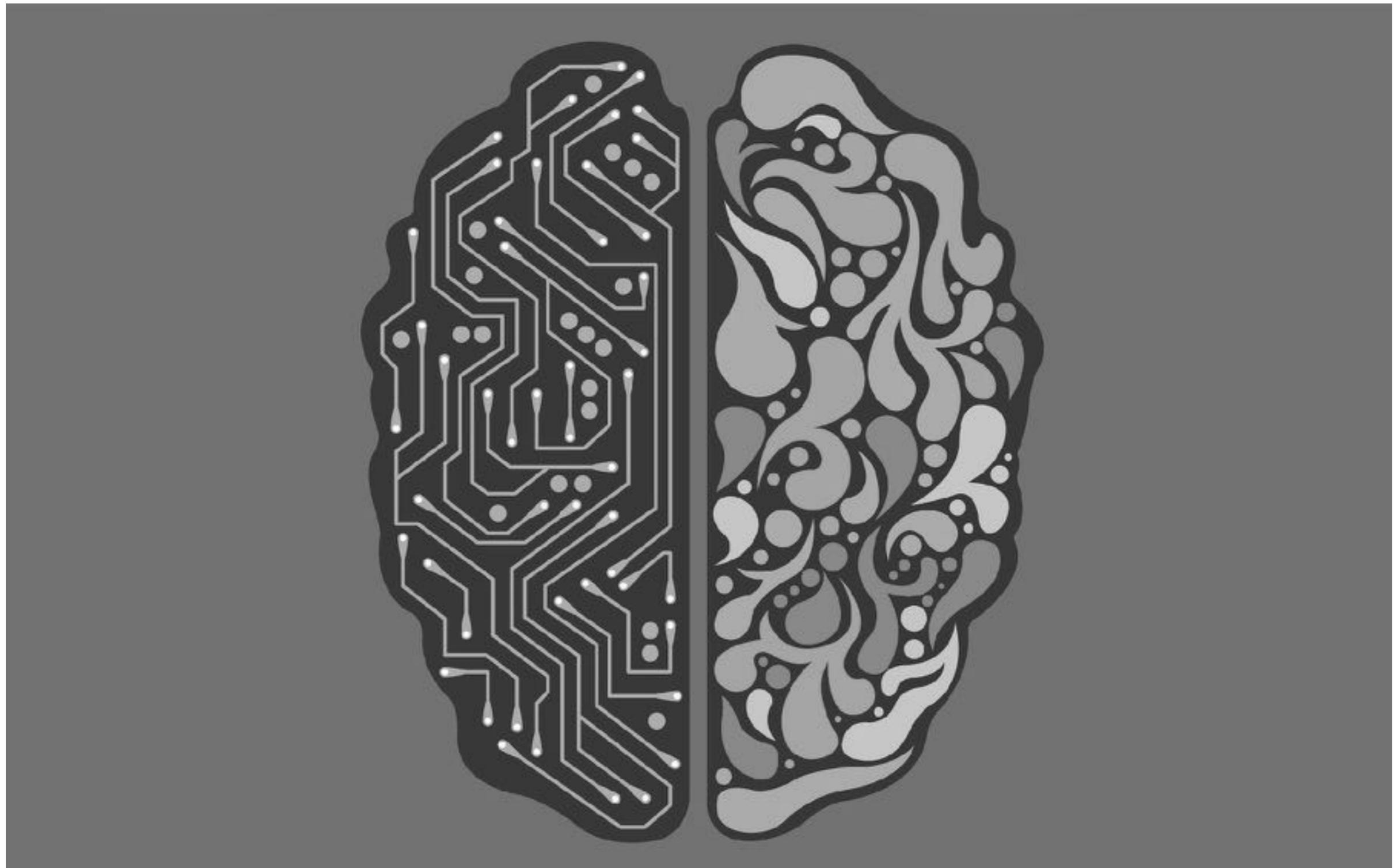
- Introduction to machine learning
- Types of ML tasks
- Data preparation
- Model evaluation
- Basic classification algorithms
- Scikit-learn tutorial
- Practical classification task
- Basic regression algorithms
- Regression model evaluation
- Practical regression task

# Outline

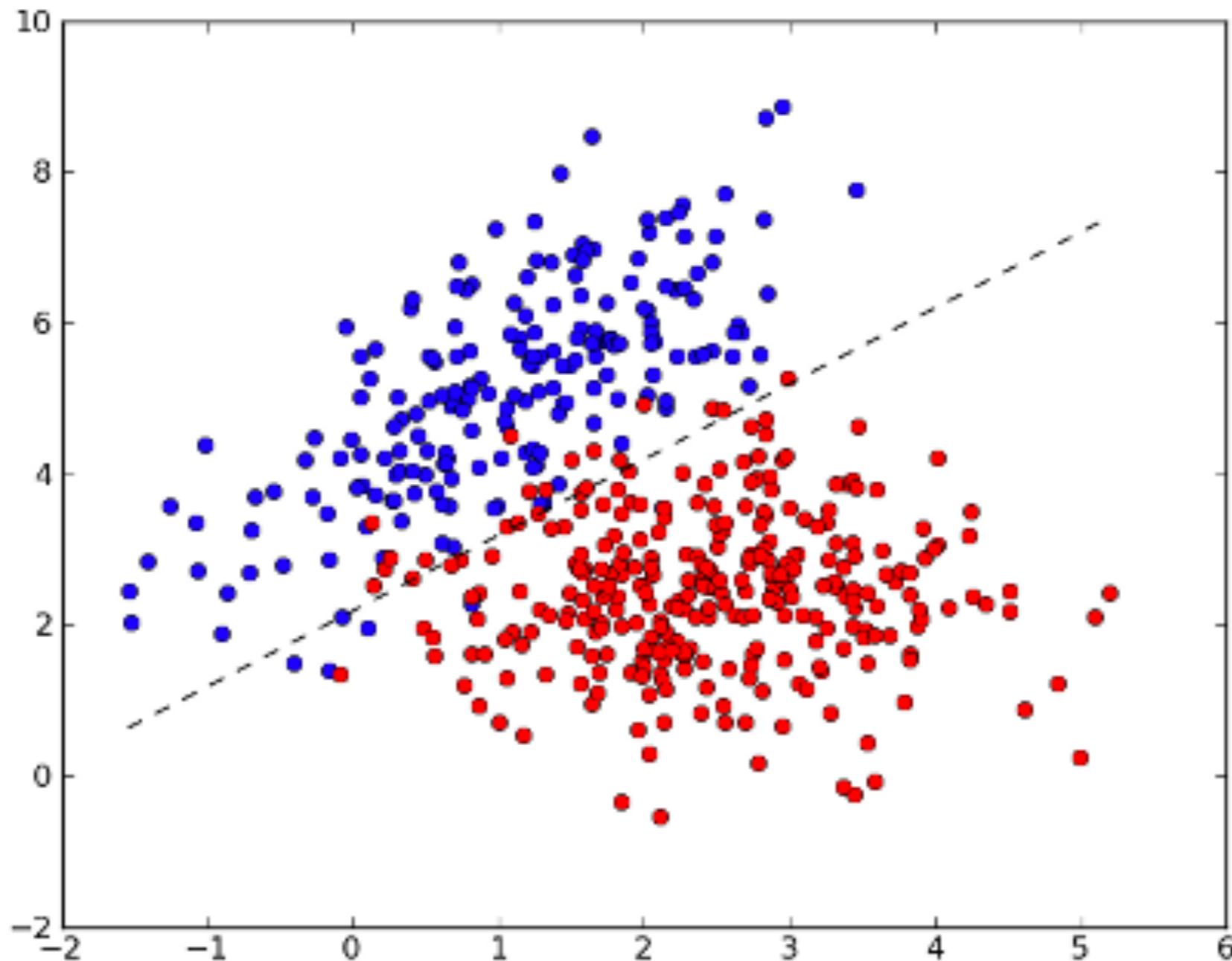
## Day 2

- Dimensionality reduction
- Basic clustering algorithms
- Practical clustering task
- Introduction to neural networks
- Activation functions for neural networks
- Multilayered neural networks
- Methods for training neural networks
- Keras tutorial
- Practical classification and regression tasks solved using neural networks

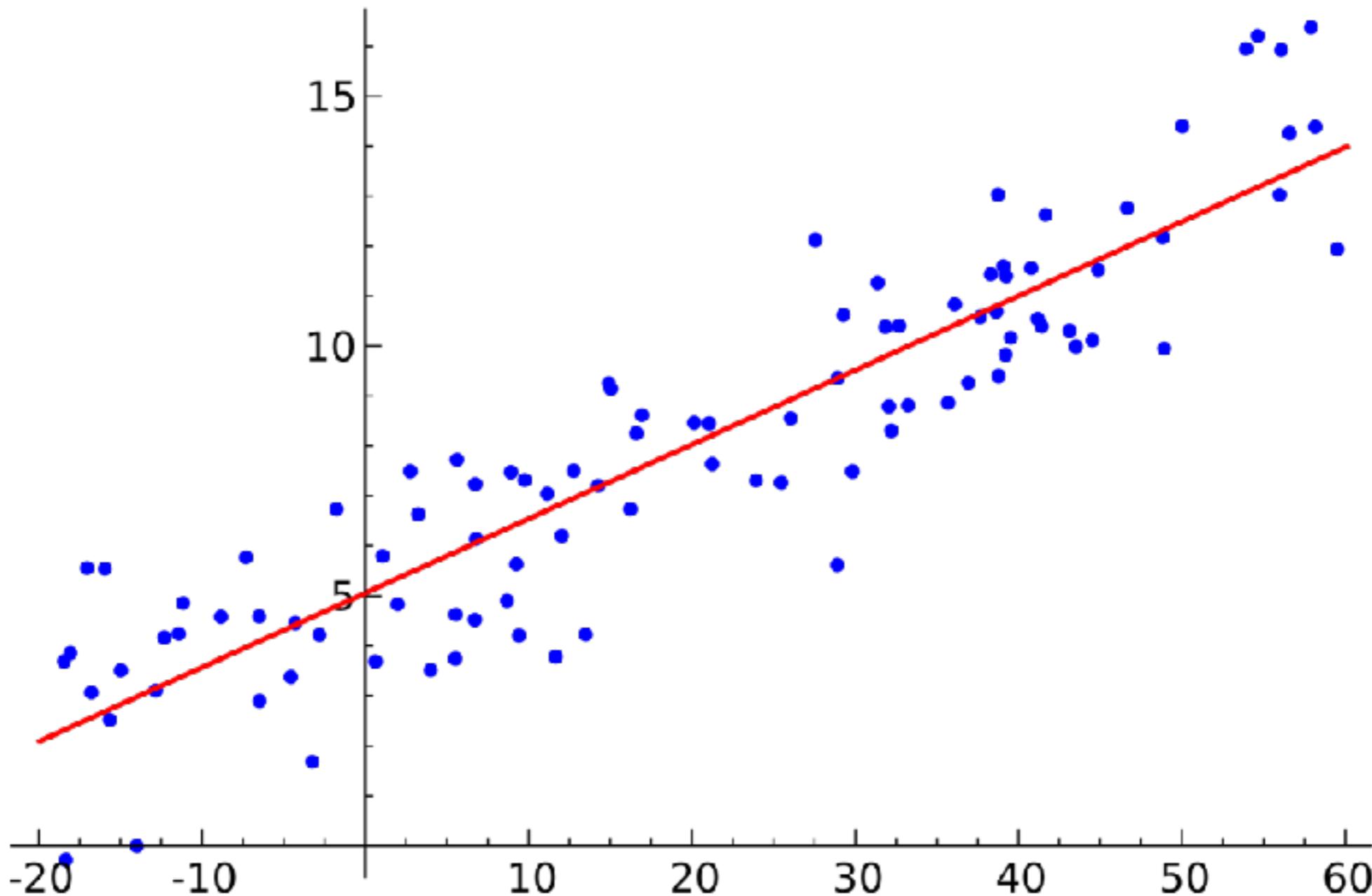
# What is (not) machine learning?



# Classification

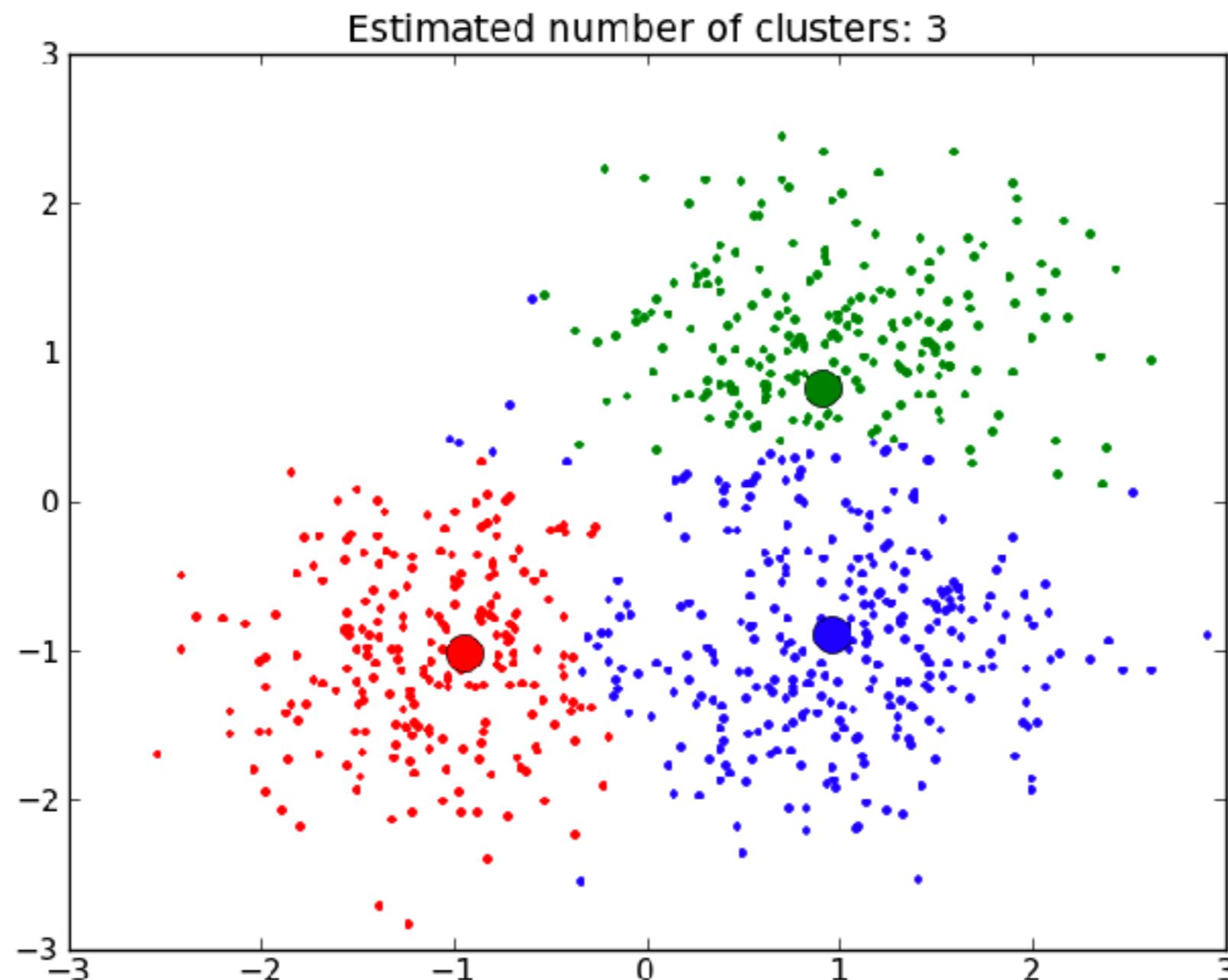


# Regression

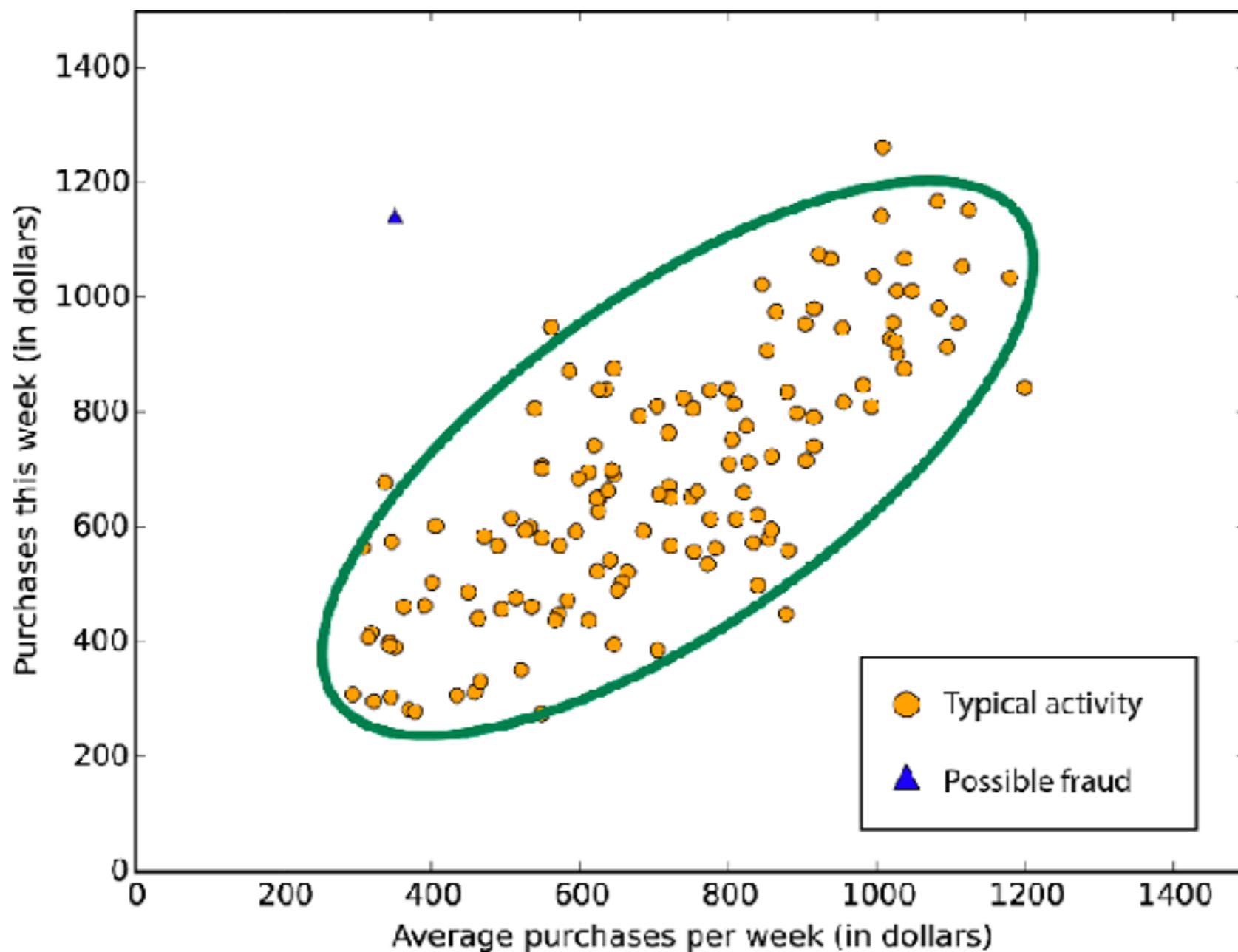


Source: [www.wikipedia.org](http://www.wikipedia.org)

# Clustering

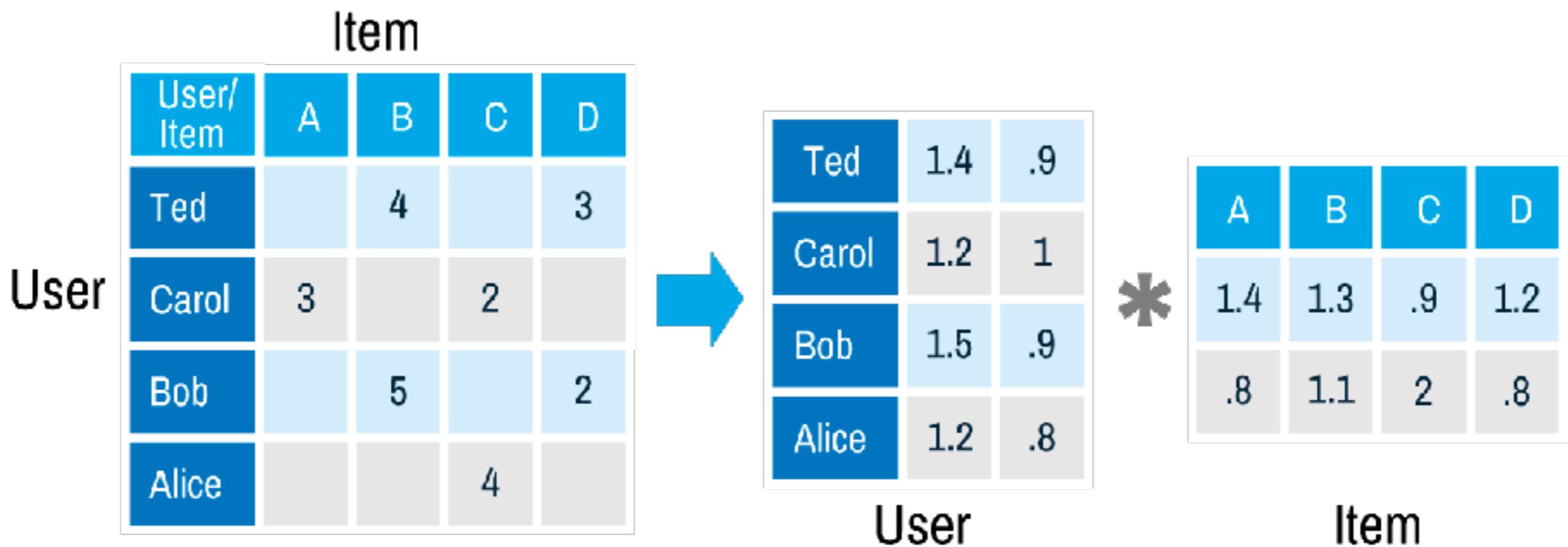


# Anomaly detection

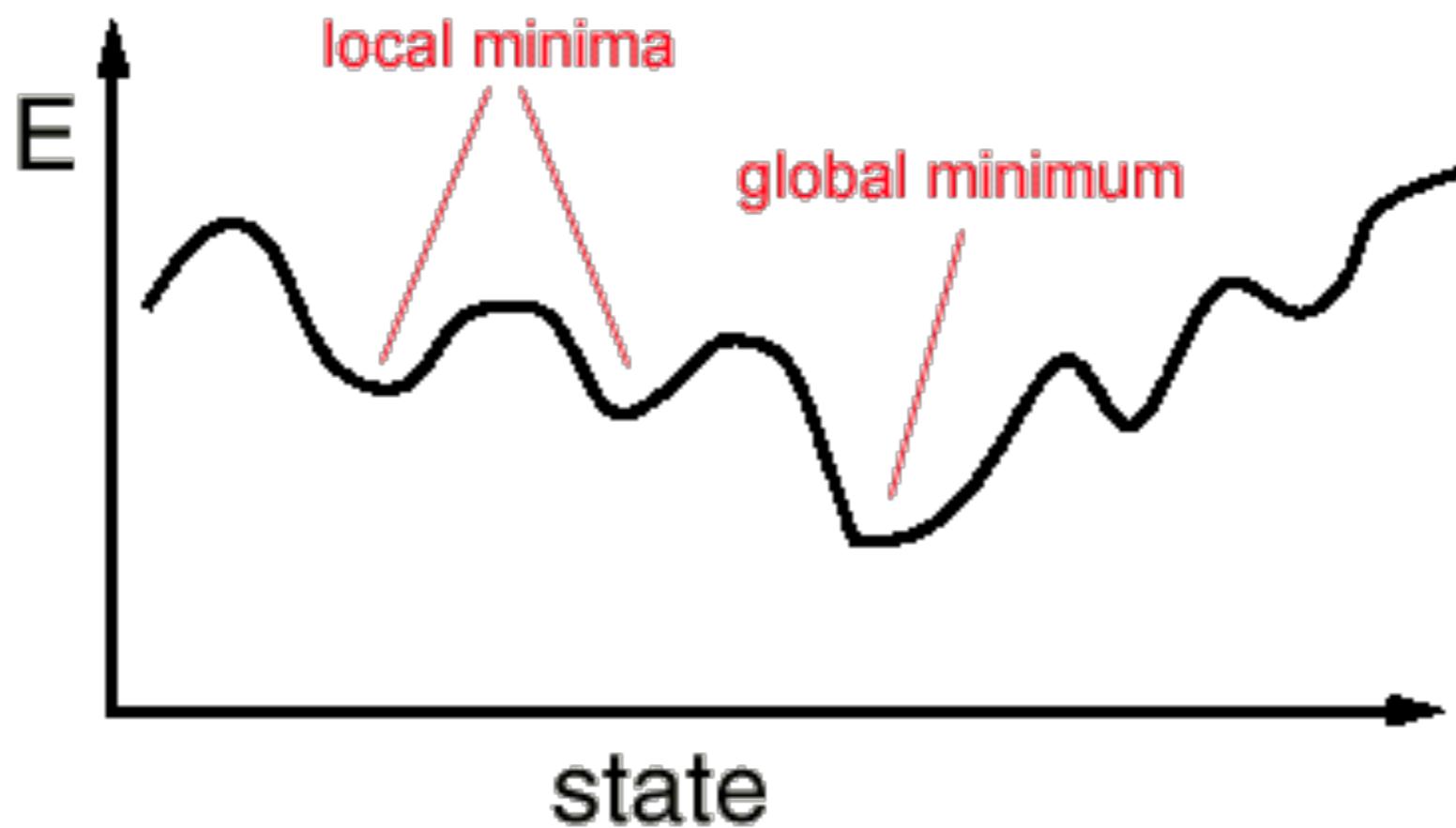


Source: <https://docs.microsoft.com/en-us/azure>

# Recommendation



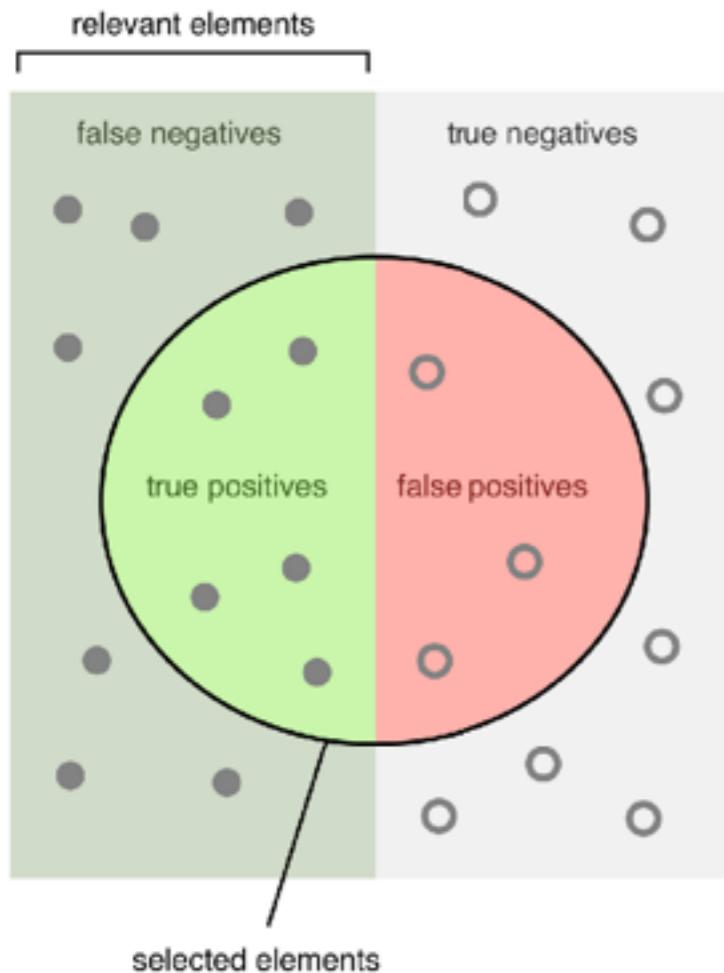
# Optimization



# Data

- Train, validation, test data sets
- Cross-validation
- Imbalanced data sets
- Baseline models

# Model evaluation



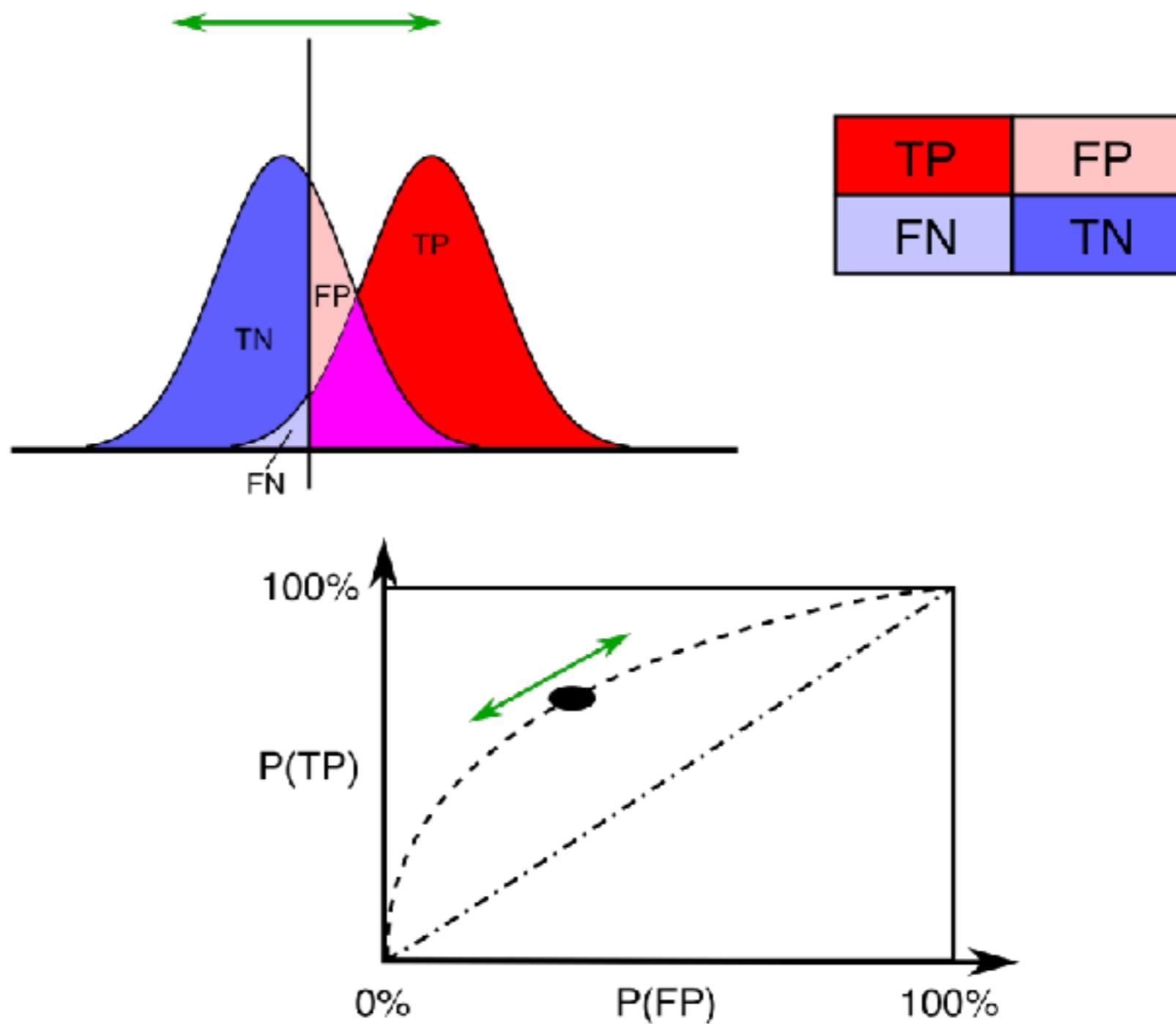
$$\text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn}$$

$$\text{Precision} = \frac{tp}{tp + fp}$$

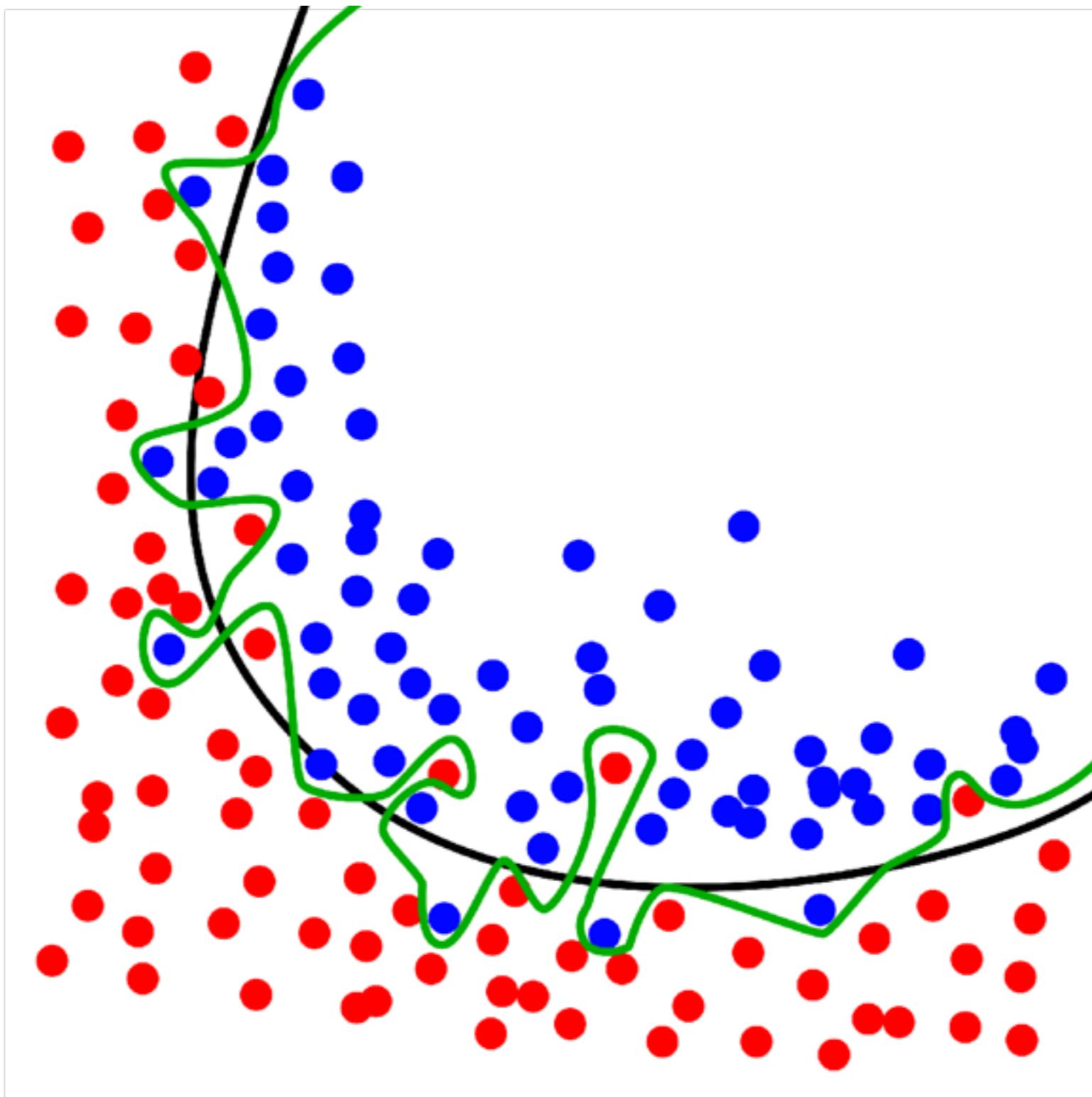
$$\text{Recall} = \frac{tp}{tp + fn}$$

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

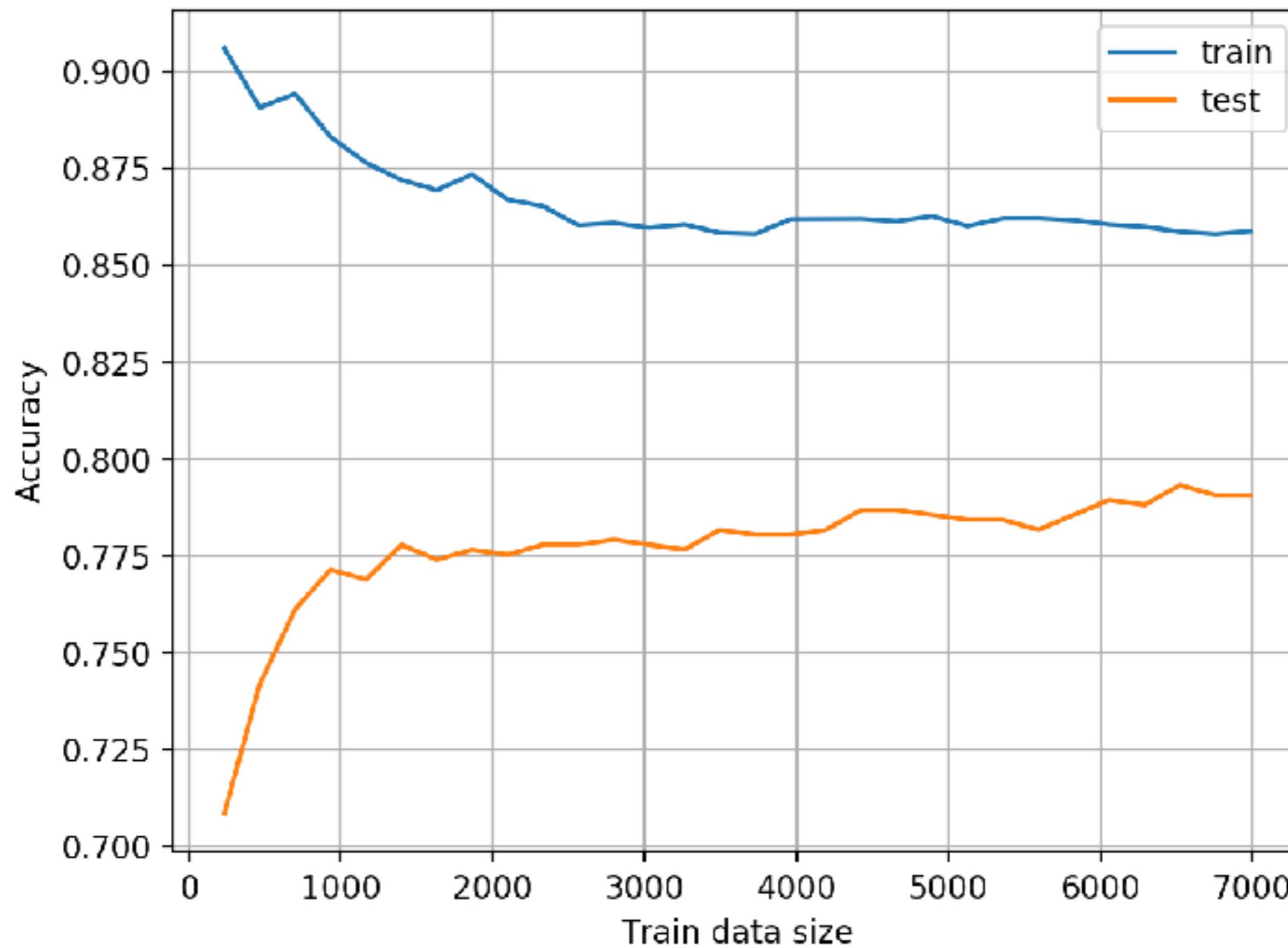
# Receiver operating characteristic (ROC Curve), AUC



# Overfitting



# Overfitting detection



# Conditional probability

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A)P(B|A)}{P(B)}$$

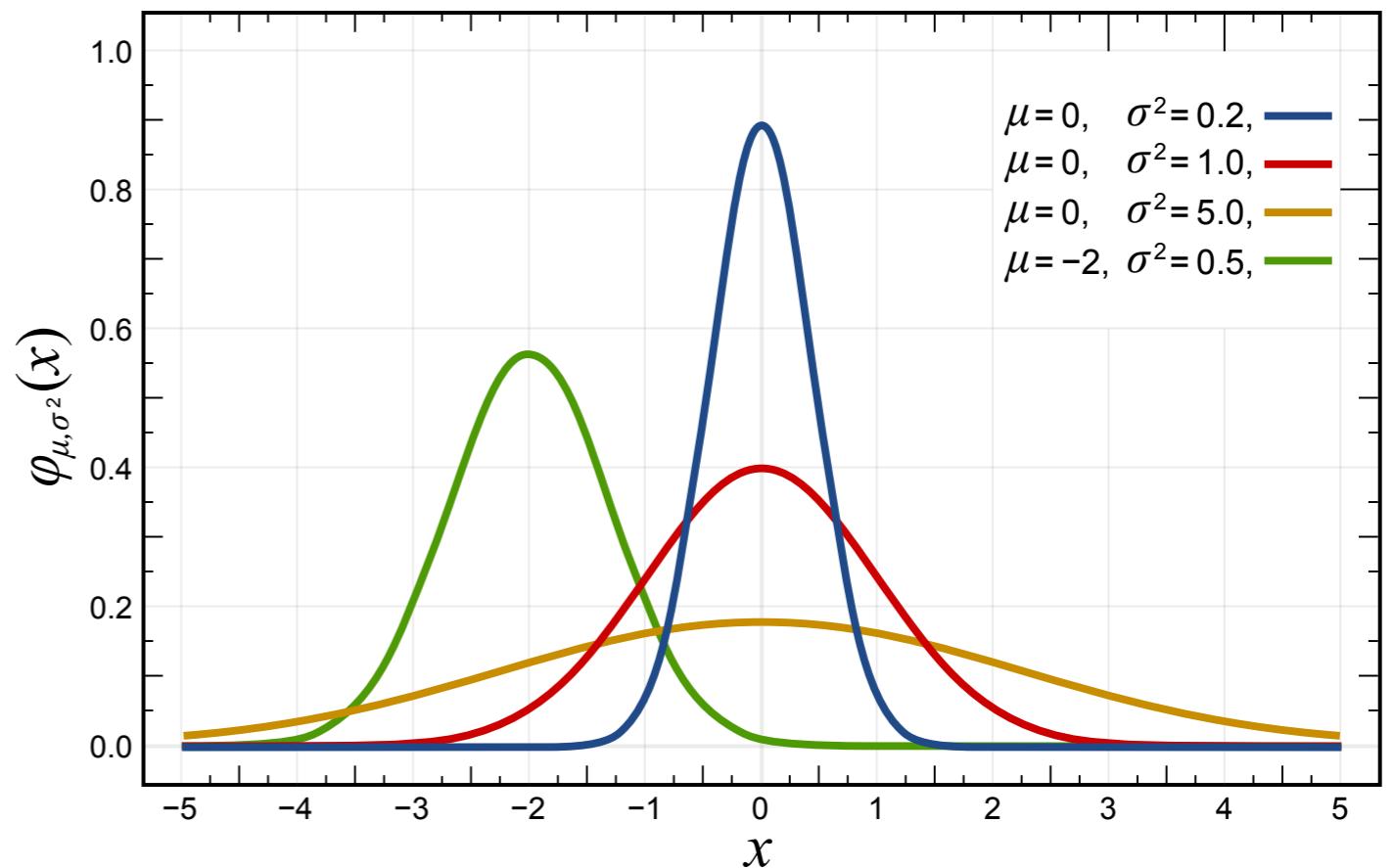
$$A \perp B \iff P(A \cap B) = P(A)P(B)$$

# Naive Bayes Classifier

client	balance	income	sex	unemployed	loan
1	H	H	F	F	T
2	L	H	M	F	T
3	L	L	M	T	F
4	H	L	F	T	T
5	L	L	F	T	F
6	H	L	M	F	?

# Gaussian Naive Bayes Classifier

$$p(x = v|C_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{(v - \mu_k)^2}{2\sigma_k^2}}$$



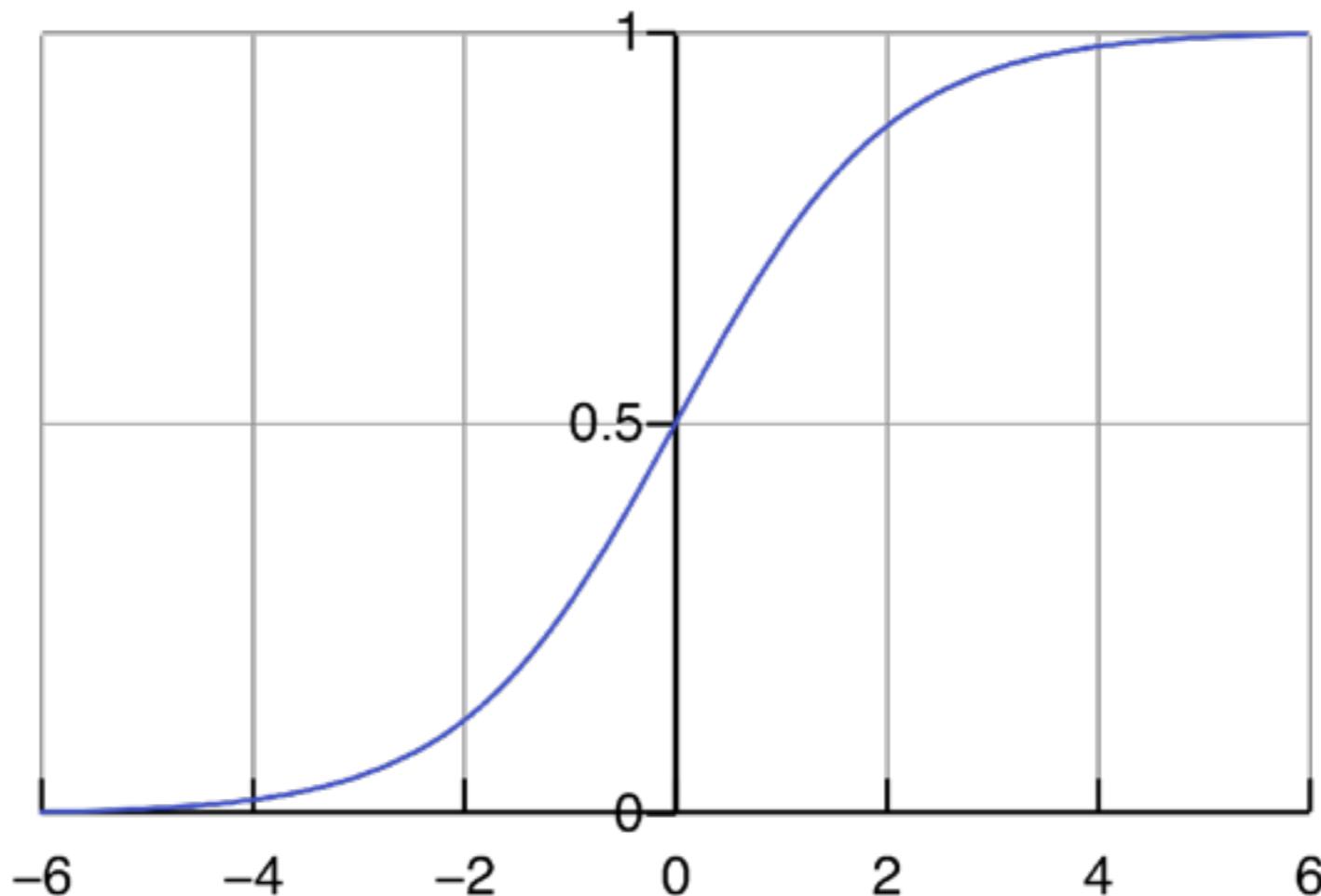
# Scikit-learn tutorial

<http://scikit-learn.org/stable/>

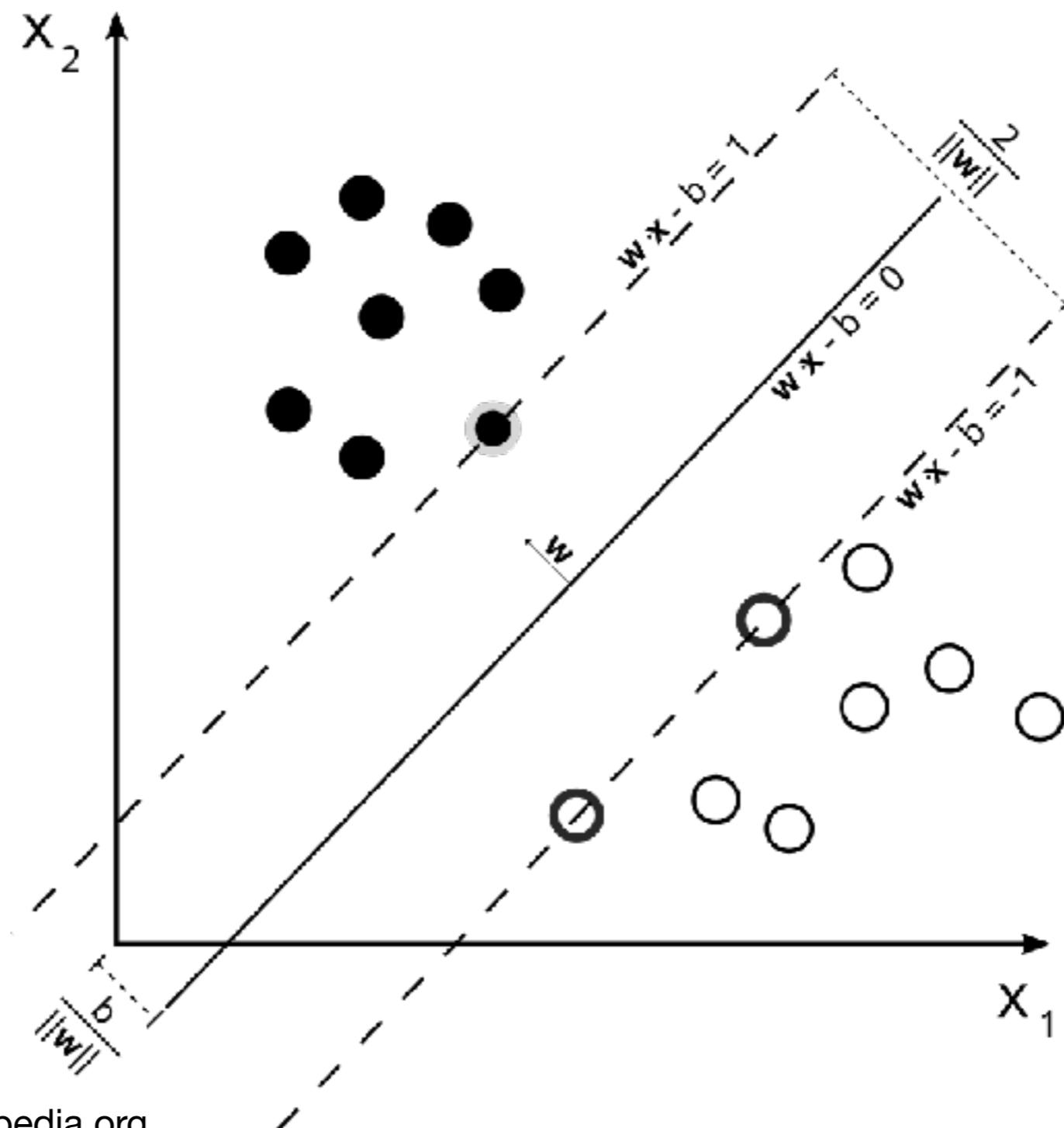
[01-Scikit-introduction.ipynb](#)

# Logistic regression

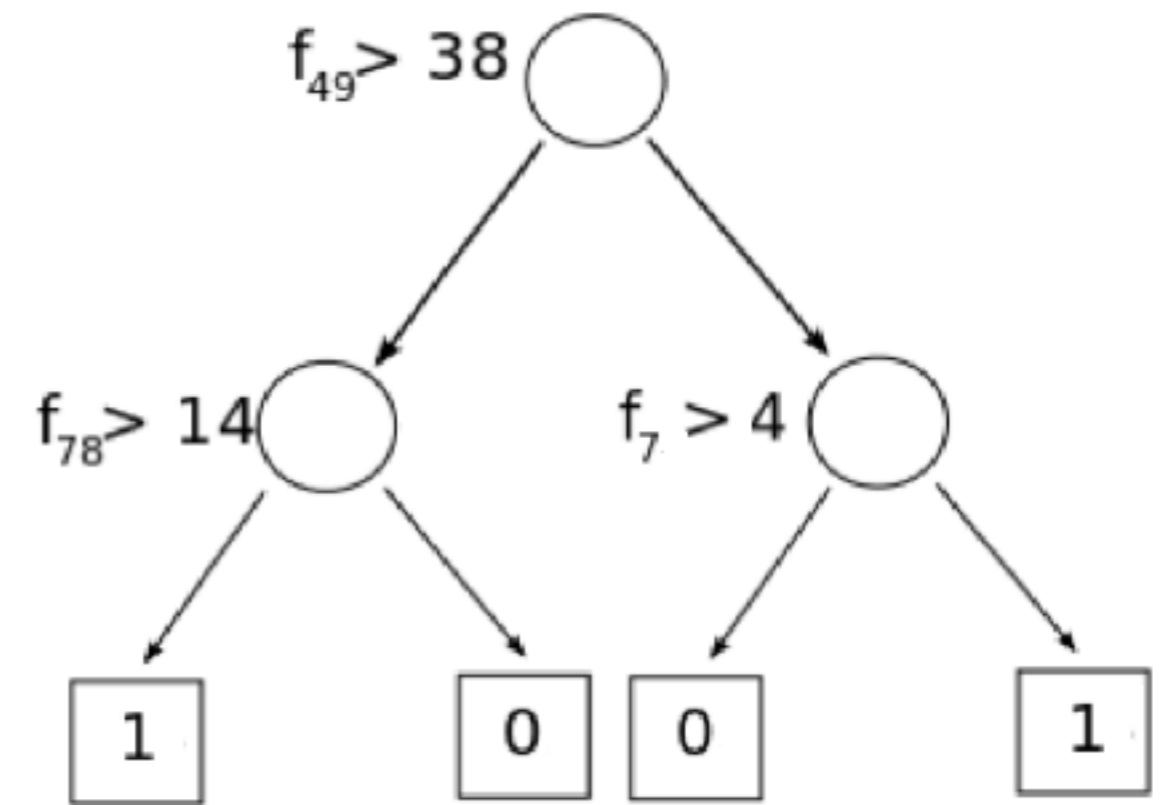
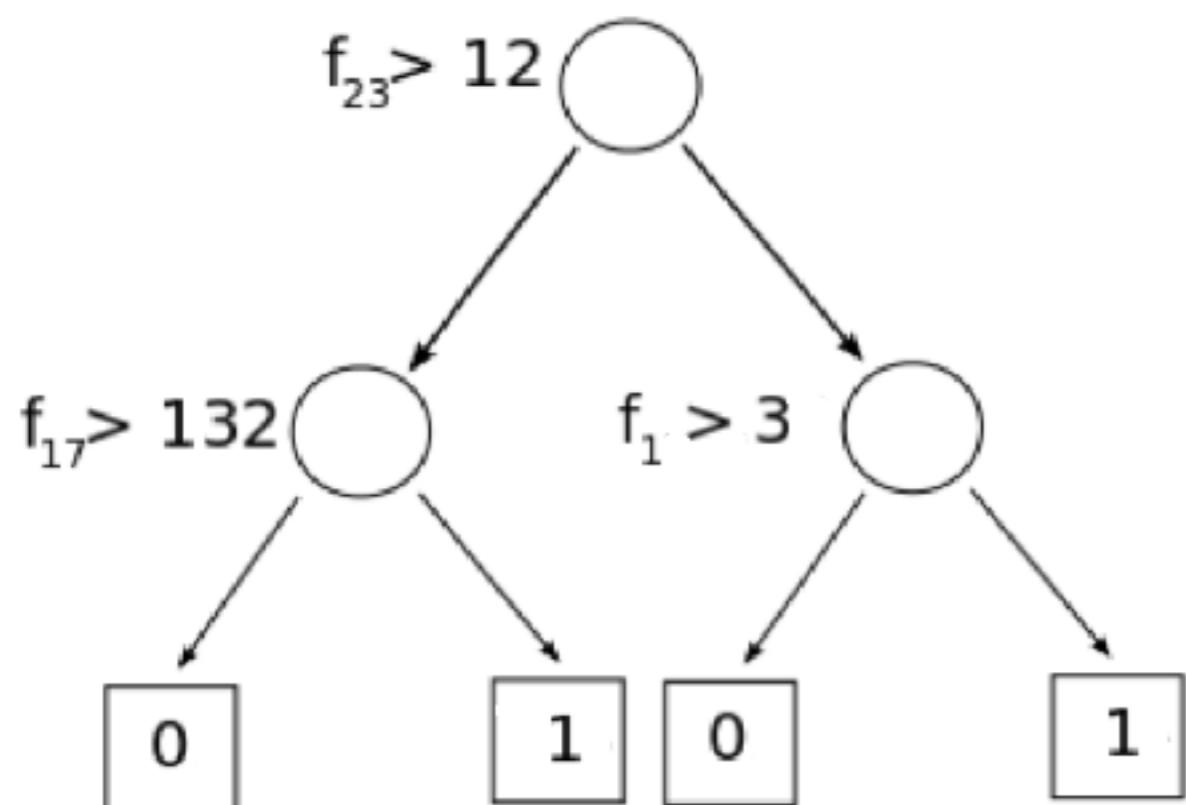
$$P(y|\vec{x}) = \frac{1}{1 + e^{-(\vec{x}\vec{w}+w_0)}}$$



# Support Vector Machines



# Boosted Decision Trees

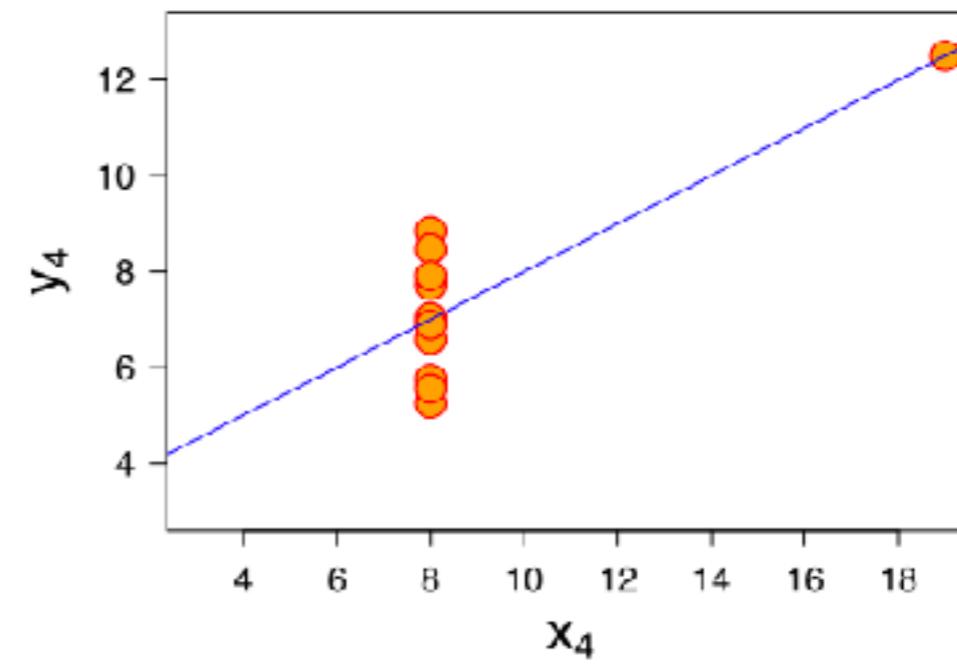
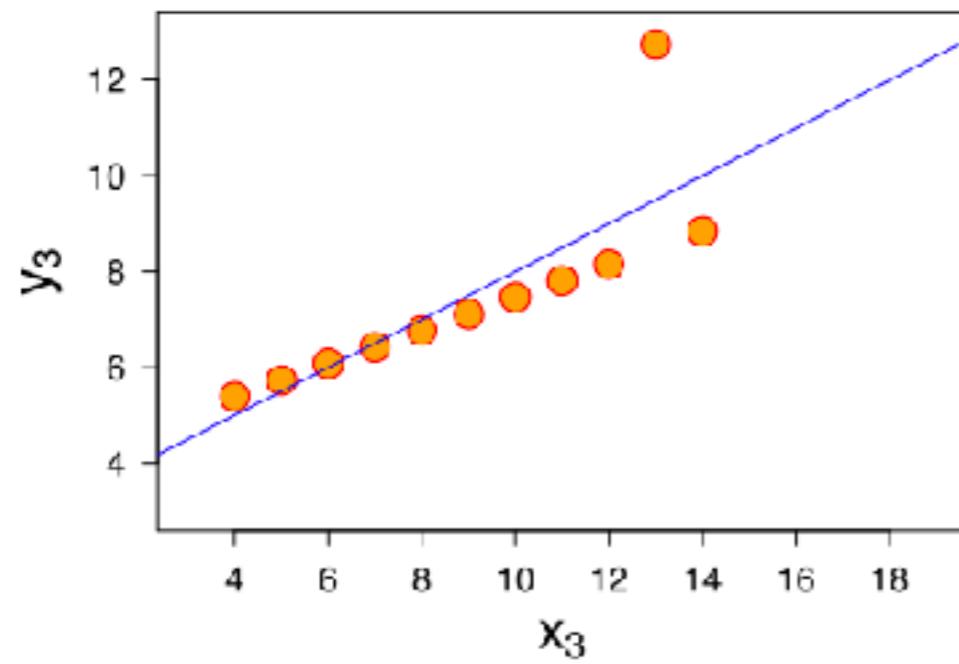
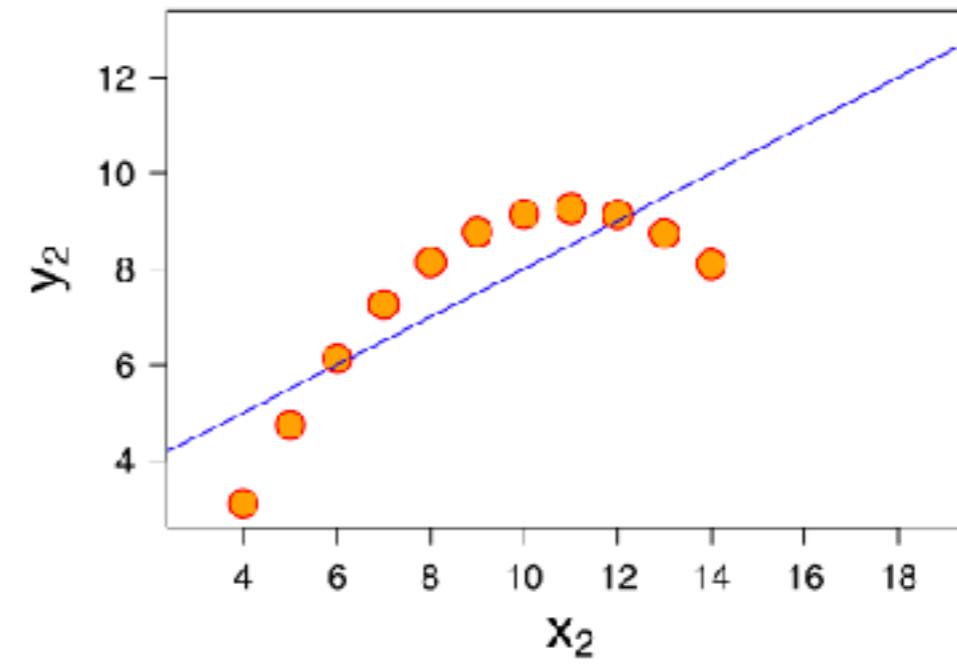
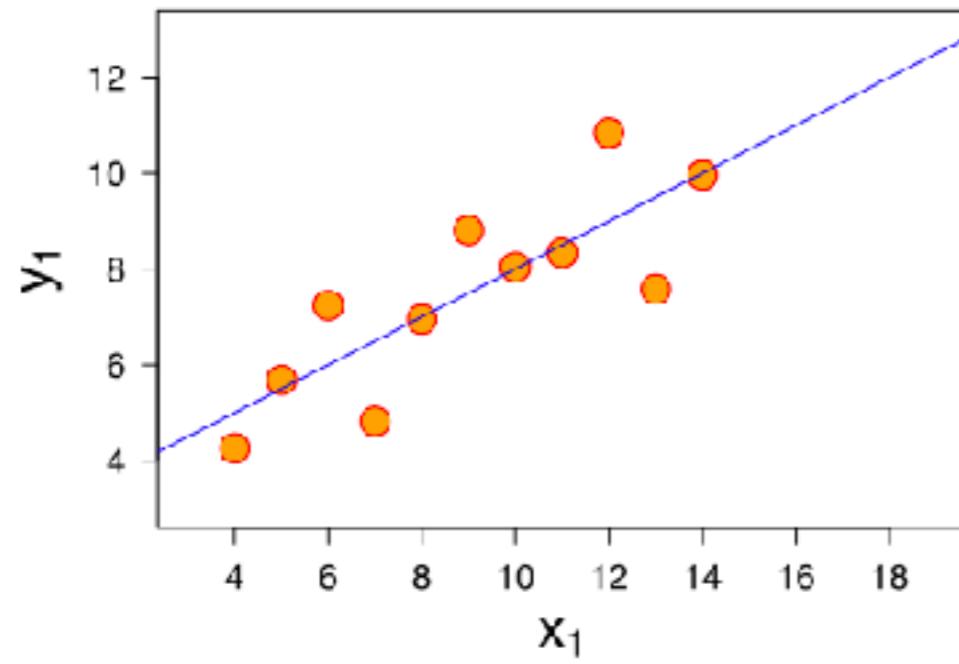


# Classification task

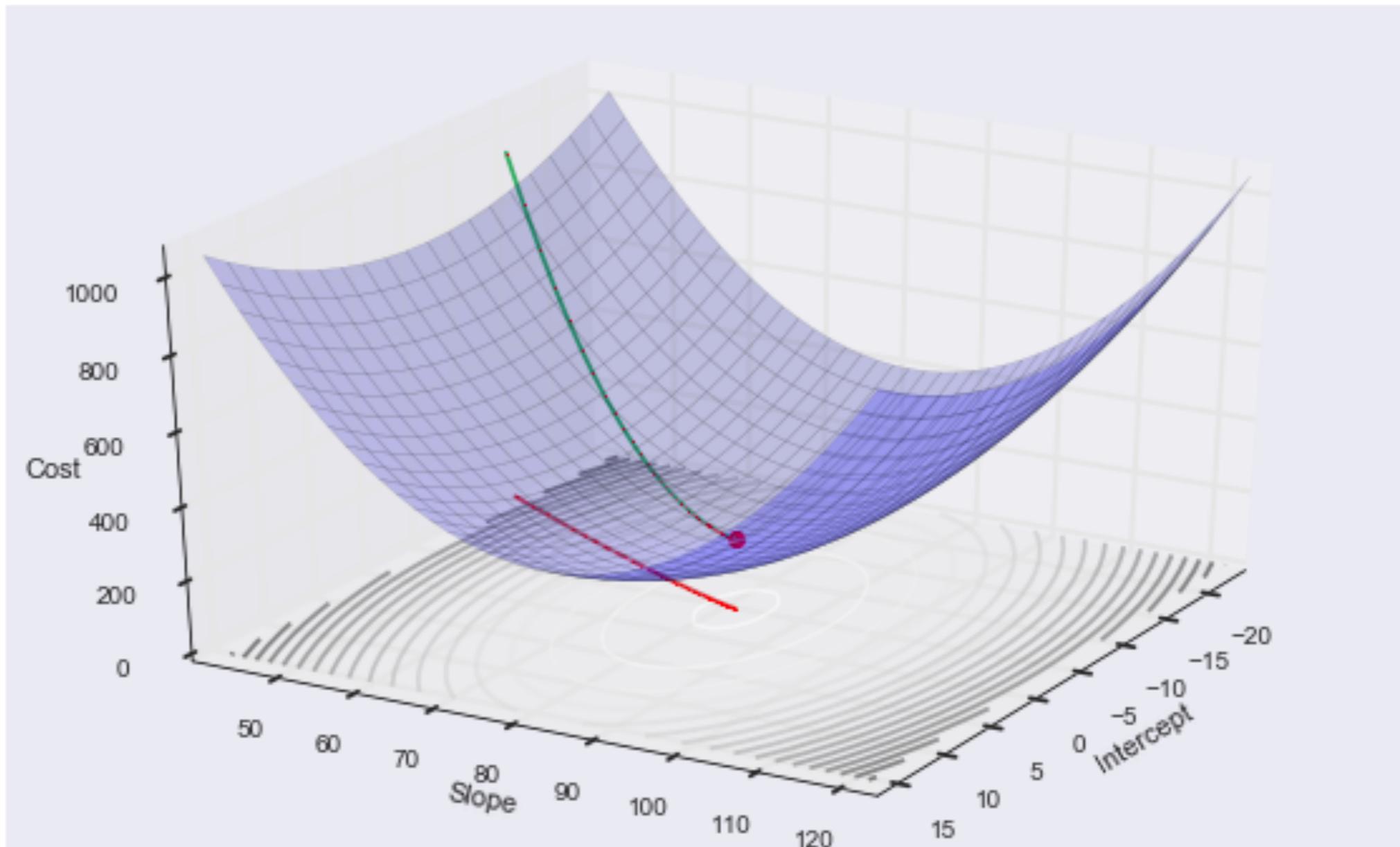
**02-Classification1-assignment.ipynb**

**03-Classification2-assignment.ipynb**

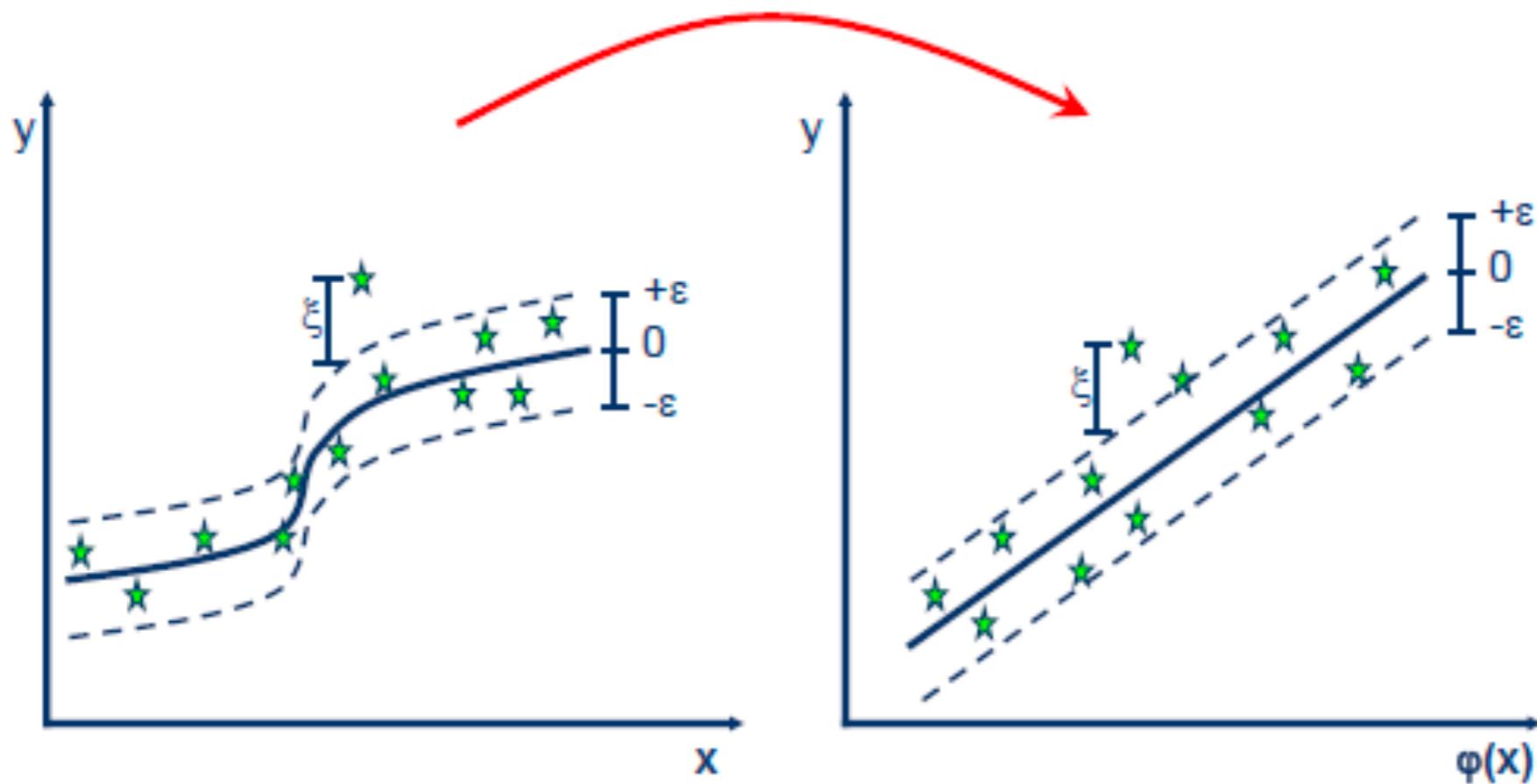
# Regression



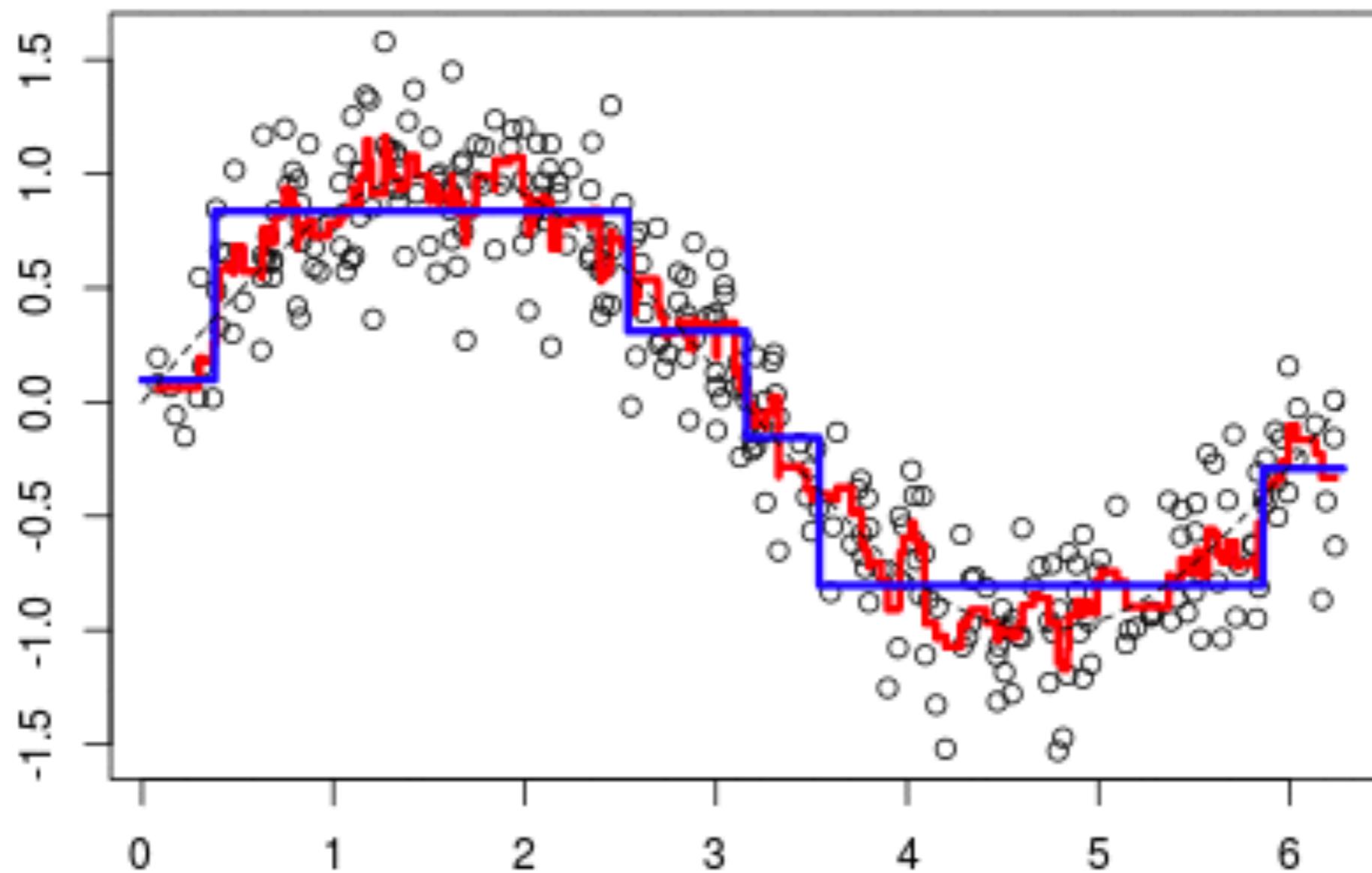
# Linear regression with SGD



# Support vector machines for Regression



# Boosted regression trees



# Evaluation of regression models

**Root mean squared error**

$$\text{RMSE} = \sqrt{\frac{\sum_i (y_i - \hat{y}_i)^2}{n}}$$

**Mean absolute error**

$$\text{MAE} = \frac{\sum_i |y_i - \hat{y}_i|}{n}$$

**R Squared**

$$\hat{R}^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

# Regression task

[04-Regression1-assignment.ipynb](#)

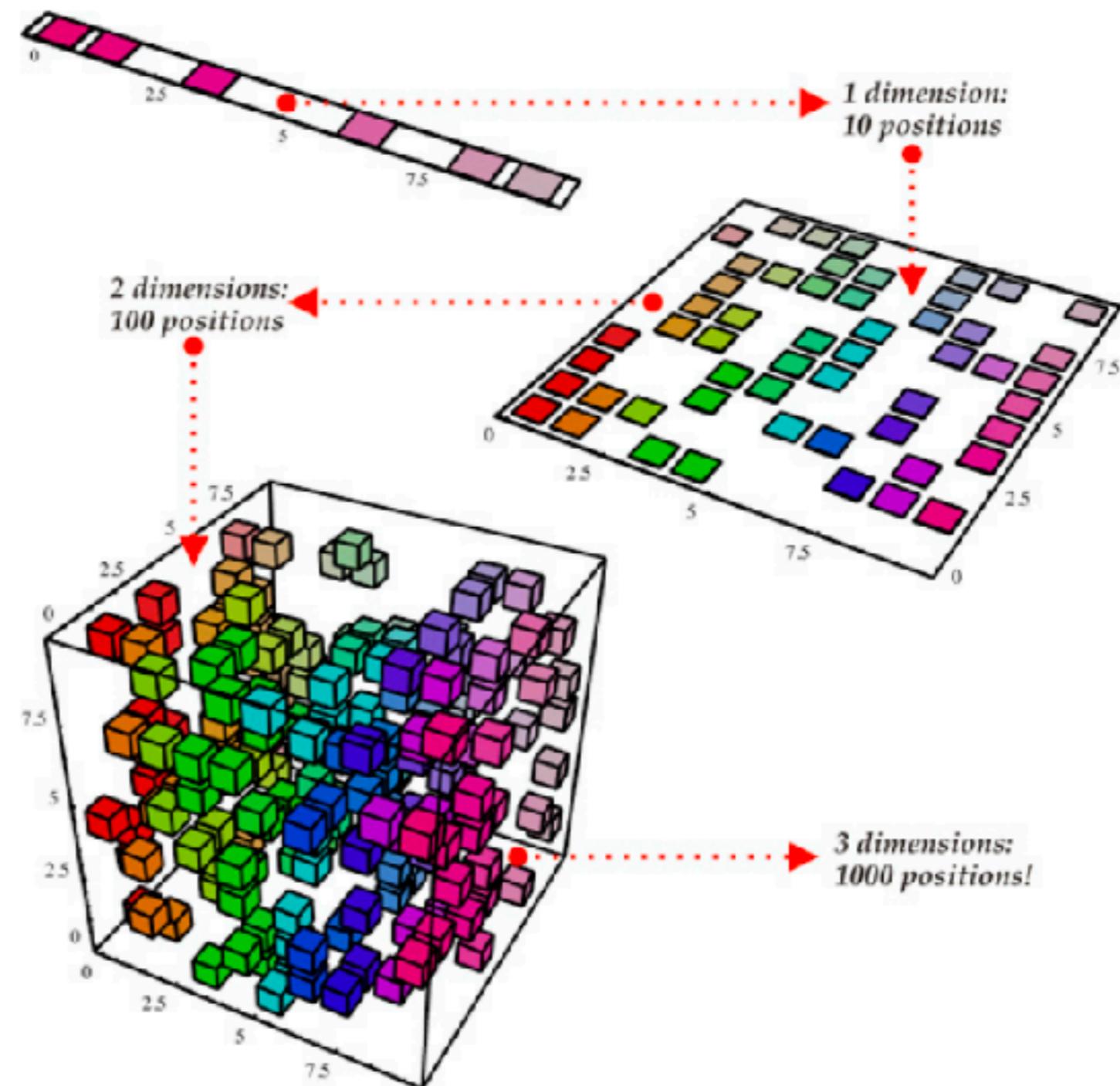
[05-Regression2-assignment.ipynb](#)

# Unsupervised techniques

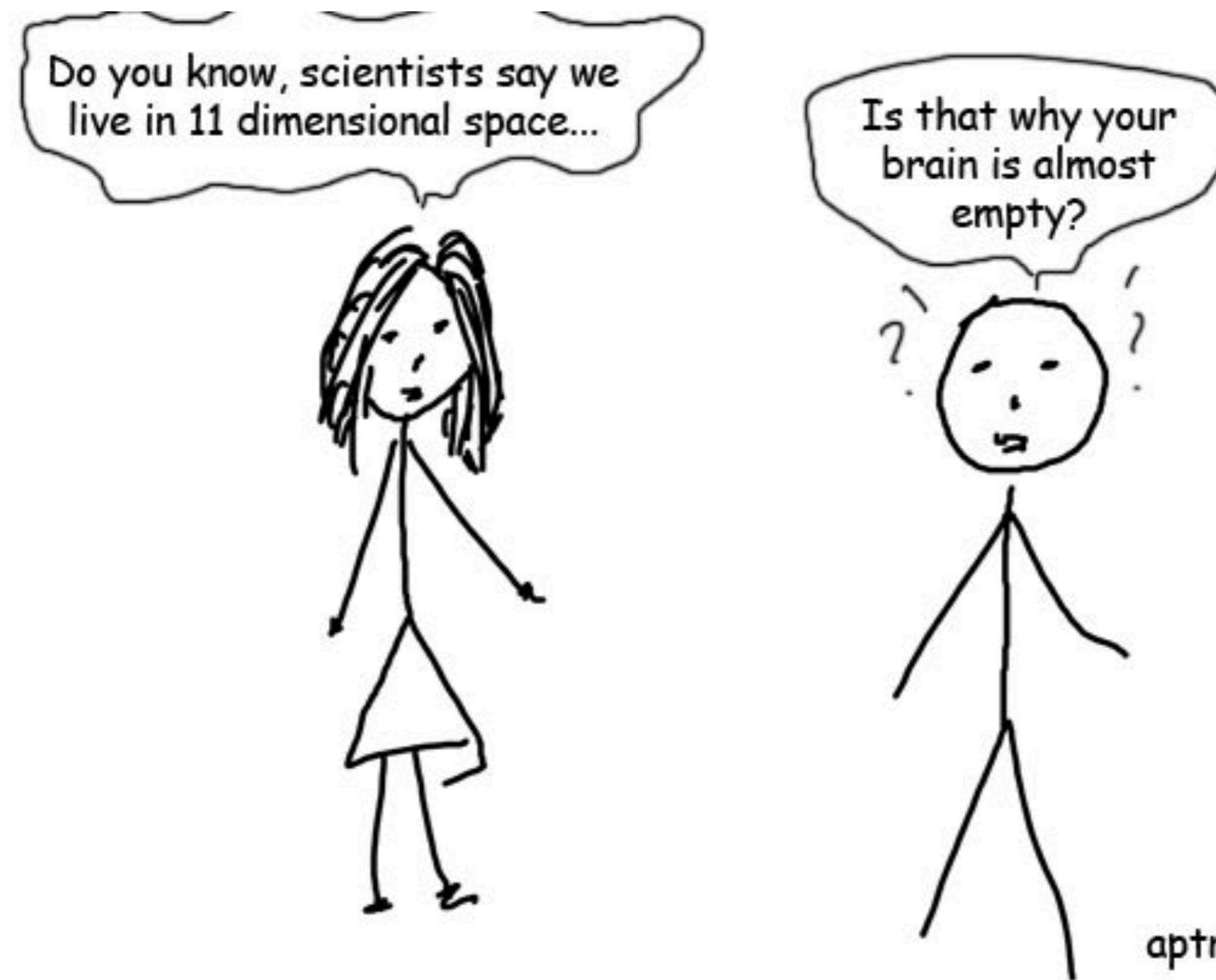


Source: tmdb.org

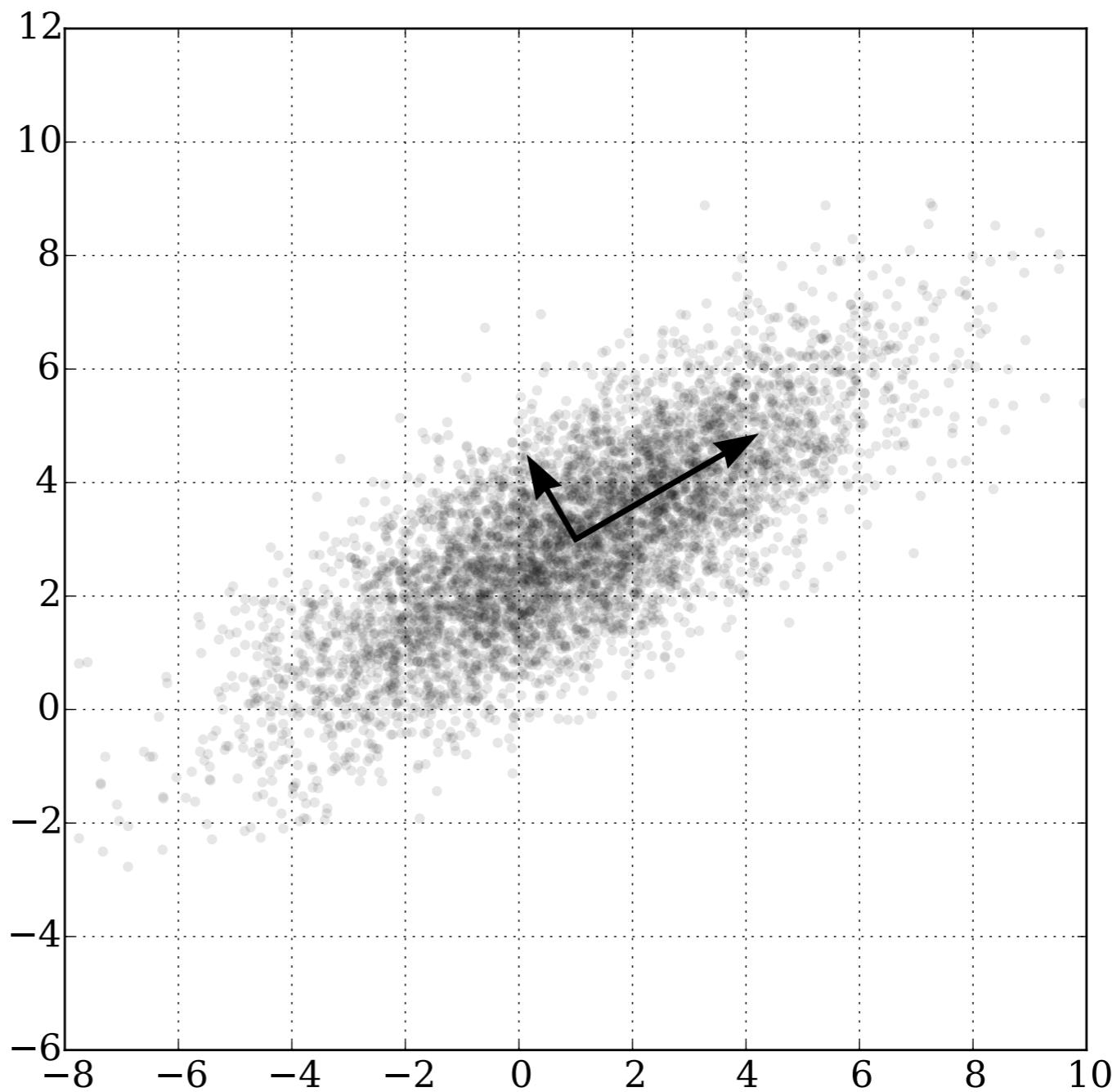
# Dimensionality reduction



# Curse of dimensionality



# Principal Component Analysis



# t-SNE and UMAP

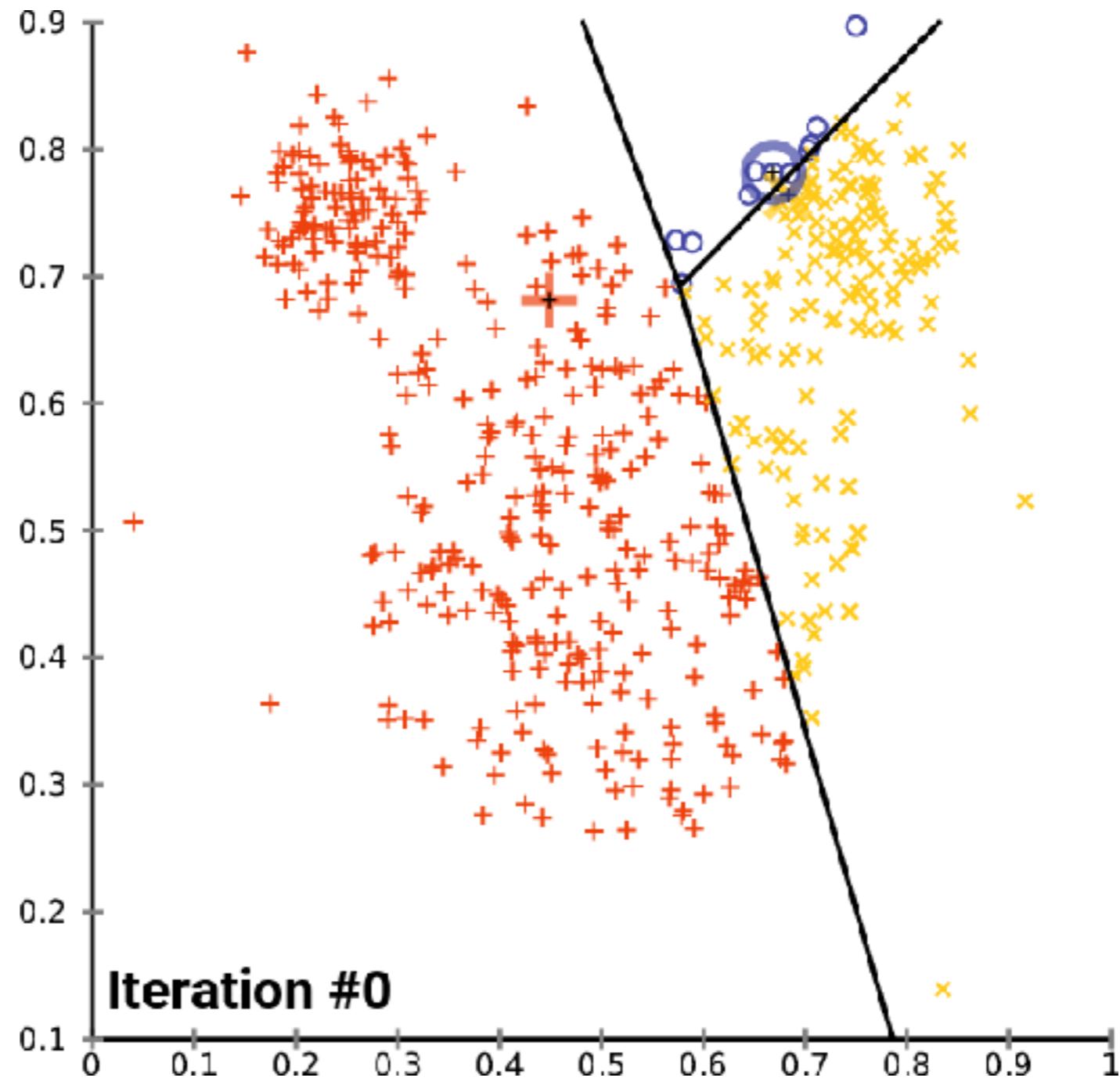
## dimensionality reduction



# Dimensionality reduction task

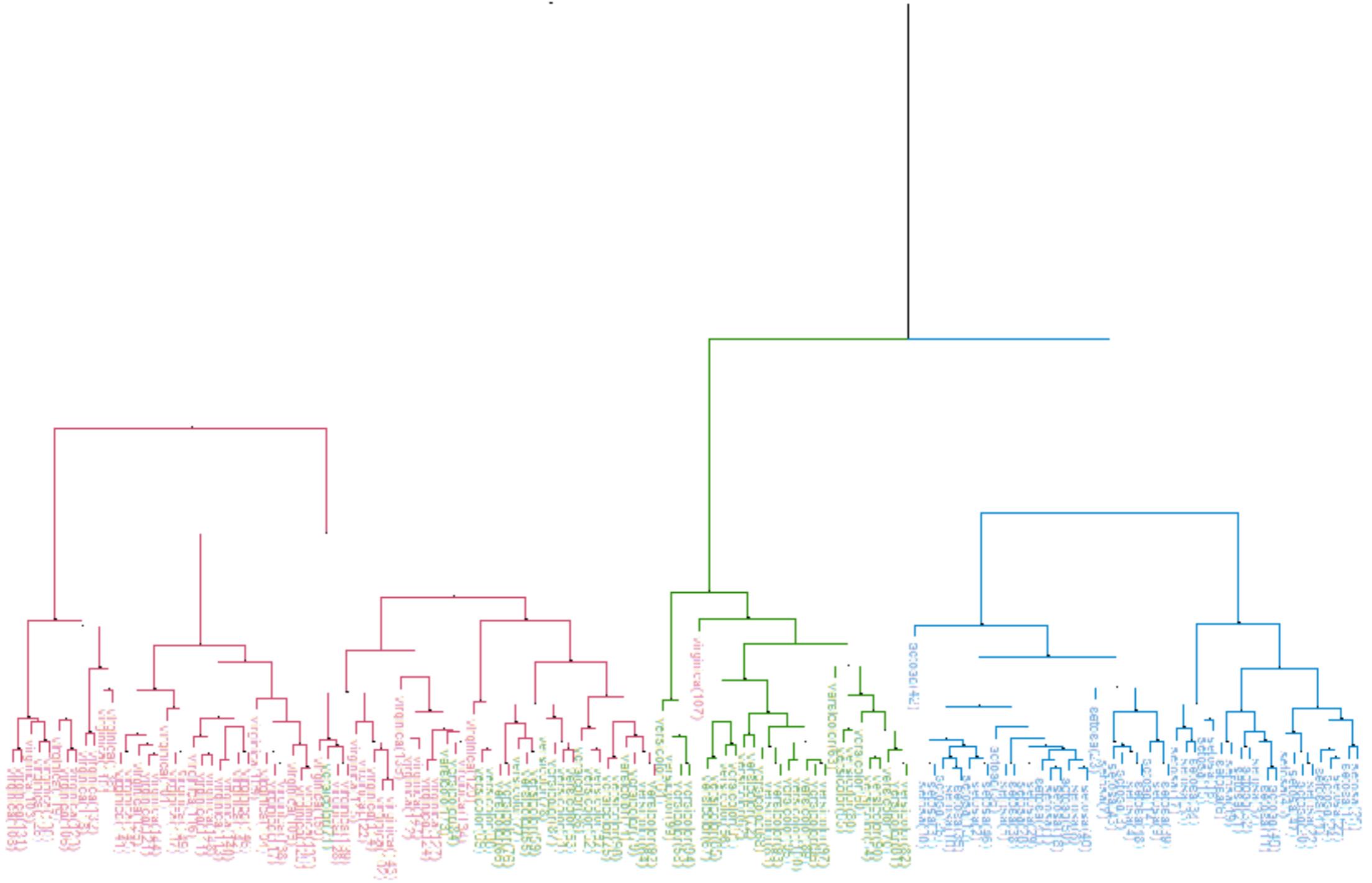
[06-Dim-reduction-assignment.ipynb](#)

# k-means clustering



Source: <https://wikipedia.org>

# Agglomerative clustering

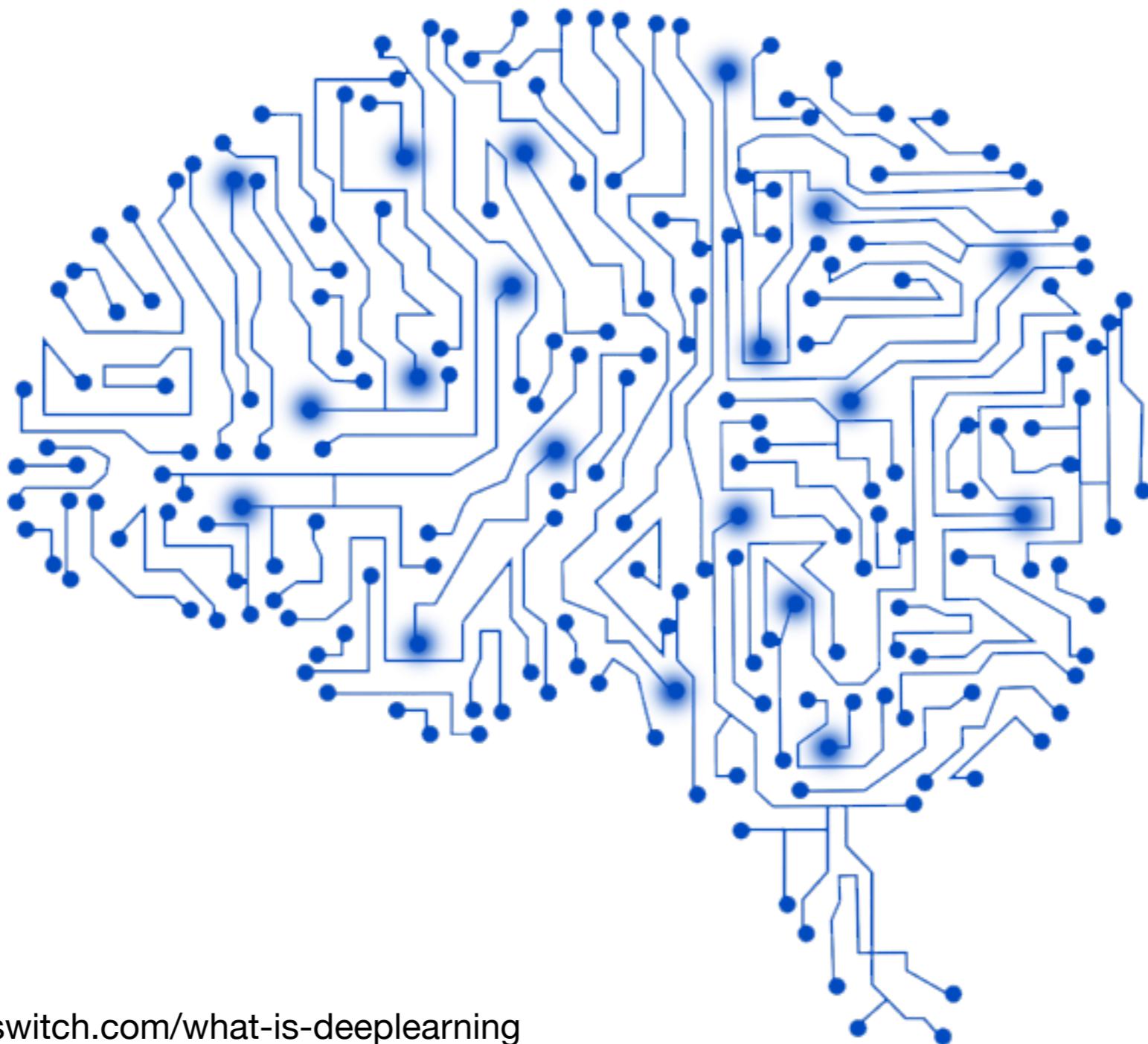


Source: <https://wikipedia.org>

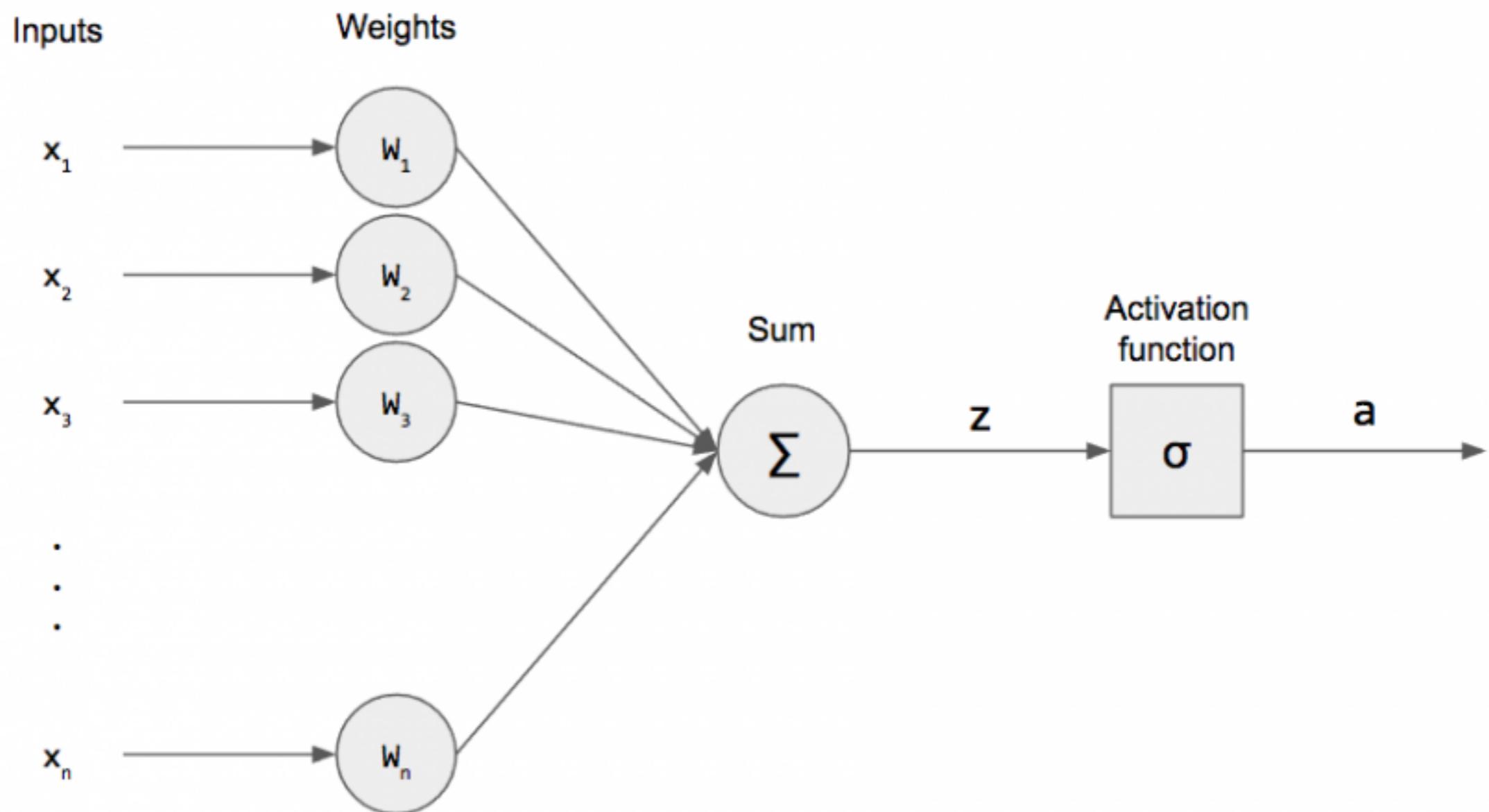
# Clustering task

[\*\*07-Clustering-assignment.ipynb\*\*](#)

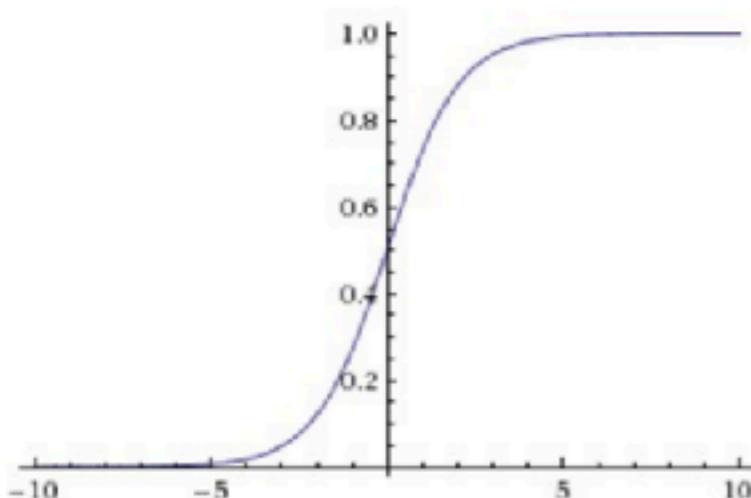
# Neural networks and deep learning



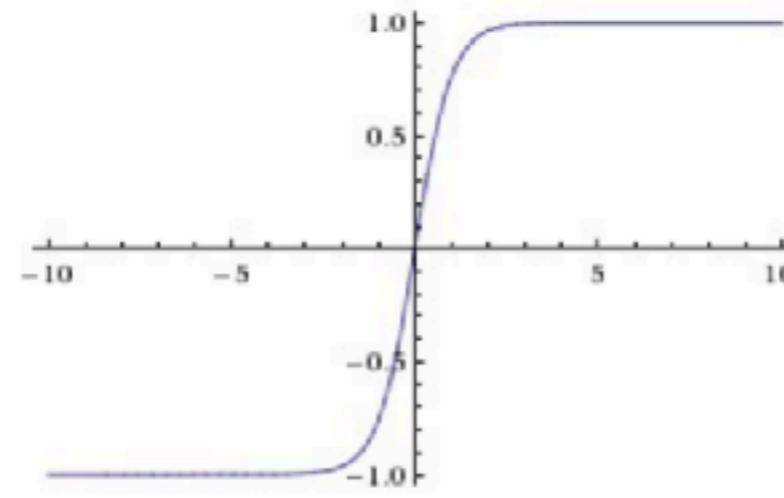
# Perceptron



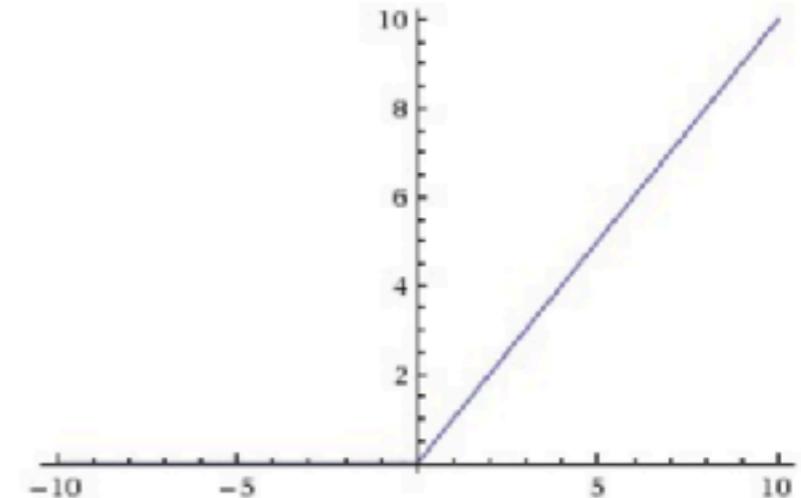
# Activation functions



Sigmoid



tanh

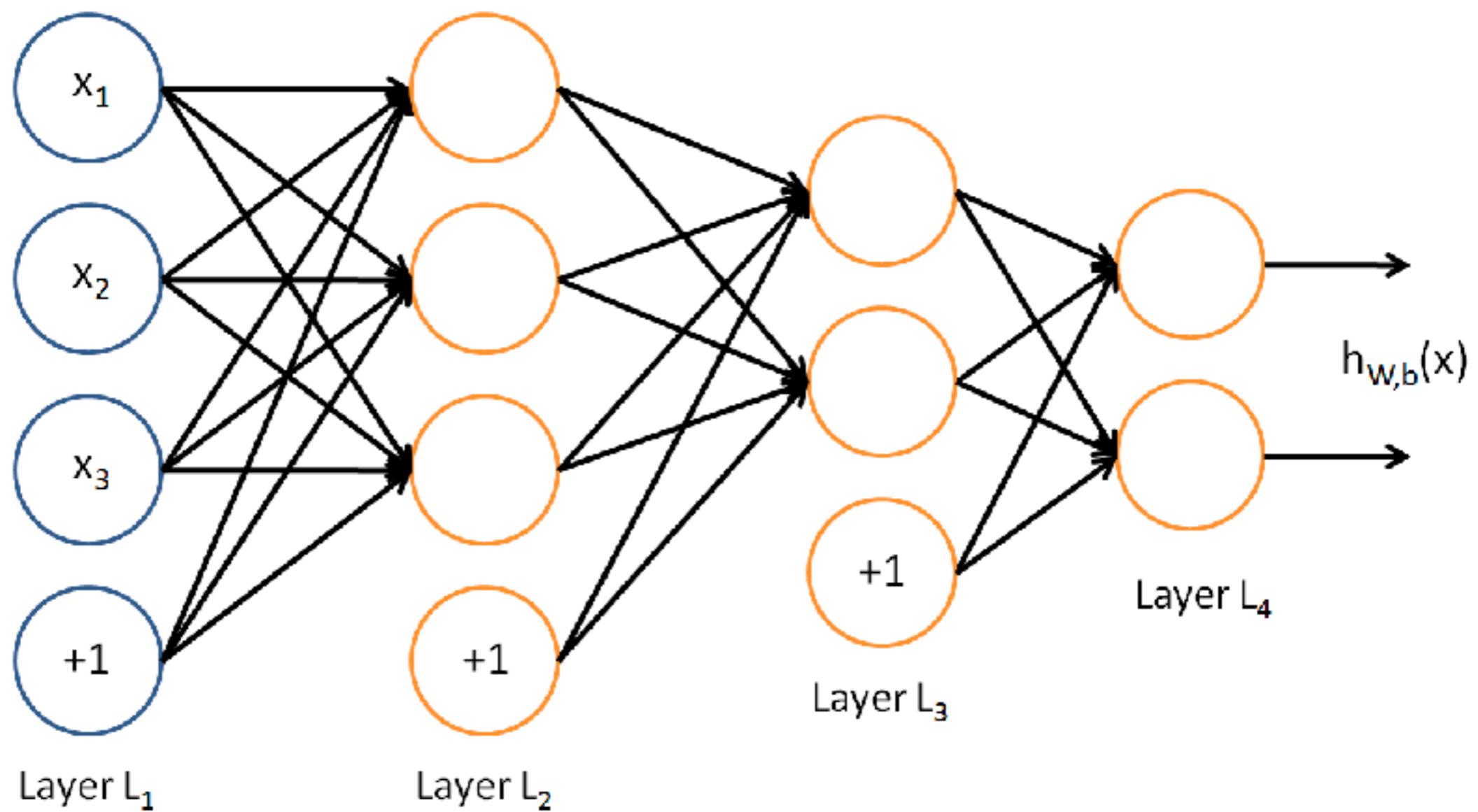


ReLU

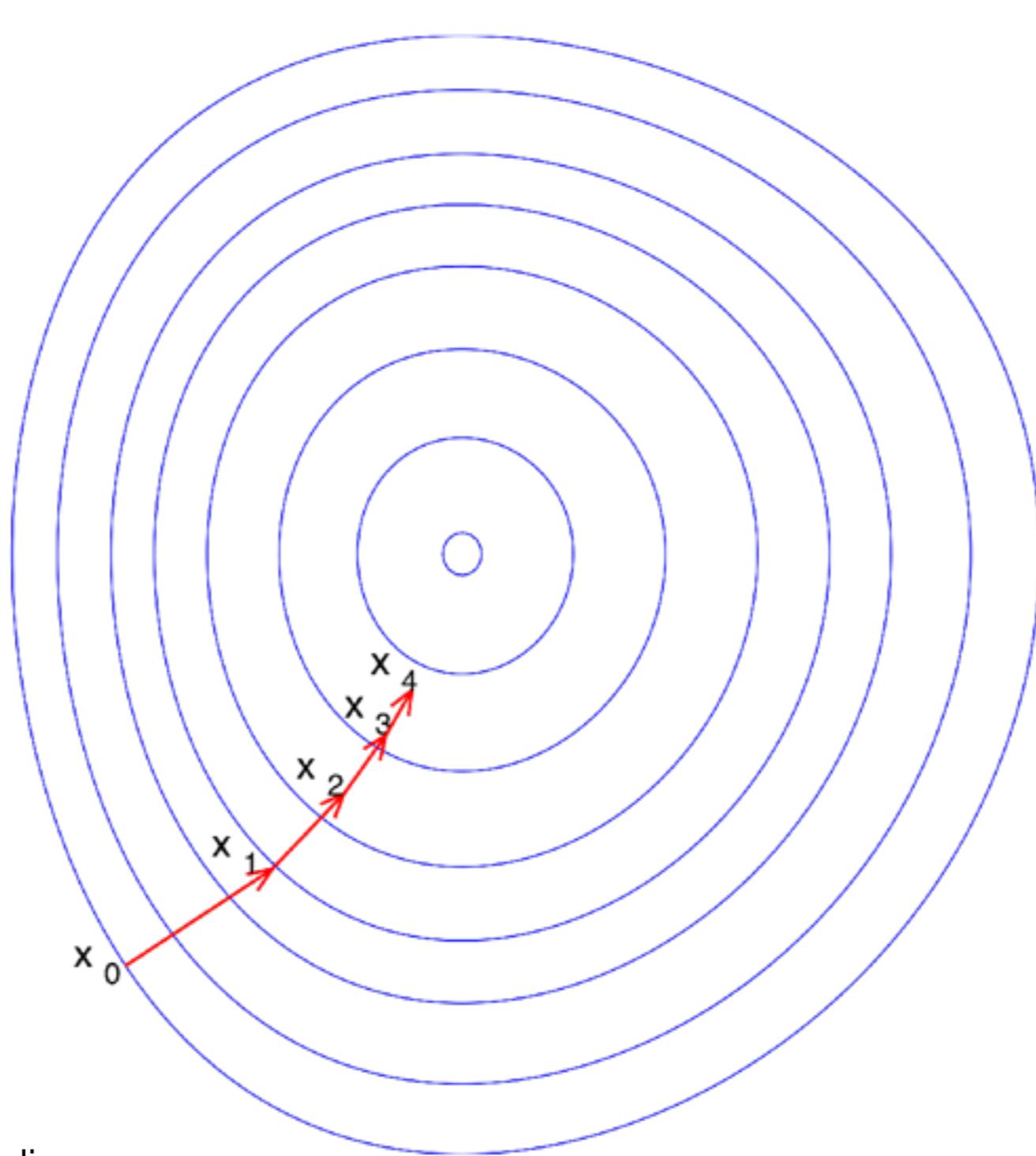
Softmax:

$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$$

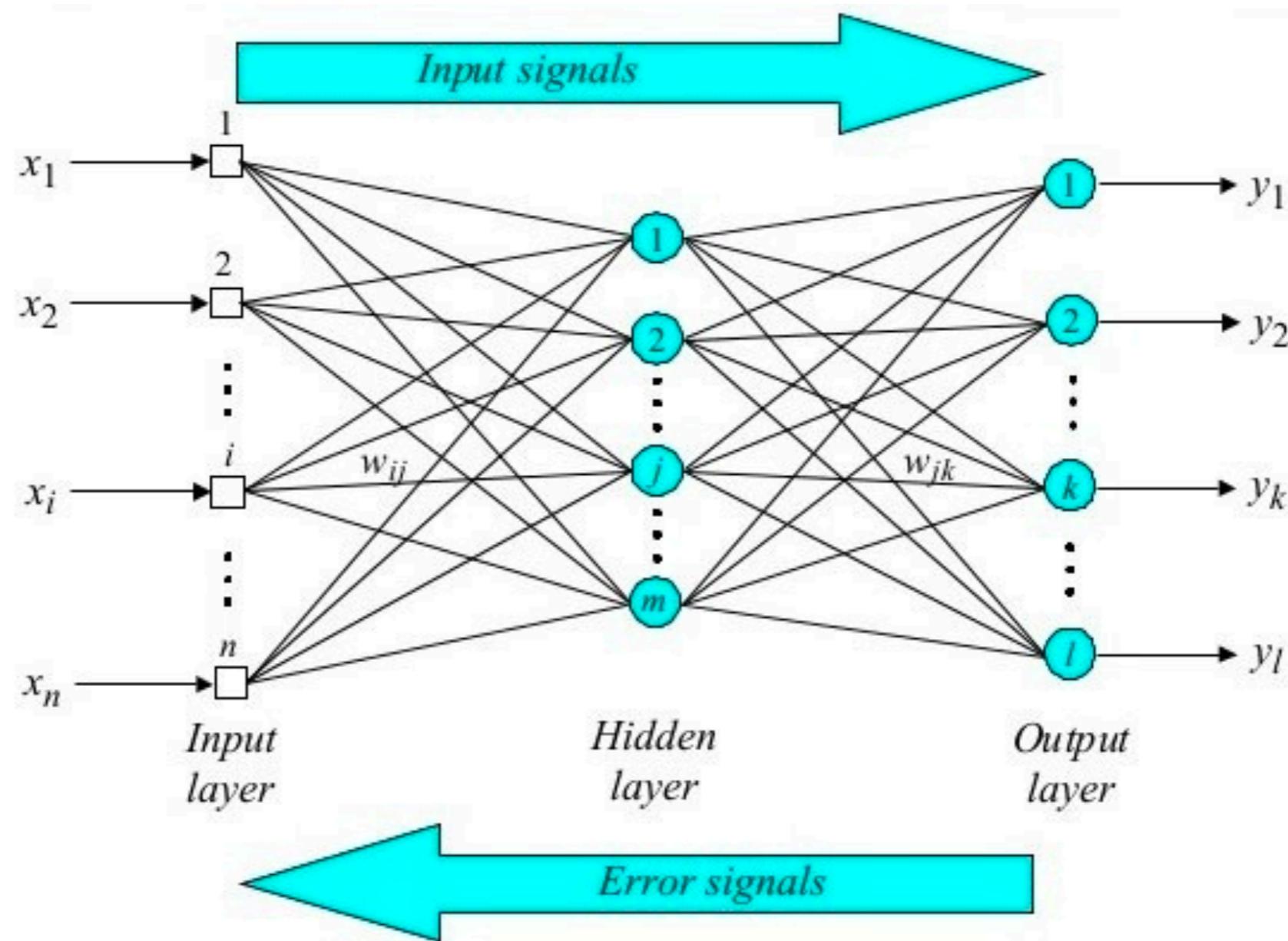
# Multilayer Neural Networks



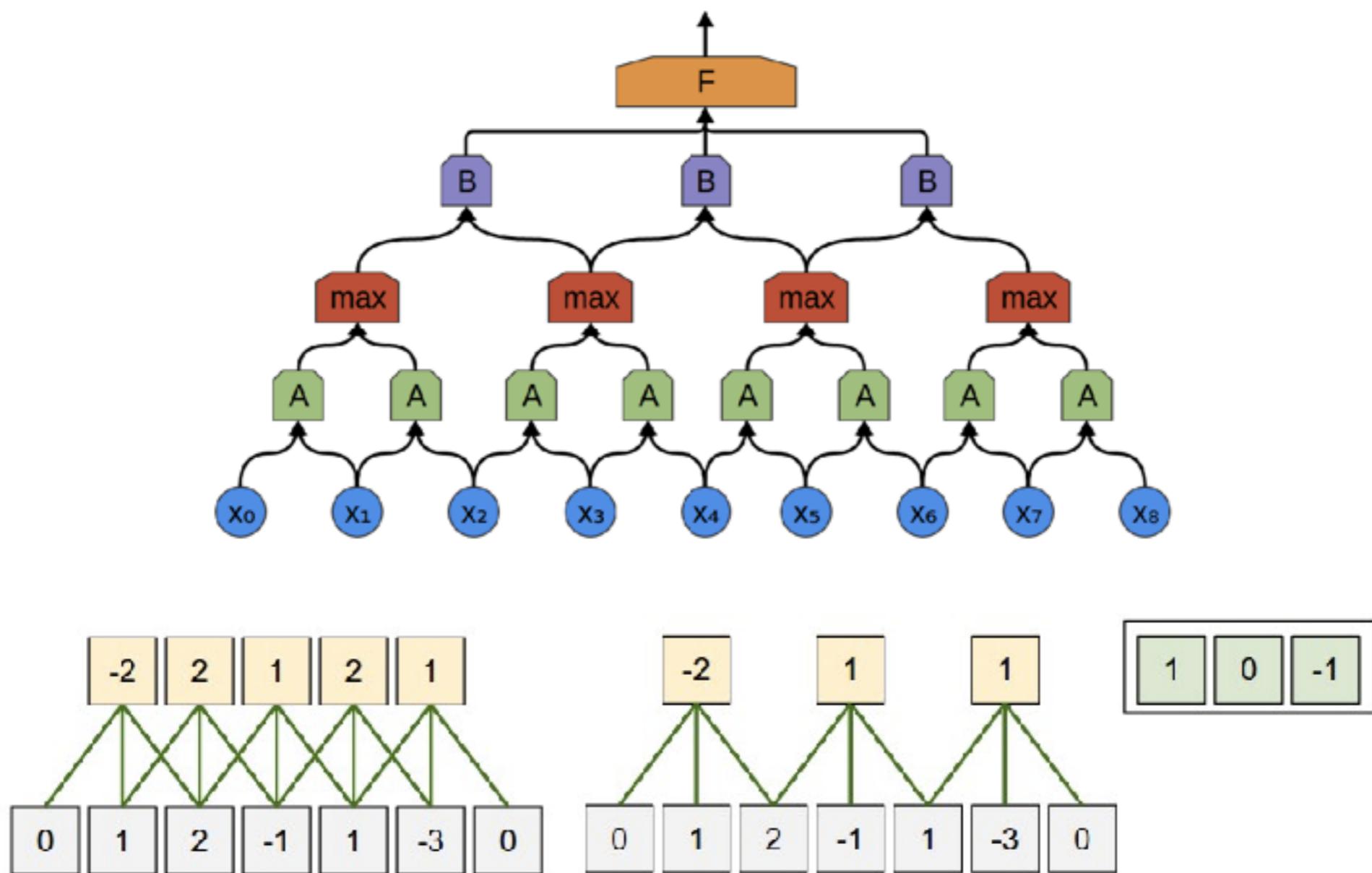
# Steepest gradient descent



# Back propagation

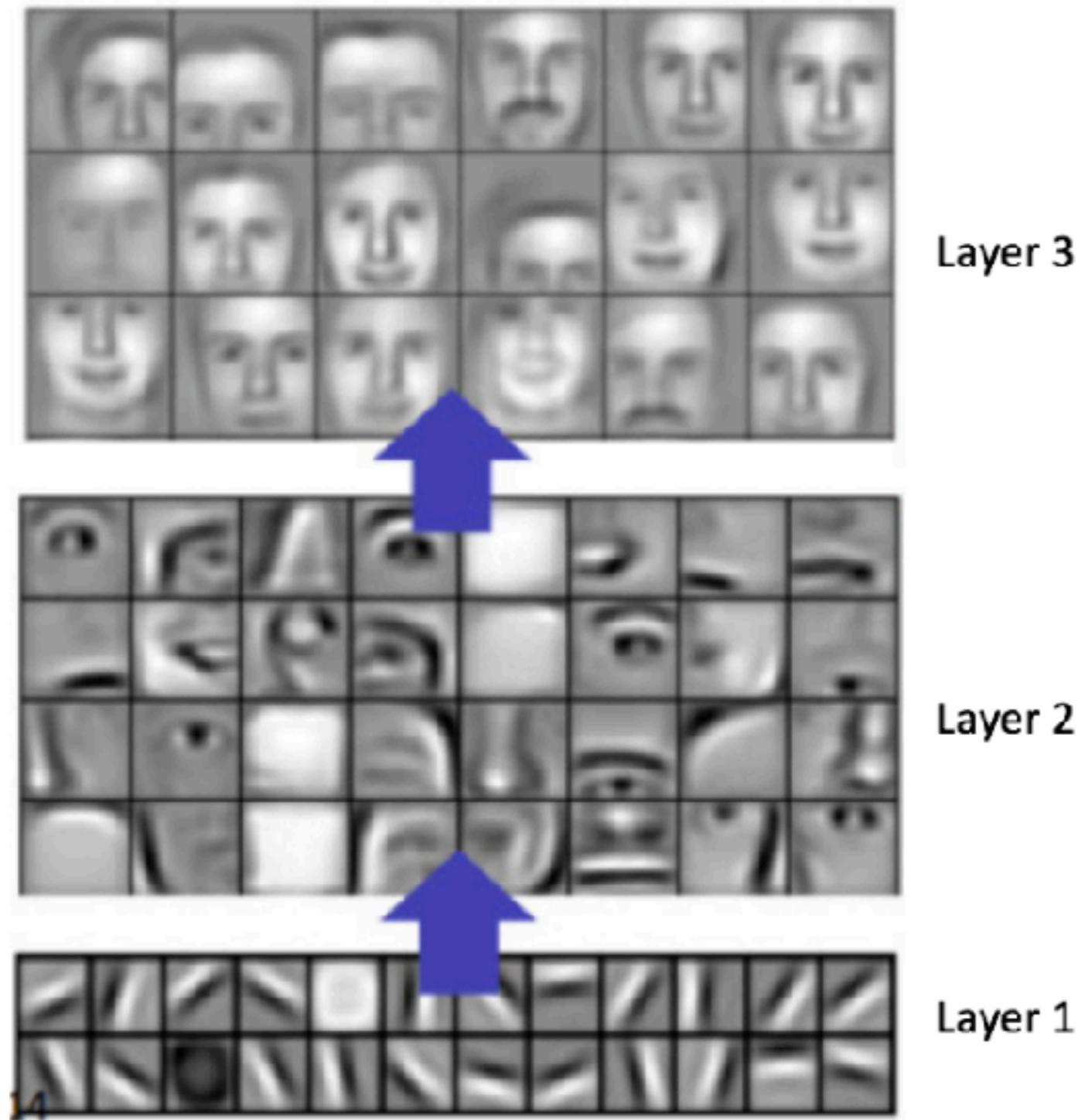


# Convolution



Source: <https://www.tensorflow.org>

# Weights visualization



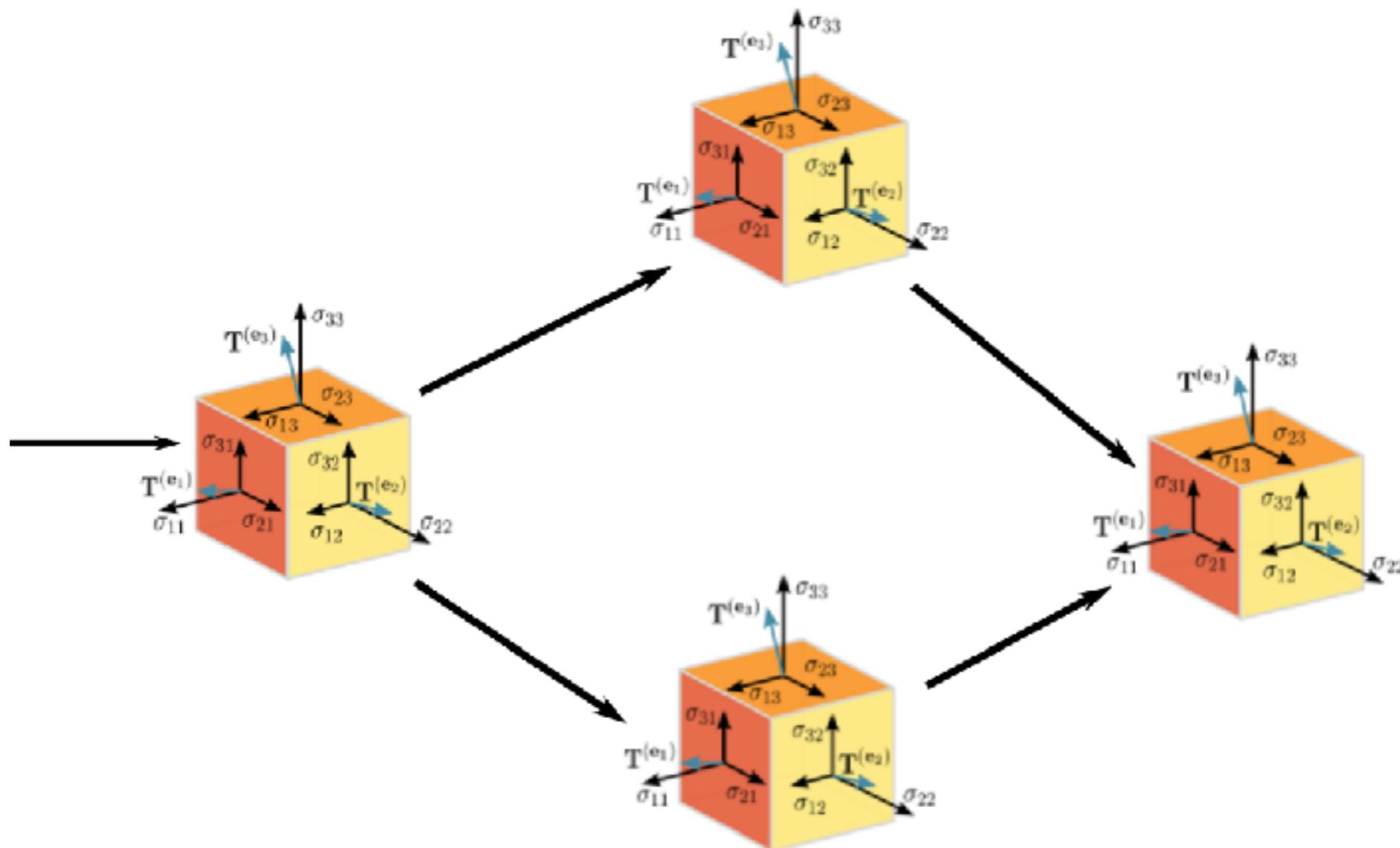
# Important terms

- deep learning
- stochastic gradient descent
- batch and mini-batch learning
- epoch
- dropout

# What is not TensorFlow



# What is TensorFlow?



# Keras tutorial

[08-Keras-introduction.ipynb](#)

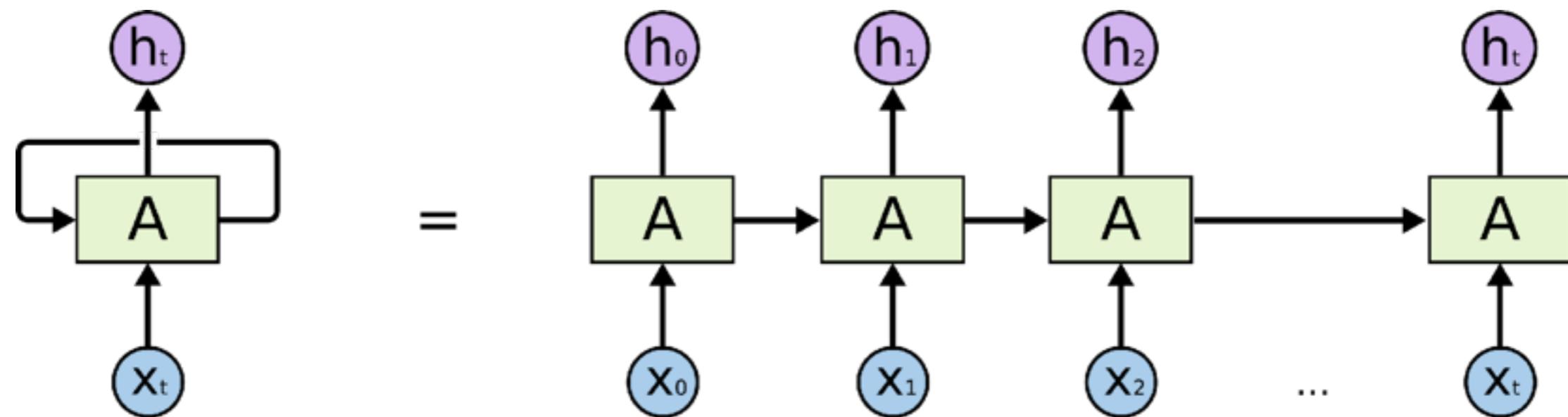
# Reimplementation of the classification and regression task using NN

[\*\*09-Classification-nn-assignment.ipynb\*\*](#)

[\*\*10-Regression-nn-assignment.ipynb\*\*](#)

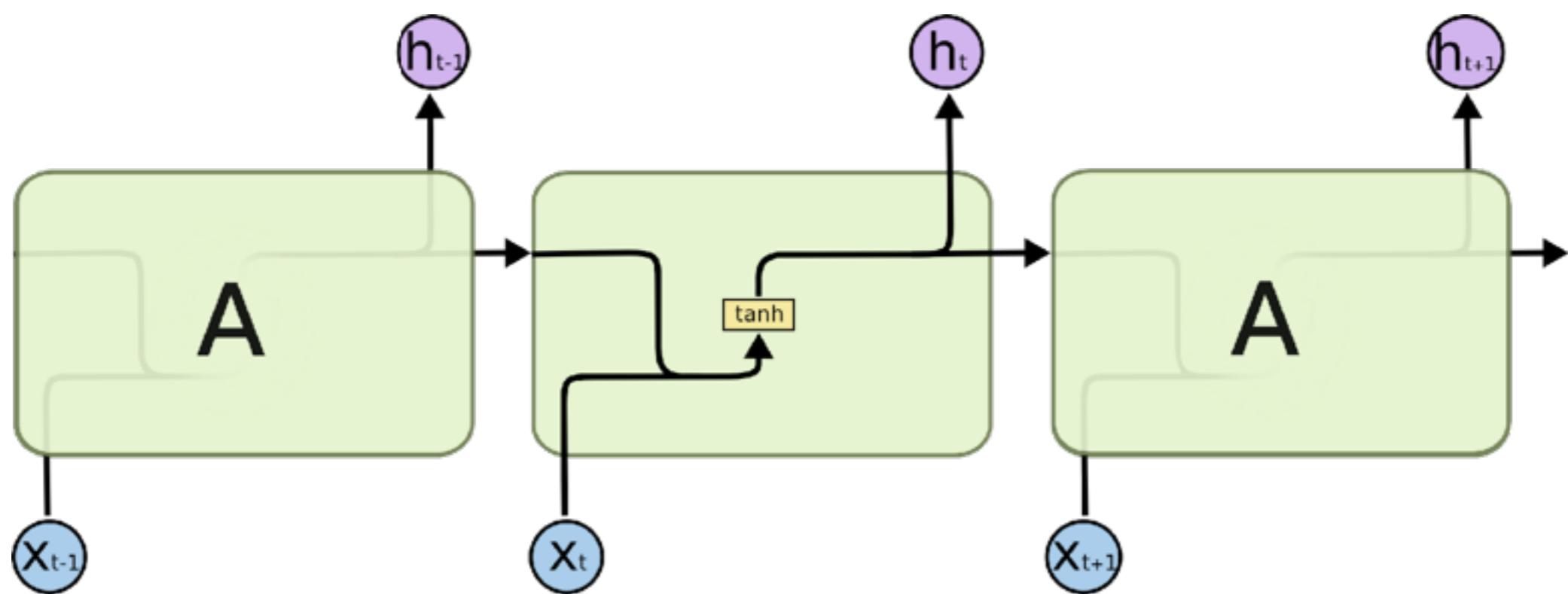
# Recurrent Neural Networks

## 1/2

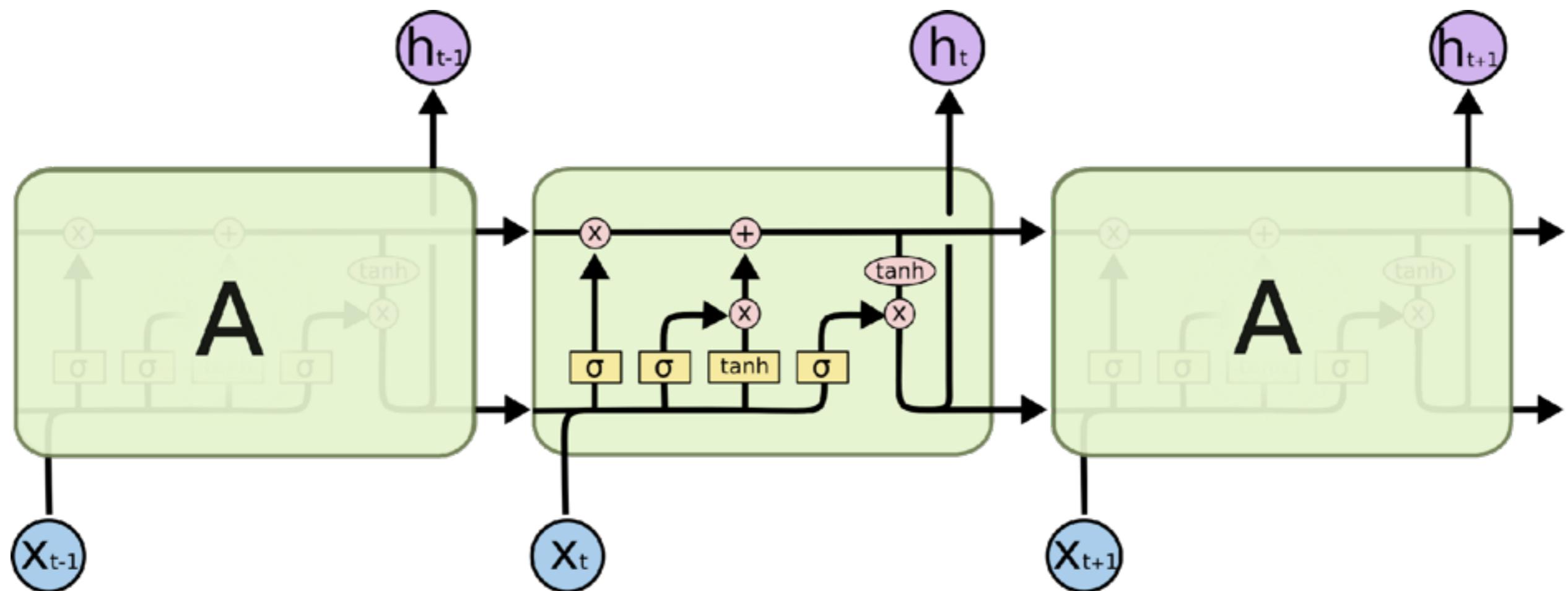


# Recurrent Neural Networks

2/2

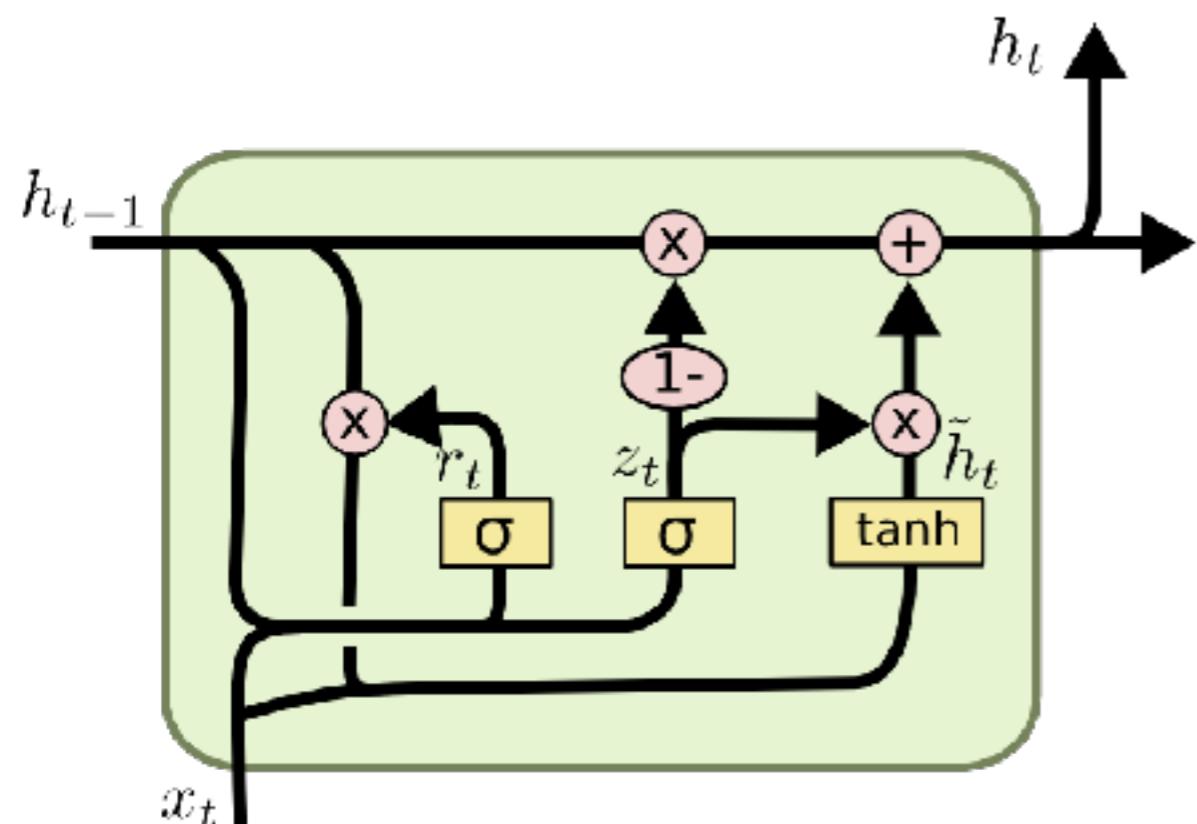


# Long Short-Term Memory



Source: <https://www.tensorflow.org>

# Gated Recurrent Unit



$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$$

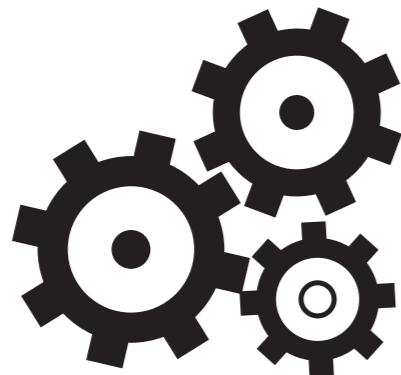
$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t])$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

# Reimplementation of the regression task using Recurrent neural network

[11-Regression-rnn.ipynb](#)

# What next?



Machine Learning Prague

**ML** MACHINE LEARNING  
**MU** meetups

**<http://www.mlguru.com>**

# Thank you for your attention

**e-mail:** jiri@mlguru.com

**Web:** www.mlguru.com

**Twitter:** @JiriMaterna

**Facebook:** <https://www.facebook.com/maternajiri>

**LinkedIn:** <https://www.linkedin.com/in/jirimaterna/>