University of Science, VNU-HCM
Faculty of Information Technology

Introduction to Data Science Course

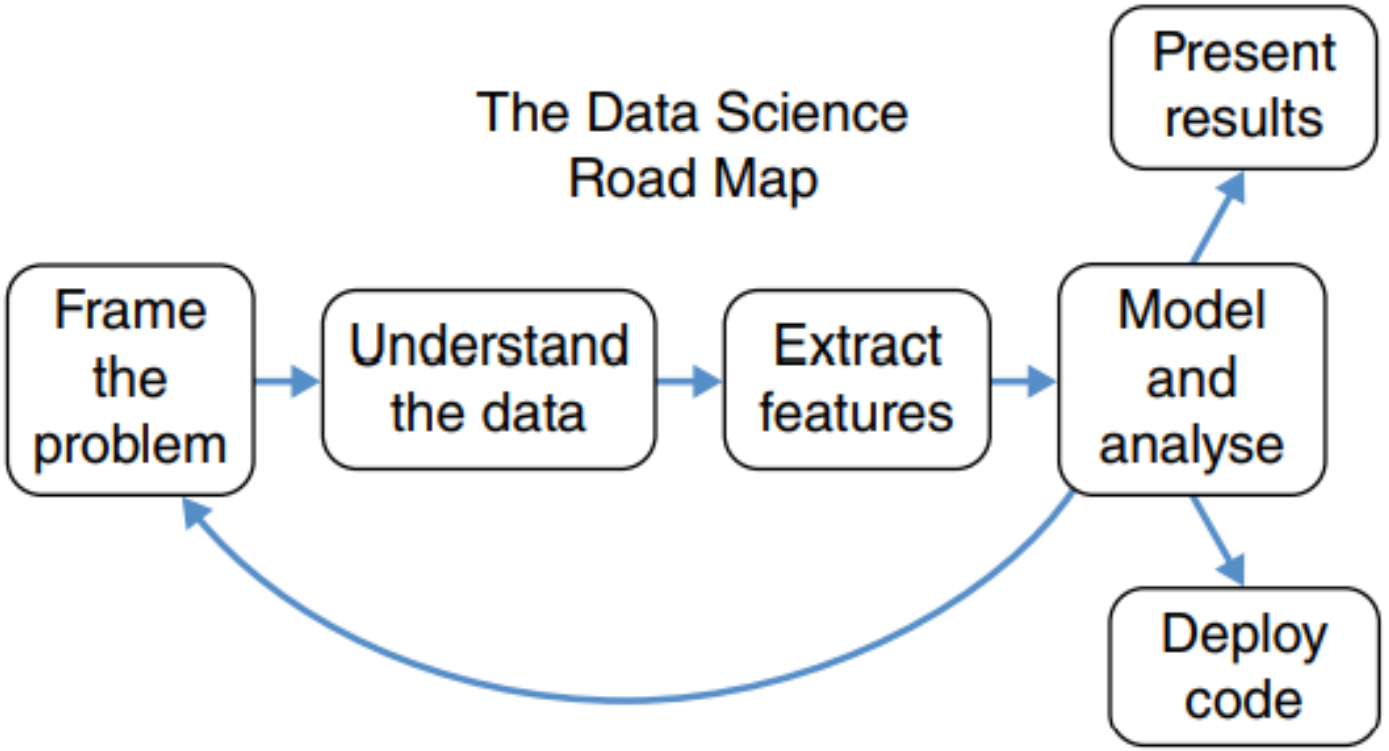
Data Modeling (Part 2)

Le Ngoc Thanh
lnthanh@fit.hcmus.edu.vn
Department of Computer Science

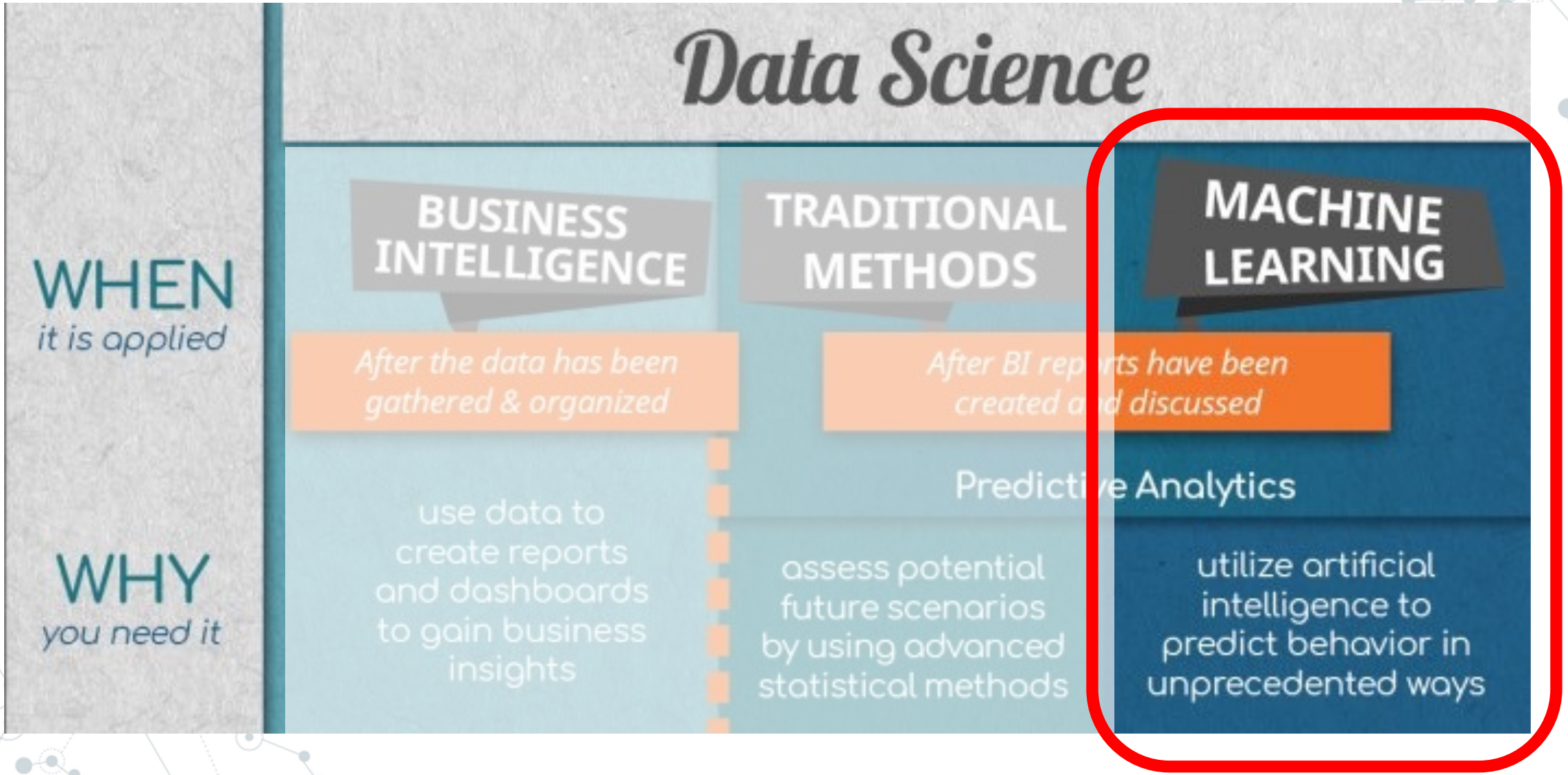
Ho Chi Minh City

Contents

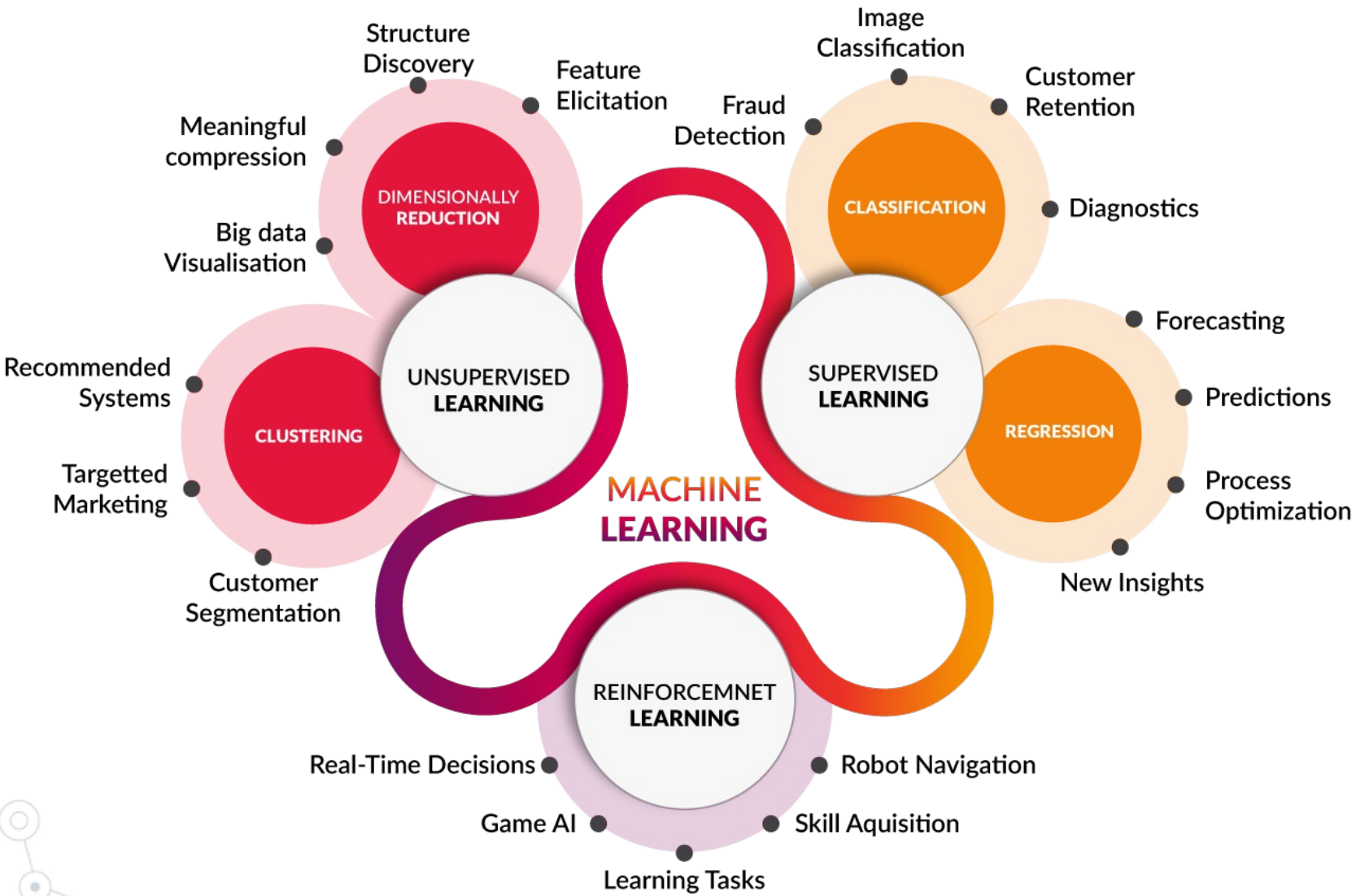
Process


The Data Science
Road Map
Frame the problem
Understand the data
Extract features
Model and analyse
Present results
Deploy code

Data Science's tasks

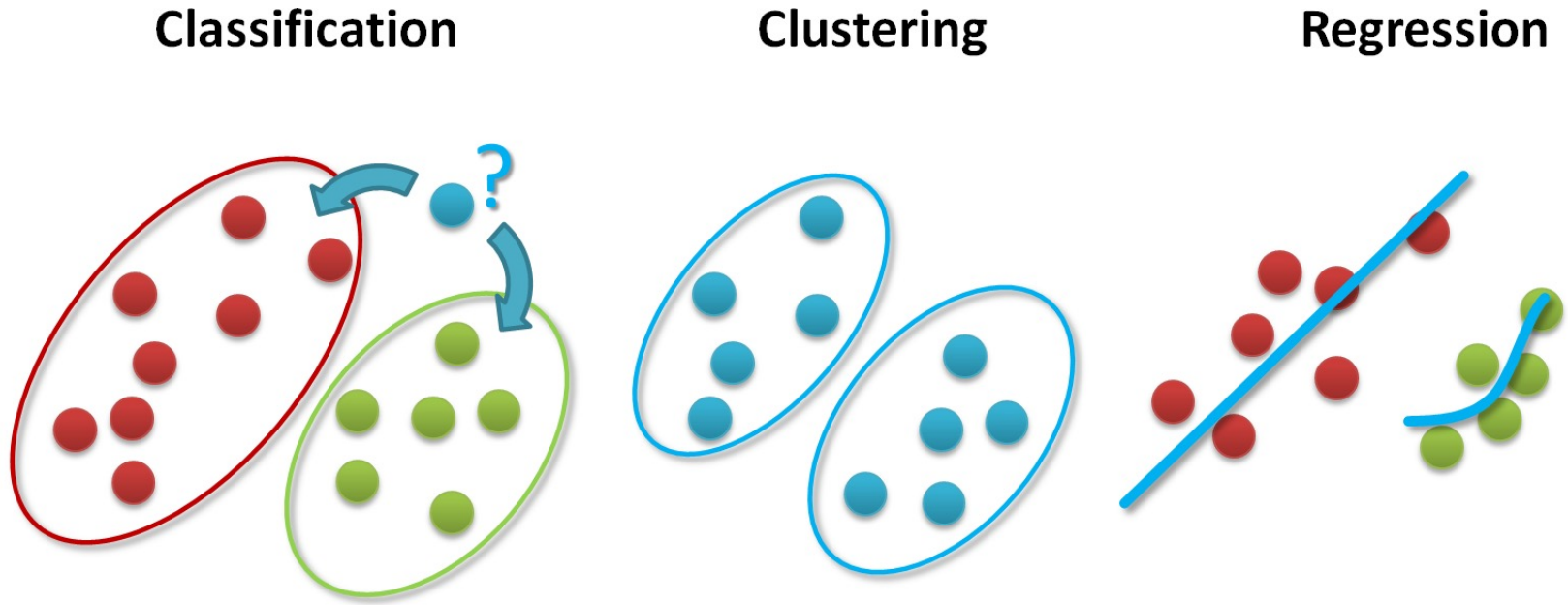


ML Tasks

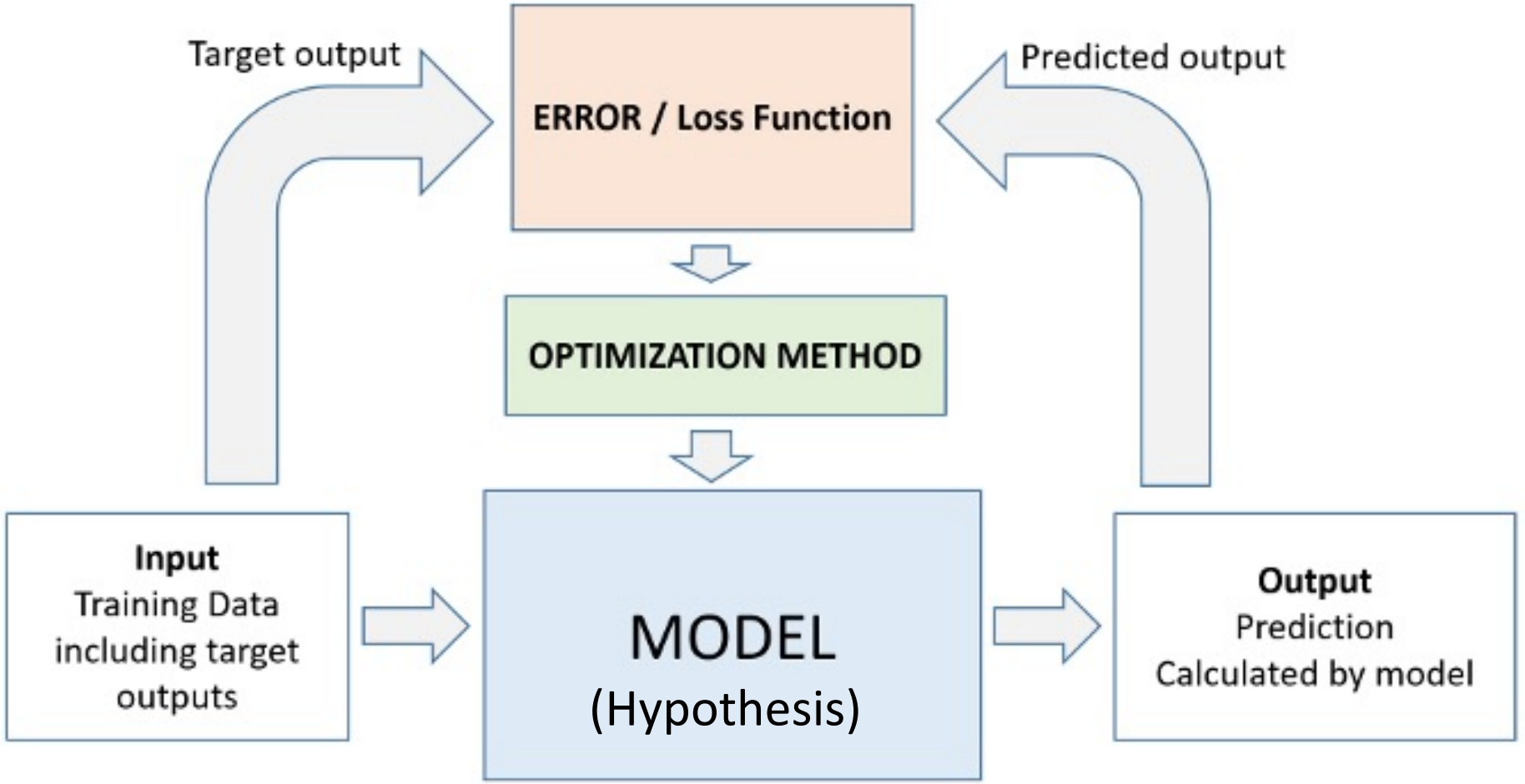

MACHINE LEARNING
UNSUPERVISED LEARNING
SUPERVISED LEARNING
REINFORCEMNET LEARNING
DIMENSIONALLY REDUCTION
CLUSTERING
CLASSIFICATION
REGRESSION
Structure Discovery
Feature Elicitation
Meaningful compression
Big data Visualisation
Image Classification
Fraud Detection
Customer Retention
Diagnostics
Recommended Systems
Targetted Marketing
Customer Segmentation
Forecasting
Predictions
Process Optimization
New Insights
Real-Time Decisions
Robot Navigation
Game AI
Skill Aquisition
Learning Tasks

The course's focus

- In this course, we focus on three main groups of ML:
  - Regression
  - Classification
  - Clustering

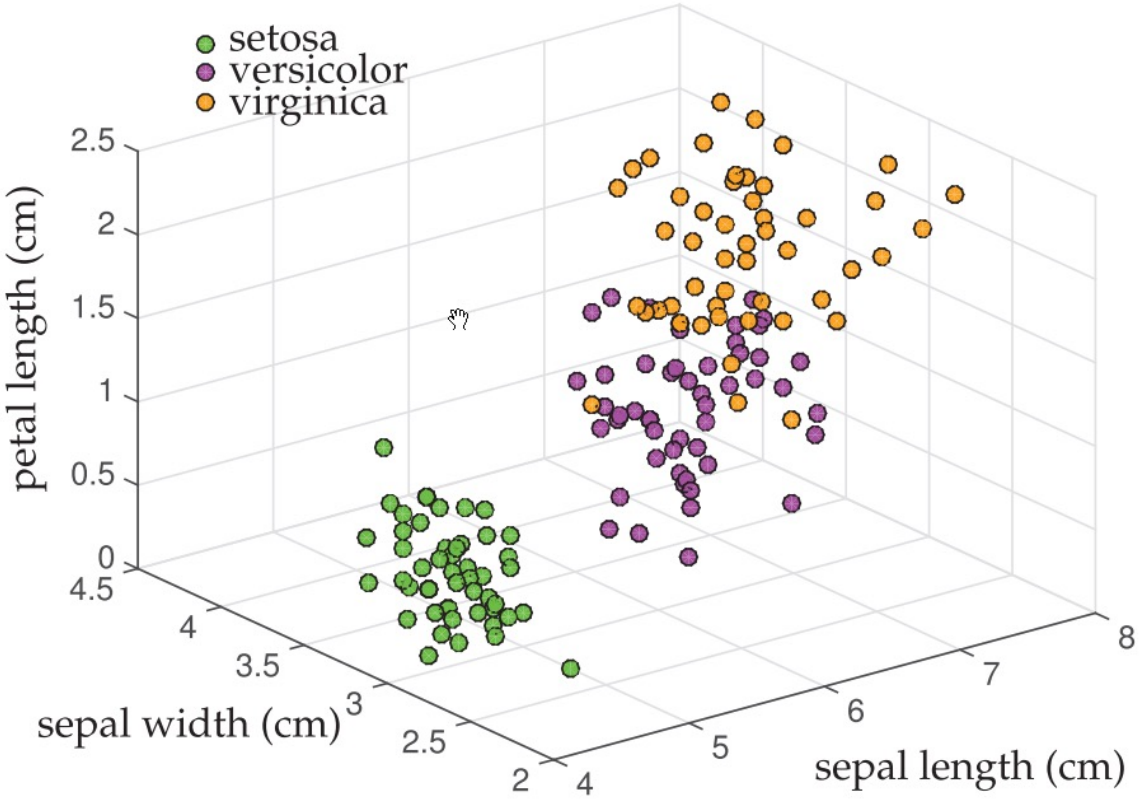


General model learning architecture



Classification

- Classification is the problem of identifying which of a set of categories an observation belongs to.



Classification

- The inputs and outputs for the learning binary classification task can be stated as follows:
  - Input

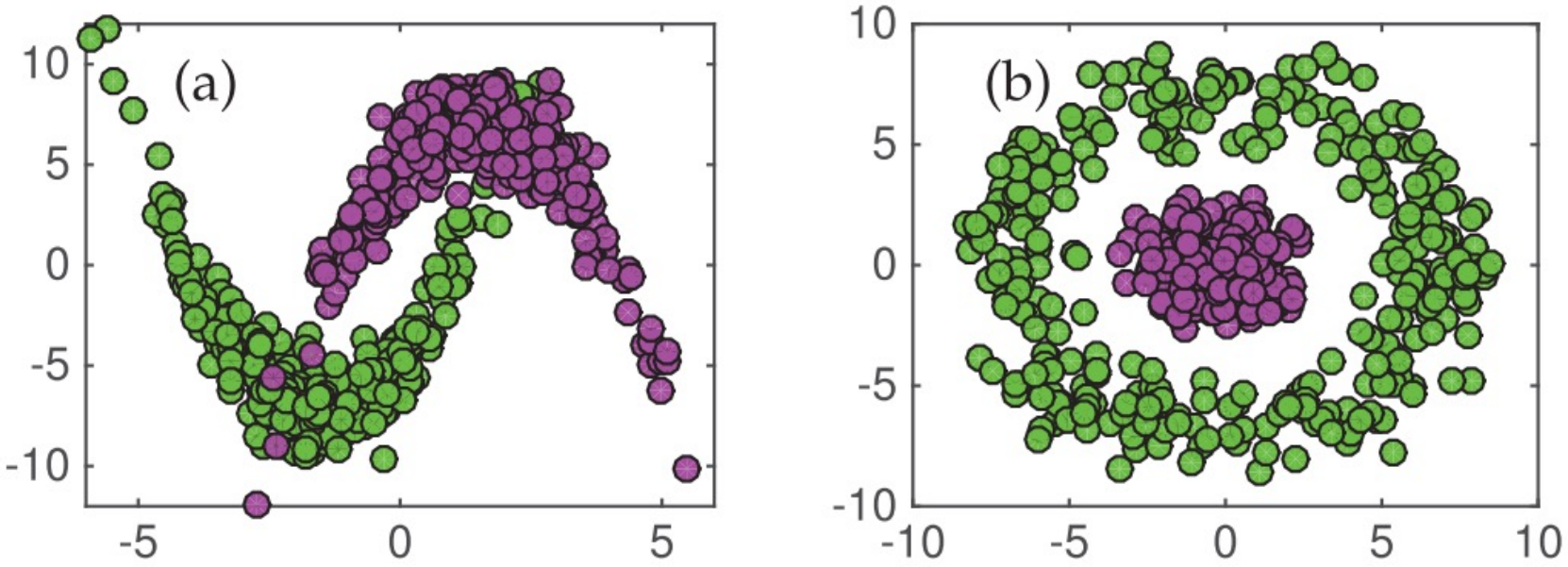
    data $\left\{\mathbf{x}_j \in \mathbb{R}^n, j \in \mathrm{Z} := \{1,2,\dots,m\}\right\}$

    labels $\left\{\mathbf{y}_j \in \{\pm 1\}, j \in Z' \subset Z\right\}$
  - Ouput

    labels $\left\{\mathbf{y}_j \in \{\pm 1\}, j \in Z\right\}$

Challenges

- Some challenges in classification task:
  - The boundary between the data forms a nonlinear manifold that is difficult to characterize.
  - If the sampling data only captures a portion of the manifold, then it will almost surely fail in characterizing population data.
  - Data can be in higher dimensional space and visualization is essentially impossible.



Well-known classification algorithms

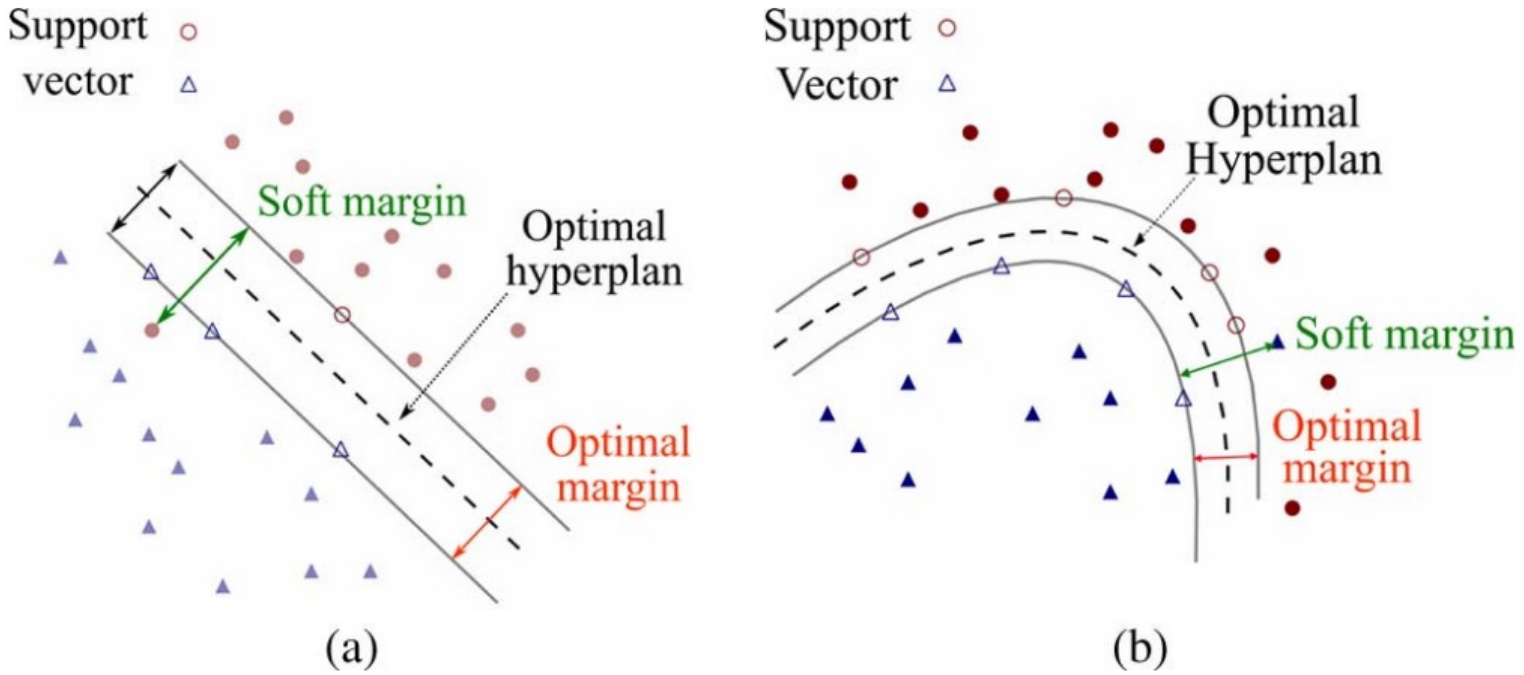
- Some well-known classifier:
  - Support Vector Machine (SVM)
  - Classification and Regression Tree (CART)
  - k-nearest Neighbors (kNN)
  - Naïve Bayes
  - Ensemble Learning and Boosting (AdaBoost)
  - Ensemble Learning of Decision Tree (C4.5)
  - Deep neural nets

Contents

- Data science and machine learning review
- Classification model
  - Support Vector Machine (SVM)
  - Neural network
- Clustering model

Support Vector Machines

- The original SVM algorithm by Vapnik and Chervonenkis evolved out of the statistical learning literature in 1963, where hyperplanes are optimized to split the data into distinct classes.



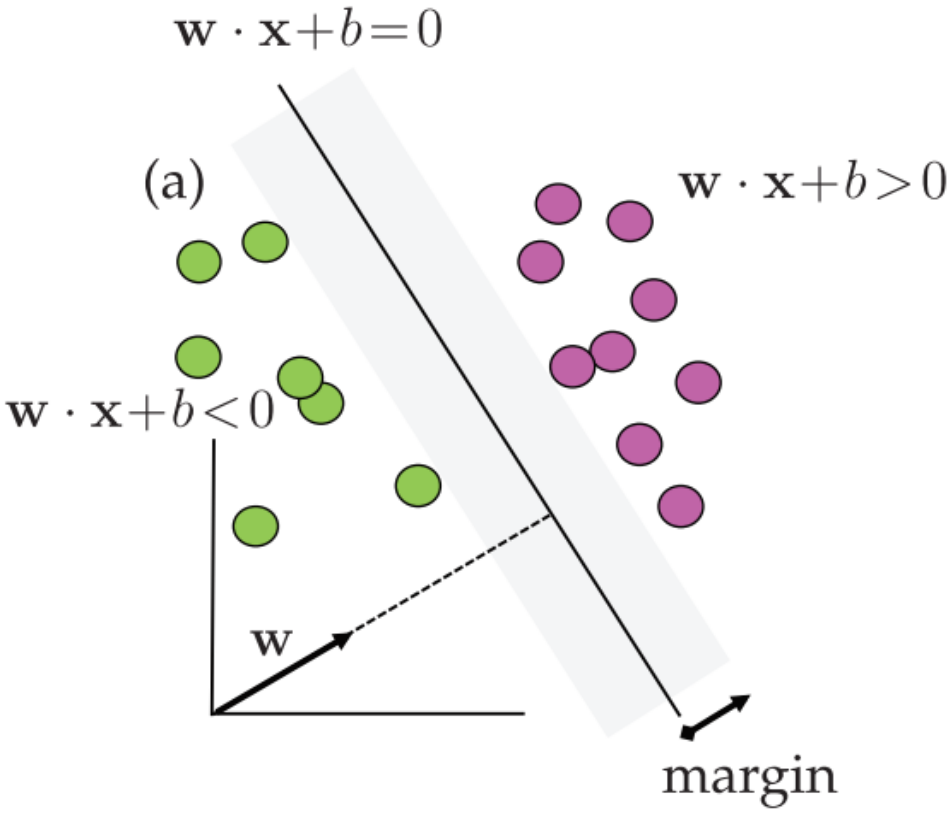
SVM classifier scheme. (a) Linear and (b) nonlinear cases.

Linear SVM

- The key idea of the linear SVM method is to construct a hyperplane:

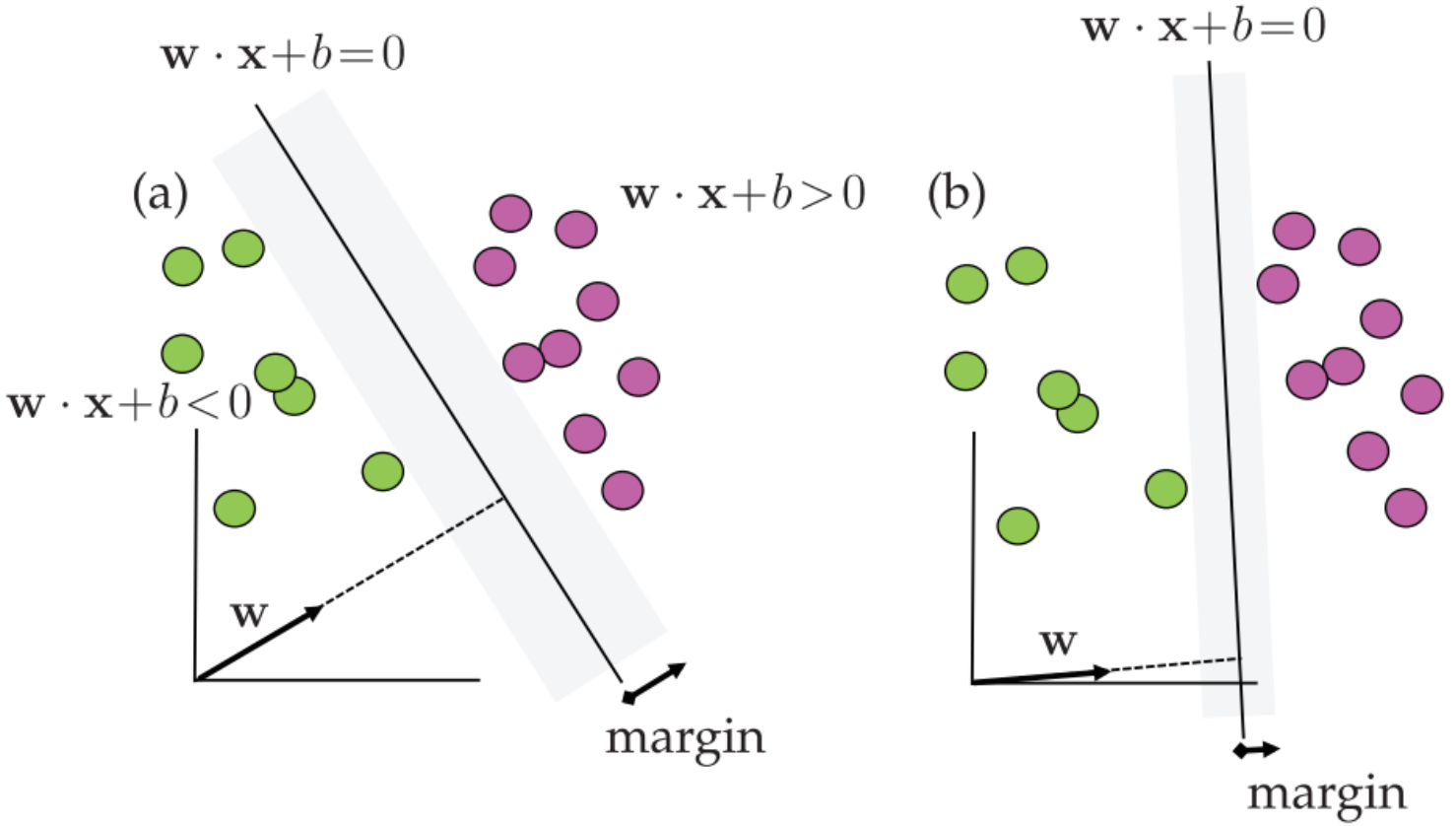
$$\mathbf{w} \cdot \mathbf{x} + b = 0$$

where the vector $\mathbf{w}$ and constant b parametrize the hyperplane.



Optimization problem in SVM

- The optimization problem in SVM includes:
  - Optimize a decision line which makes the fewest labeling errors.
  - Optimizes the largest margin between the data.



Objective function

- The loss function is defined as follows:

$$L(\mathbf{y}_j, \hat{\mathbf{y}}_j) = L\left(\mathbf{y}_j, \text{sign}(\mathbf{w} \cdot \mathbf{x}_j + b)\right)$$

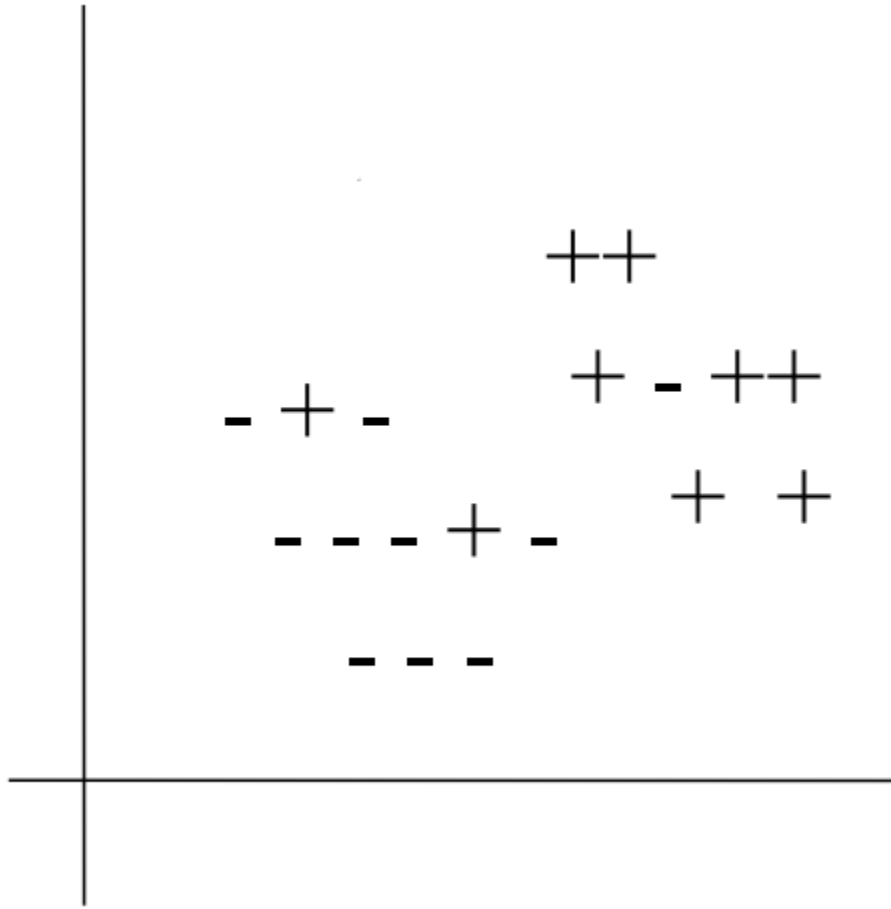
$$= \begin{cases} 0 \text{ if data is correctly labeled} \\ +1 \text{ if data is incorrectly labeled} \end{cases}$$

- The goal is also to make the margin as large as possible:

$$\text{argmin}_{\mathbf{w},b} \sum_{j=1}^{m} L(\mathbf{y}_j, \hat{\mathbf{y}}_j) + \frac{1}{2}\|\mathbf{w}\|^2 \text{ subject to } \min_j \left|\mathbf{w} \cdot \mathbf{x}_j + b\right| = 1$$

Noisy/Nonlinear Classification

- How to separate the following data by SVM?

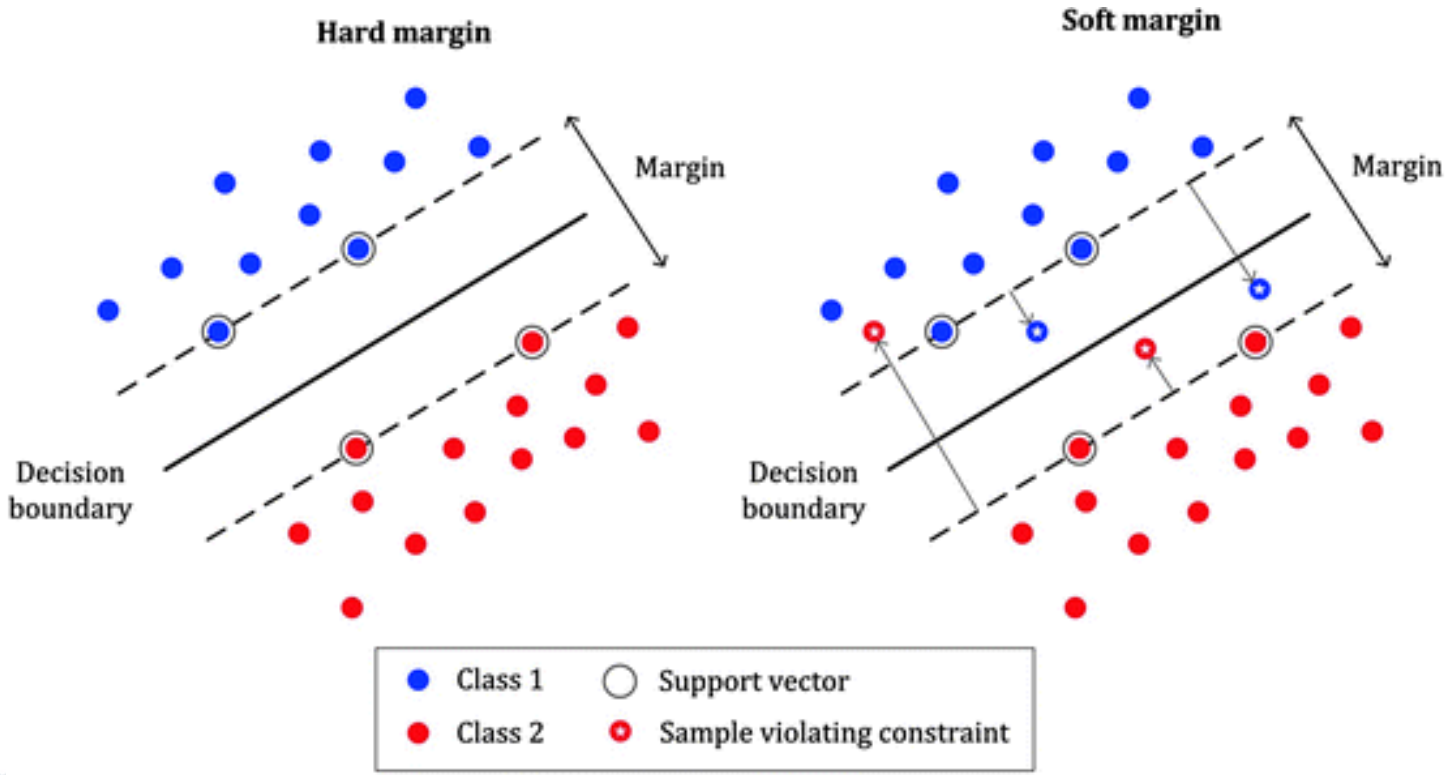


Noisy/Nonlinear Classification

- Two basic approaches:
  - Use a linear classifier, but allow some (penalized) errors
    - Soft margin, slack variables
  - Project data into higher dimensional space
    - Do linear classification there
    - Kernel functions

Soft margin

- Margin violation means choosing an hyperplane, which can allow some data points to stay in either in between the margin area or in the incorrect side of hyperplane.



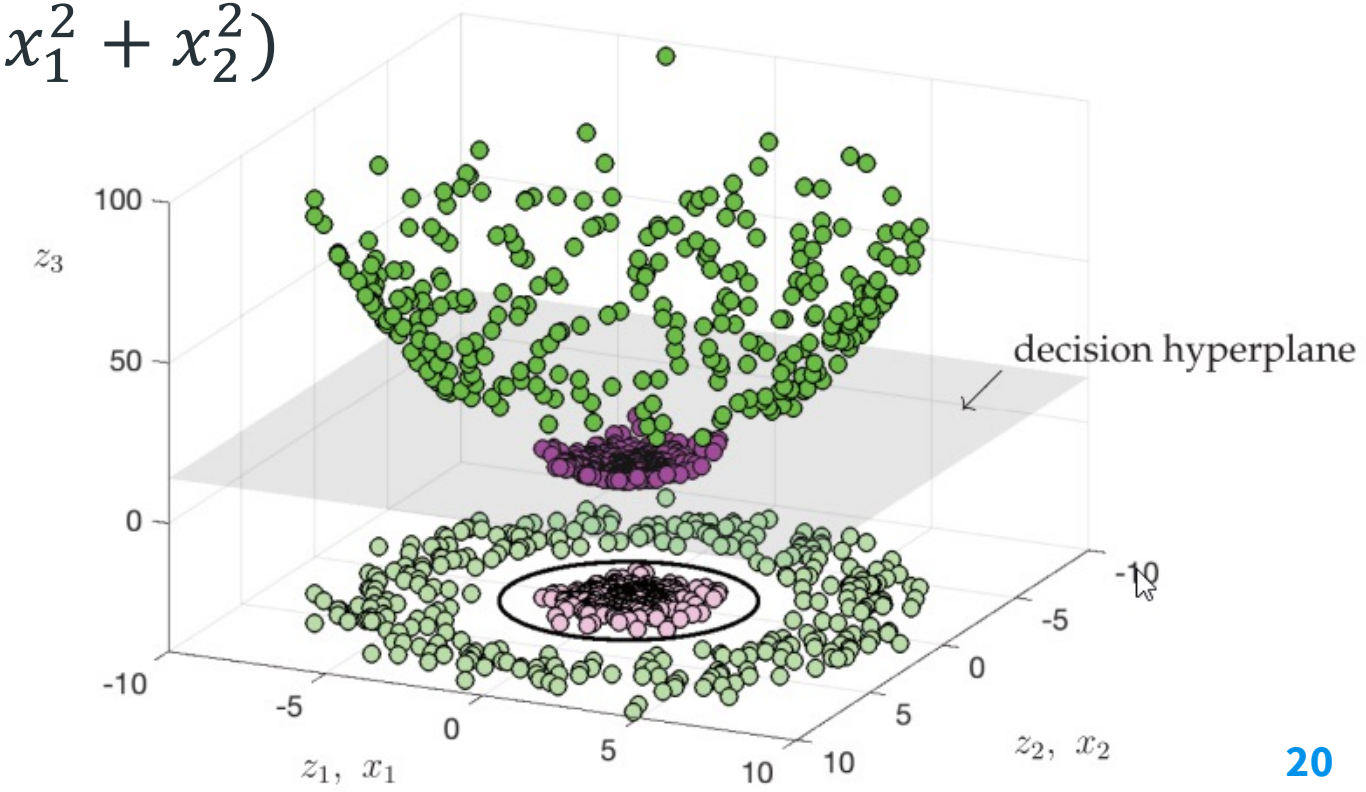
Map to higher-dimensional space

- Embedding the data in a higher dimensional space:

$$\mathbf{x} \rightarrow \Phi(\mathbf{x})$$

- For example:

$$(x_1, x_2) \rightarrow (z_1, z_2, z_3) := (x_1, x_2, x_1^2 + x_2^2)$$



Kernel trick

- Kernel function (similarity function) takes input vectors in the original space and return dot product of these vectors in the feature space (a real number).

$$K(\mathbf{x}, \mathbf{z}) = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{z})$$

Kernel trick

- The objective function only includes the dot product of the transformed feature vectors.

$$\mathbf{w} = \sum_{j=1}^{m} \alpha_j \, \Phi(\mathbf{x}_j)$$

$$f(\mathbf{x}) = \mathbf{w} \cdot \Phi(\mathbf{x}) + b = \sum_{j=1}^{m} \alpha_j \, \Phi(\mathbf{x}_j) \cdot \Phi(\mathbf{x}) = \sum_{j=1}^{m} \alpha_j \, K(\mathbf{x}_j, \mathbf{x})$$

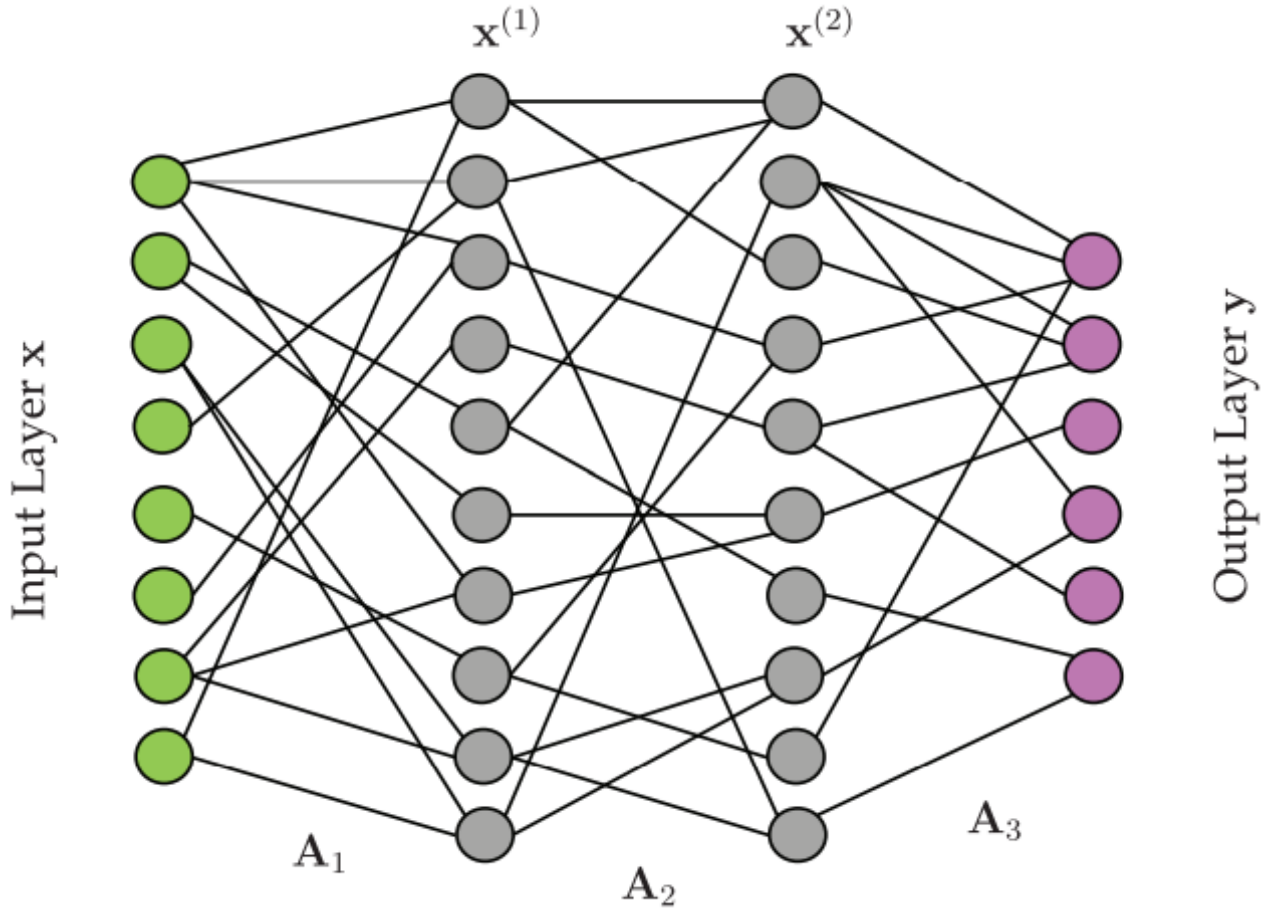
- Therefore, just substitute these dot product terms with the kernel function, and don't even use $\Phi(\mathbf{x})$.

Contents

- Data science and machine learning review
- Classification model
  - Support Vector Machine (SVM)
  - Neural network
- Clustering model

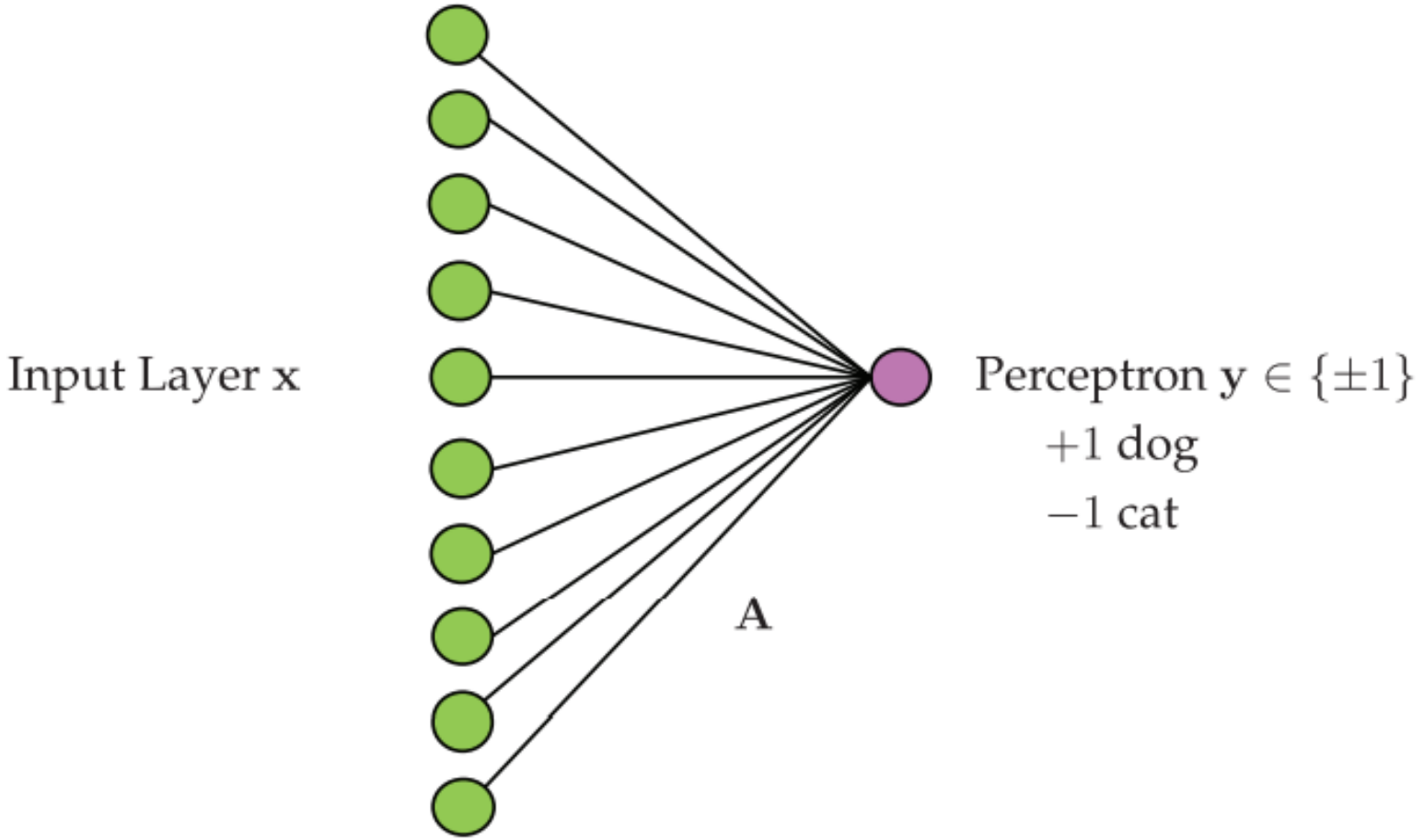
Generic architecture of a multi-layer NN

- For classification tasks, the goal of the NN is to map a set of input data to a classification.

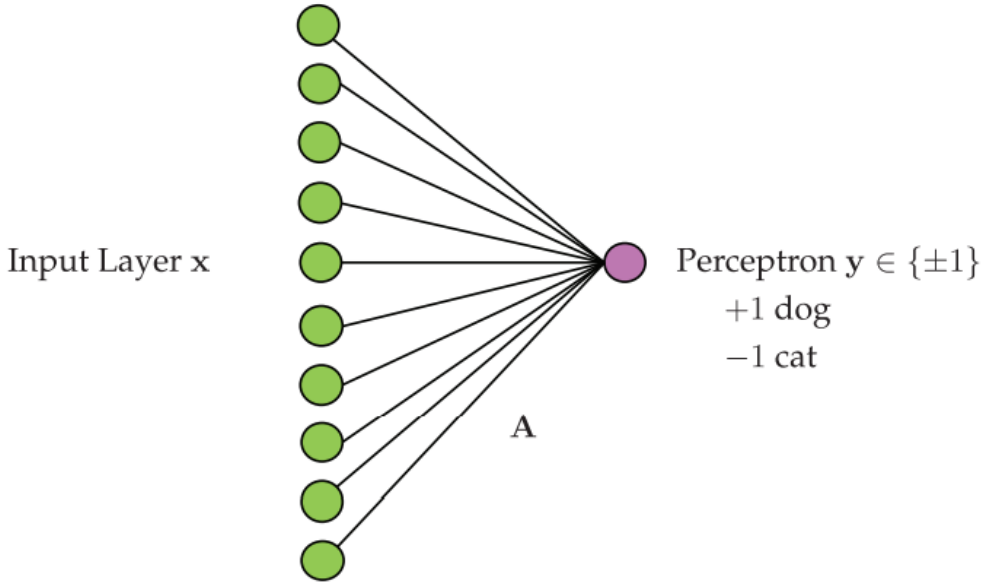


One-layer network

- First, consider a single layer network for binary classification:



One-layer network



- Hypothesis with linear mapping:

$$\mathbf{AX} = \mathbf{Y}$$

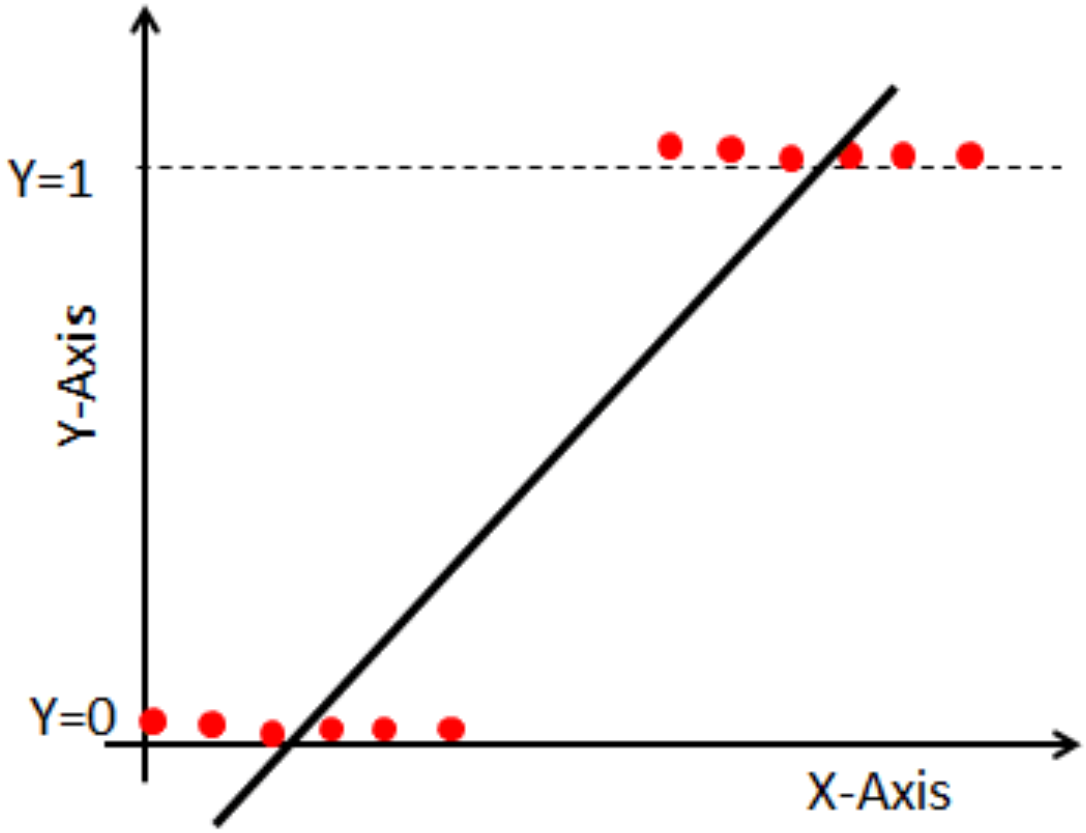
$$\rightarrow [a_1\ a_2\ \ldots a_n] \begin{bmatrix} | & | & | & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \ldots & \mathbf{x}_p \\ | & | & | & | \end{bmatrix} = [+1\ +1\ \ldots -1\ -1]$$

where each column of the matrix $\mathbf{X}$ is a dog or cat image ($\mathbf{x}_j \in \mathbb{R}^n$) and the columns of $\mathbf{Y}$ are its corresponding labels.

One-layer network

- The linear mapping:

$$\mathbf{AX} = \mathbf{Y}$$

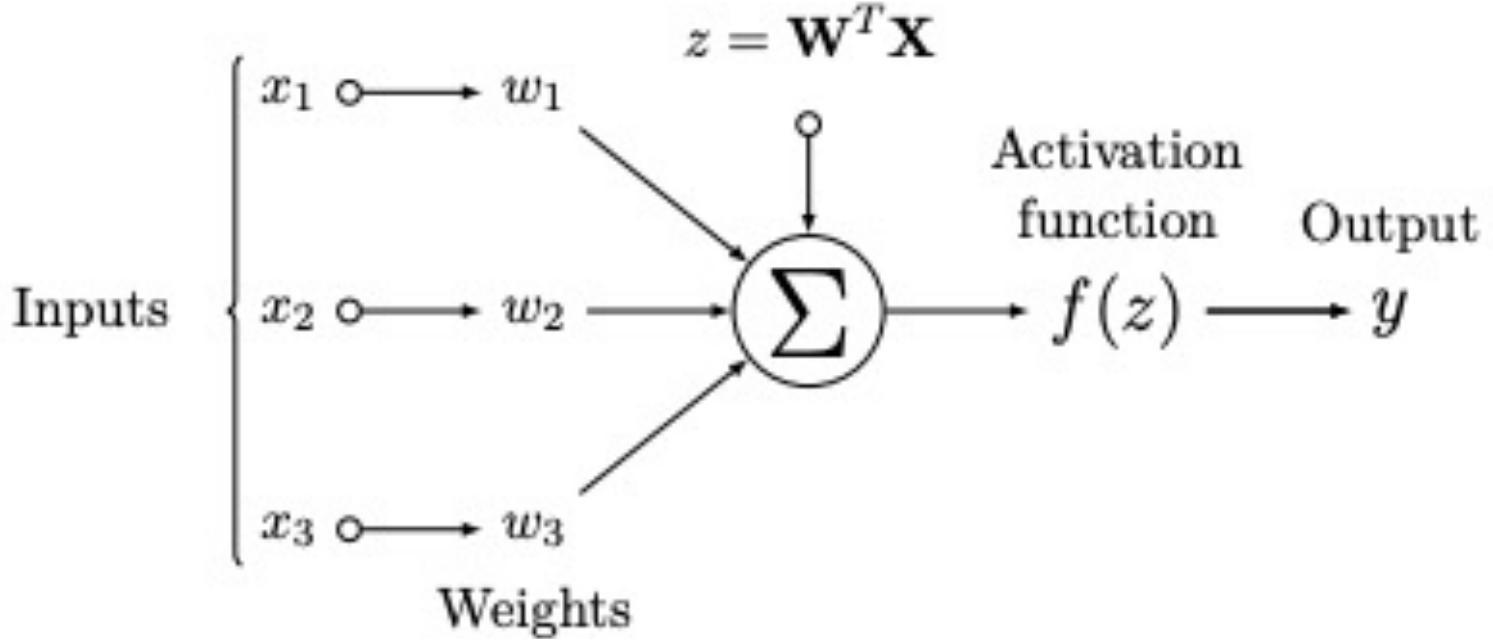


Nonlinear transformations

- Hypothesis with nonlinear mapping:

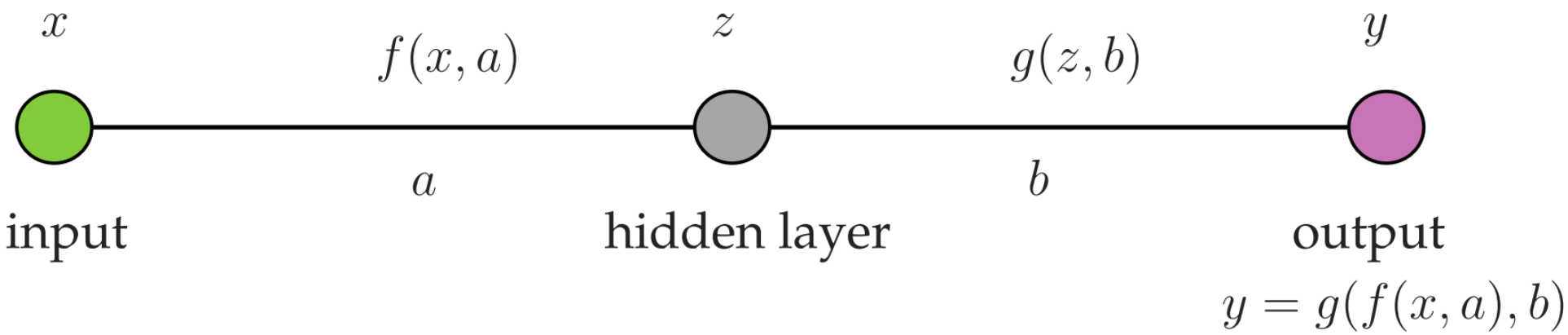
$$\mathbf{y} = f(\mathbf{A}, \mathbf{x})$$

where $f(\cdot)$ called an activation function (transfer function) in neural networks.



Neural network optimization

- The optimization of NN (determine the weights of the network) is done through the backpropagation process.
  - The process forces the optimization to backprogate error through the network relies on the chain rule of differentiation.
- For example, with simple neural network:



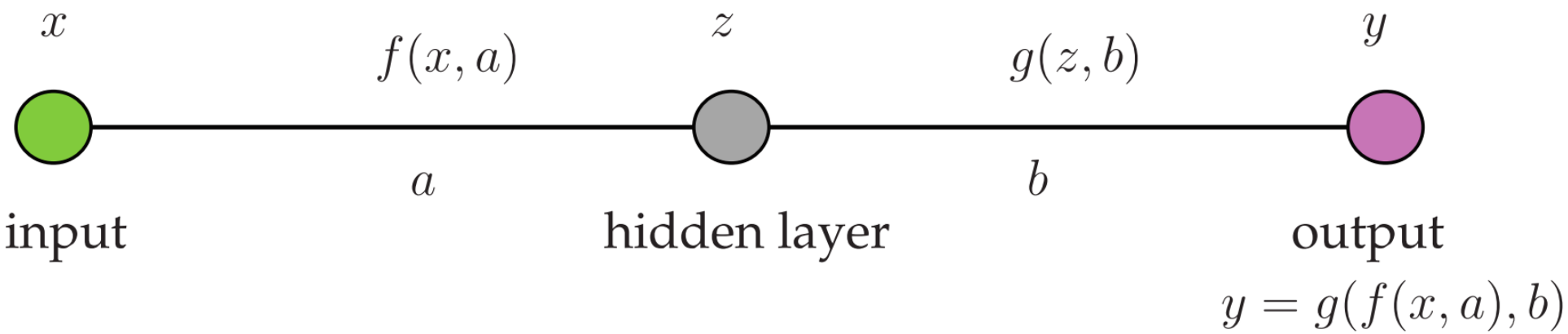
Neural network optimization

- The compositional structure is:

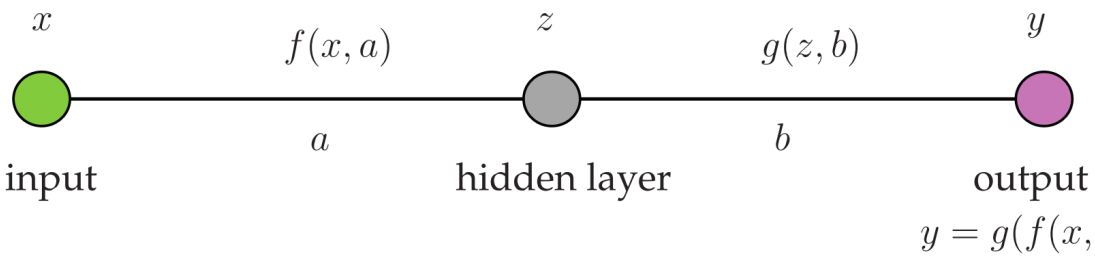
$$y = g(z, b) = g(f(x, a), b)$$

- The error function is:

$$E = \frac{1}{2}(y_0 - y)^2$$



Neural network optimization



- Partial derivative:

$$\frac{\partial E}{\partial a} = -(y_0 - y)\frac{\partial y}{\partial b} = -(y_0 - y)\frac{\partial y}{\partial z}\frac{\partial z}{\partial a} \qquad \text{(chain rule)}$$

$$\frac{\partial E}{\partial b} = -(y_0 - y)\frac{\partial y}{\partial b}$$

- Gradient descent:

$$a_{k+1} = a_k + \eta\frac{\partial E}{\partial a_k}$$

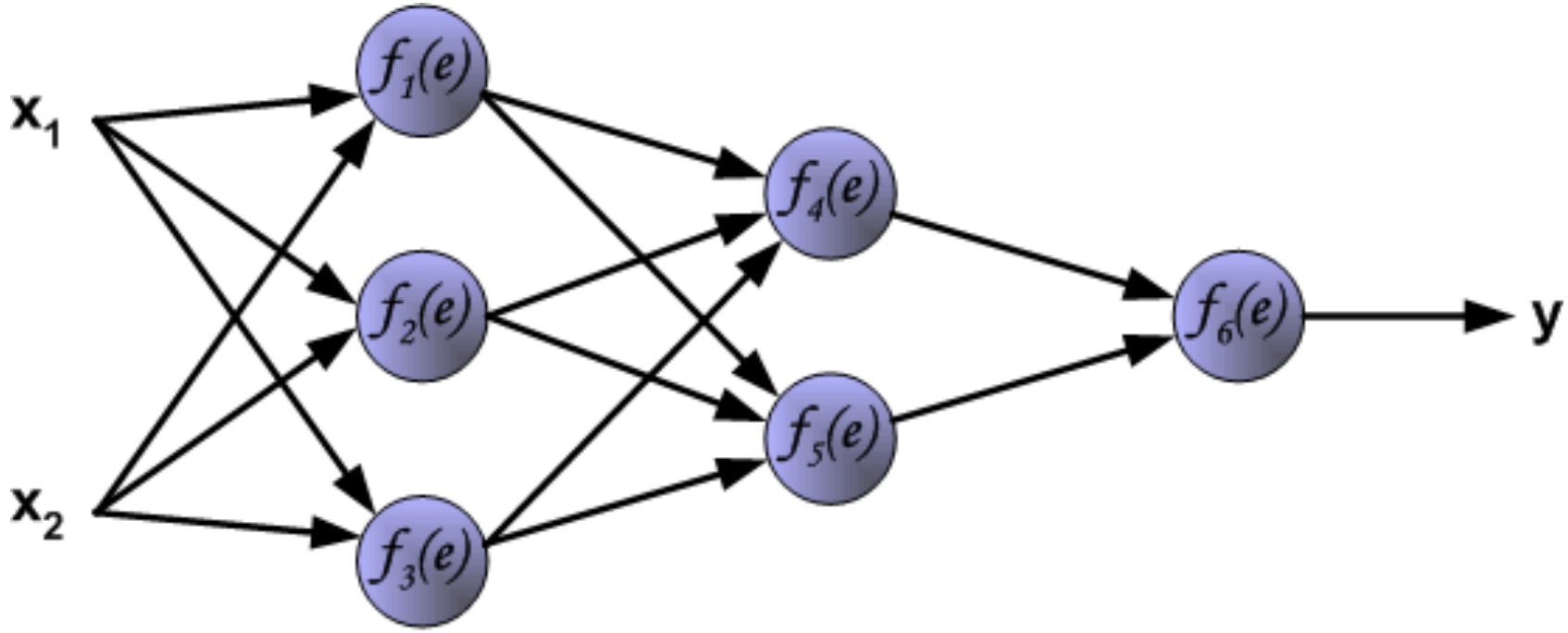
$$b_{k+1} = b_k + \eta\frac{\partial E}{\partial b_k}$$

Overall progress of neural network

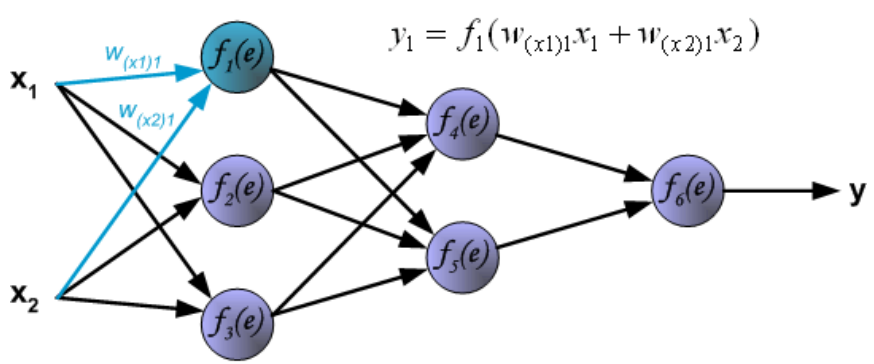
1. A NN is specified along with a labeled training set.
2. The initial weights of the network are set to random values.
3. The training data is run through the network to produce an output y, whose ideal ground-truth output is y_0.
4. The derivatives with respect to each network weight is then computed using backprop formulas.
5. For a given learning rate η, the network weights are updated as in gradient descent equation.
6. Return to step (3) and continue iterating until a maximum number of iterations is reached or convergence is achieved.

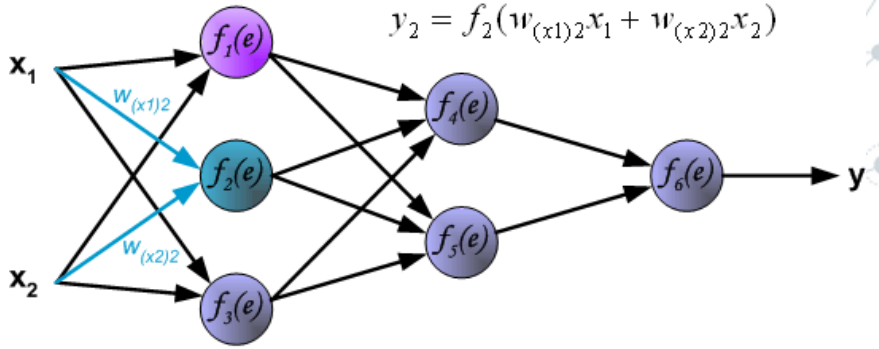
Overall progress of neural network

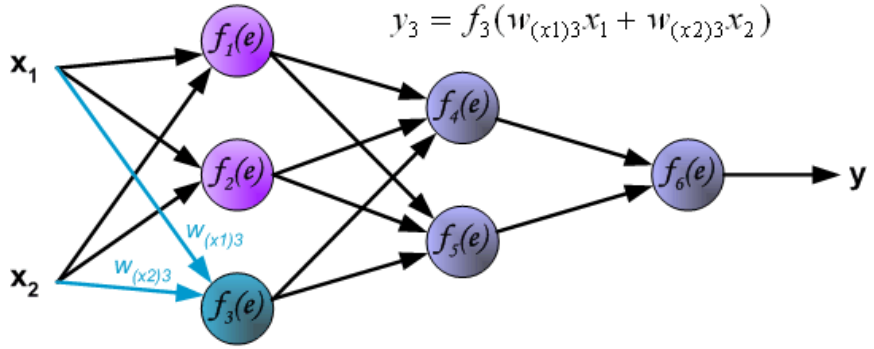
1. A NN is specified along with a labeled training set.
2. The initial weights of the network are set to random values.

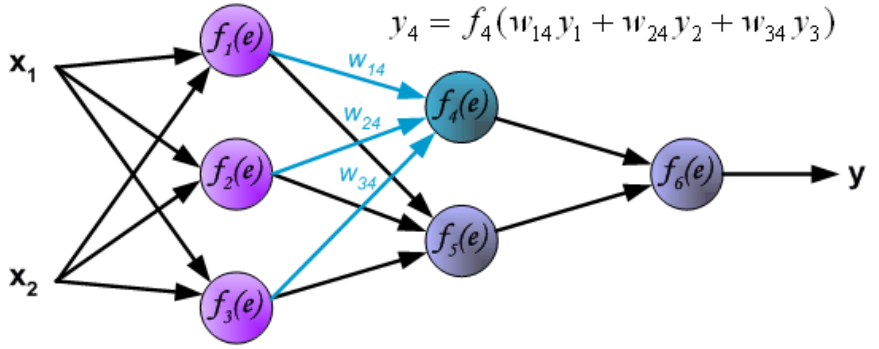


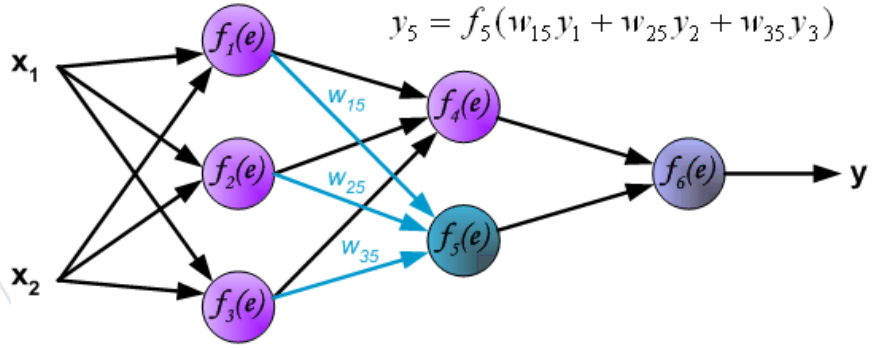
3. The training data is run through the network

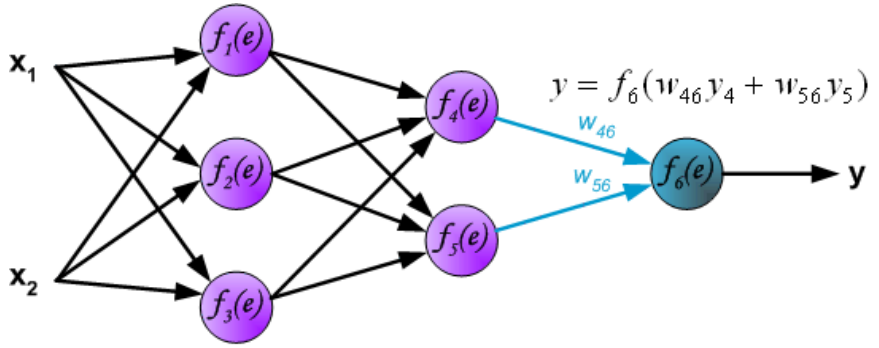




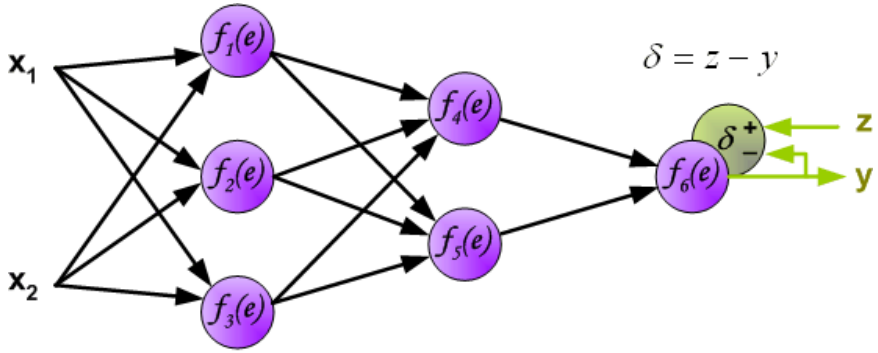


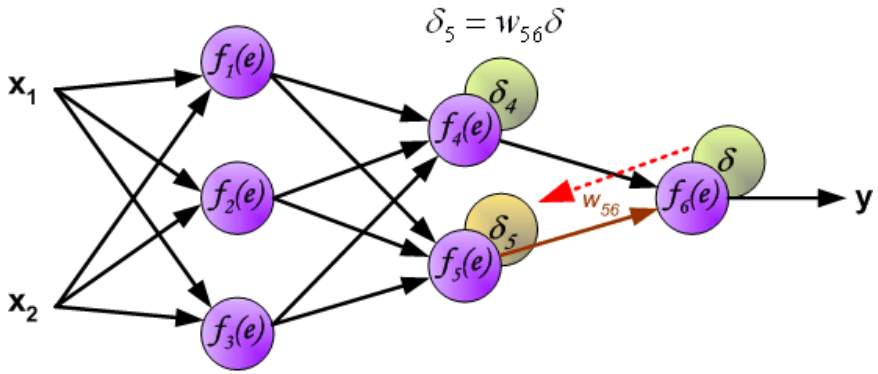


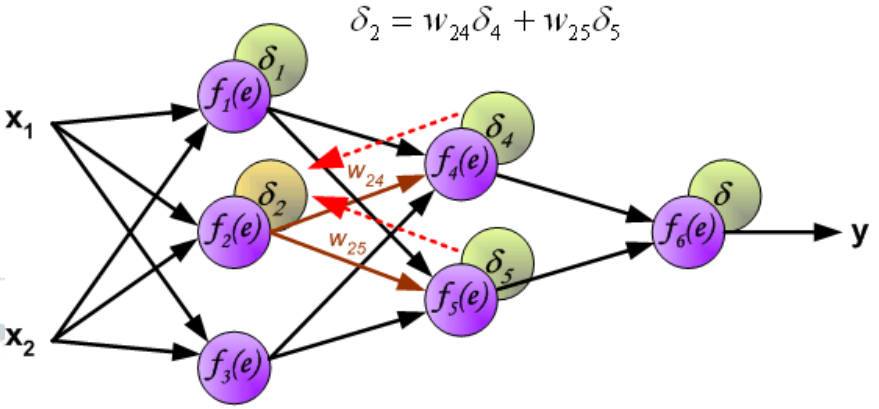


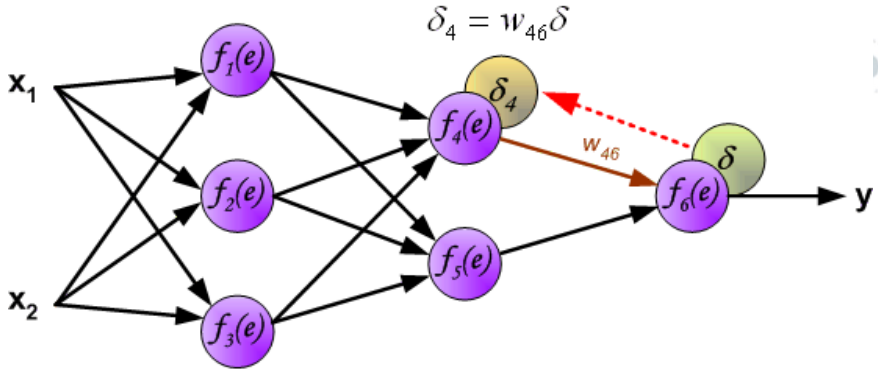


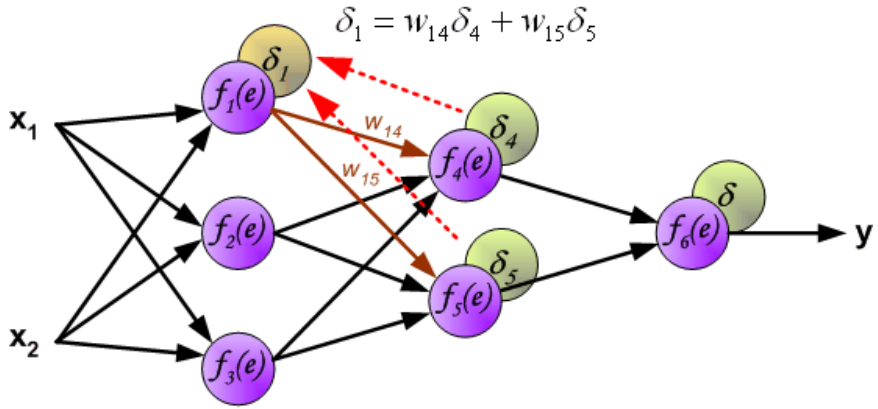
Backpropagate error

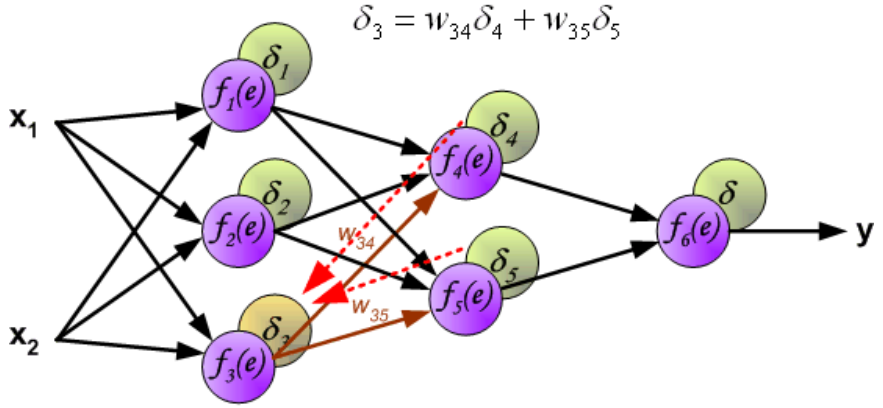








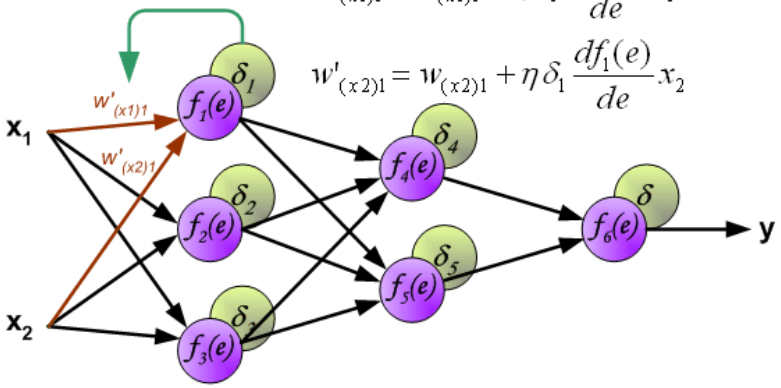




4. The derivatives with respect to each network weight is computed.
5. The network weights are updated as in gradient descent equation.

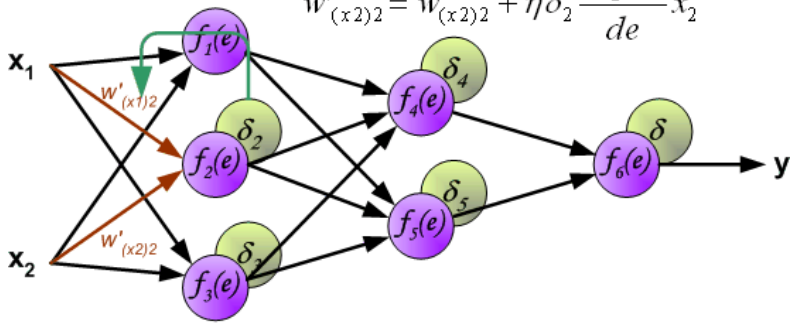
$$w'_{(x1)1} = w_{(x1)1} + \eta\delta_1 \frac{df_1(e)}{de} x_1$$

$$w'_{(x2)1} = w_{(x2)1} + \eta\delta_1 \frac{df_1(e)}{de} x_2$$



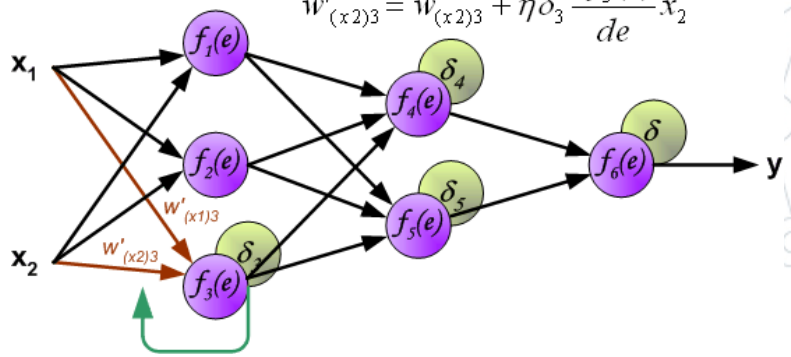
$$w'_{(x1)2} = w_{(x1)2} + \eta\delta_2 \frac{df_2(e)}{de} x_1$$

$$w'_{(x2)2} = w_{(x2)2} + \eta\delta_2 \frac{df_2(e)}{de} x_2$$



$$w'_{(x1)3} = w_{(x1)3} + \eta\delta_3 \frac{df_3(e)}{de} x_1$$

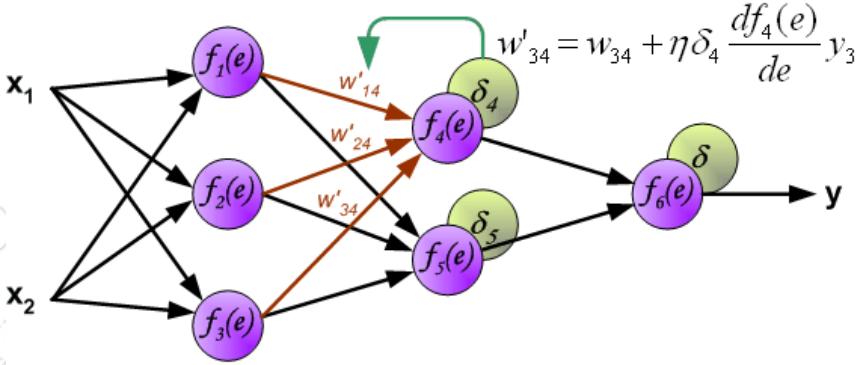
$$w'_{(x2)3} = w_{(x2)3} + \eta\delta_3 \frac{df_3(e)}{de} x_2$$



$$w'_{14} = w_{14} + \eta\delta_4 \frac{df_4(e)}{de} y_1$$

$$w'_{24} = w_{24} + \eta\delta_4 \frac{df_4(e)}{de} y_2$$

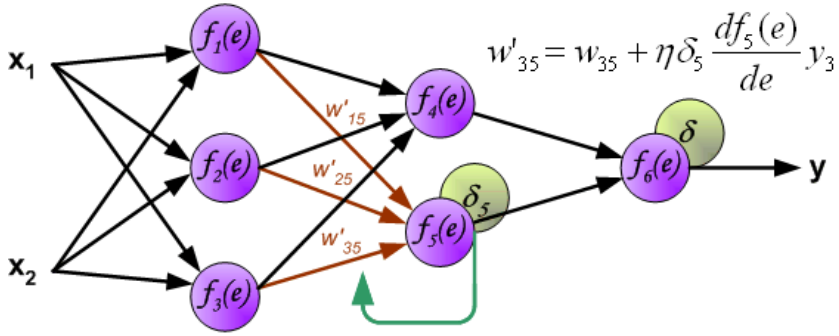
$$w'_{34} = w_{34} + \eta\delta_4 \frac{df_4(e)}{de} y_3$$



$$w'_{15} = w_{15} + \eta\delta_5 \frac{df_5(e)}{de} y_1$$

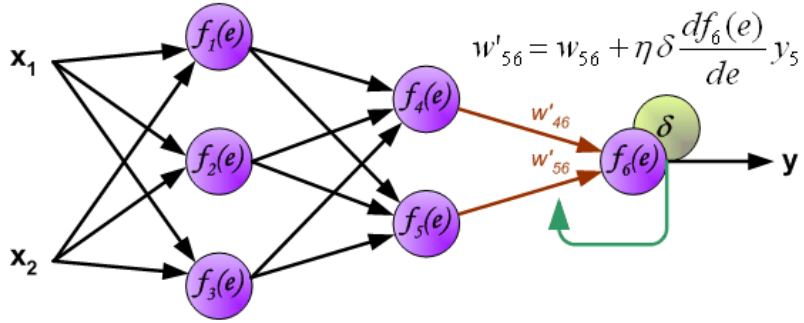
$$w'_{25} = w_{25} + \eta\delta_5 \frac{df_5(e)}{de} y_2$$

$$w'_{35} = w_{35} + \eta\delta_5 \frac{df_5(e)}{de} y_3$$



$$w'_{46} = w_{46} + \eta\delta \frac{df_6(e)}{de} y_4$$

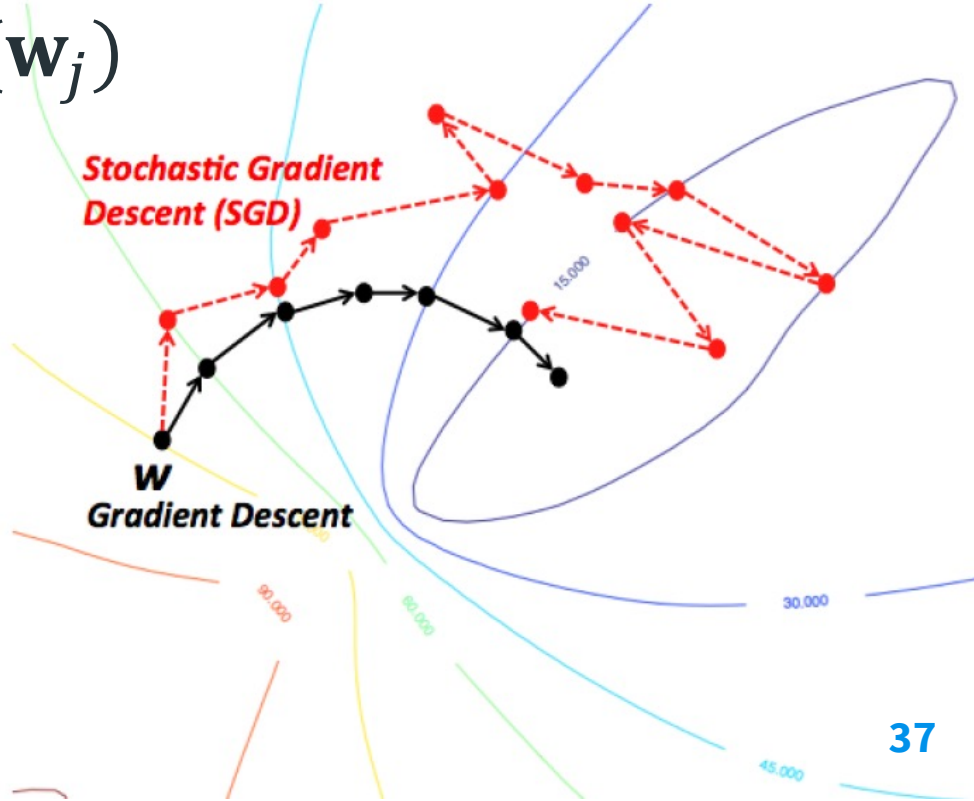
$$w'_{56} = w_{56} + \eta\delta \frac{df_6(e)}{de} y_5$$



Stochastic Gradient Descent

- Stochastic Gradient Descent (SGD): a single, randomly data point (k) is chosen to approximate the gradient at each step of the iteration.

$$\mathbf{w}_{j+1} = \mathbf{w}_j - \eta \nabla E_k(\mathbf{w}_j)$$

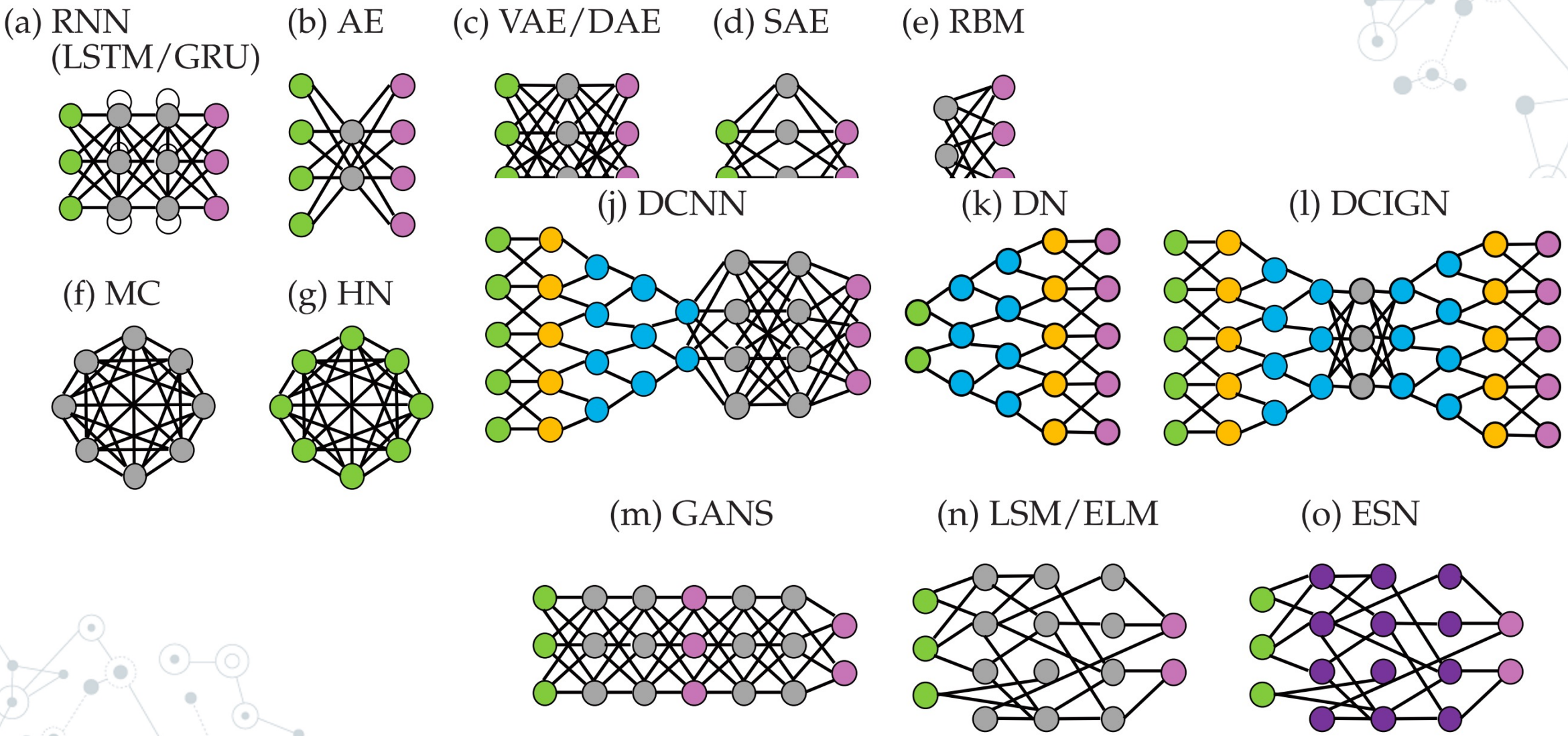


Batch gradient descent

- If instead of a single point, a subset of points (K) is used, then we have the following batch gradient descent algorithm.

$$\mathbf{w}_{j+1} = \mathbf{w}_j - \eta \nabla E_K(\mathbf{w}_j)$$

Deep Learning

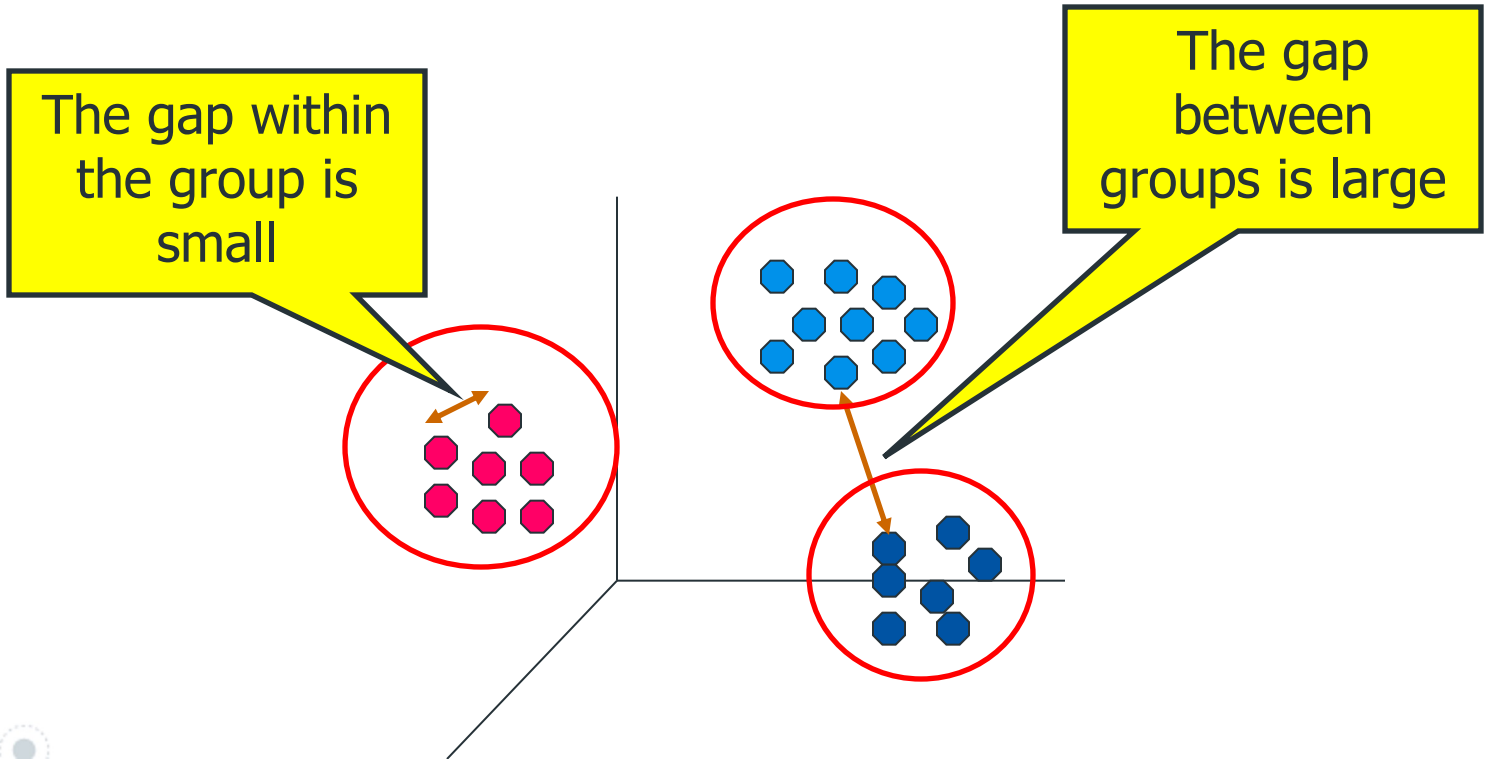


Contents

- Data science and machine learning review
- Classification model
- Clustering model

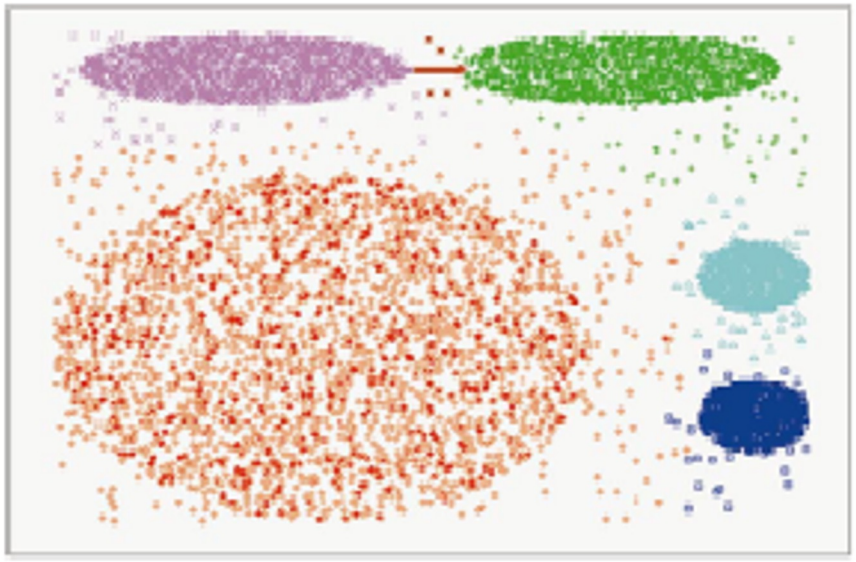
Clustering

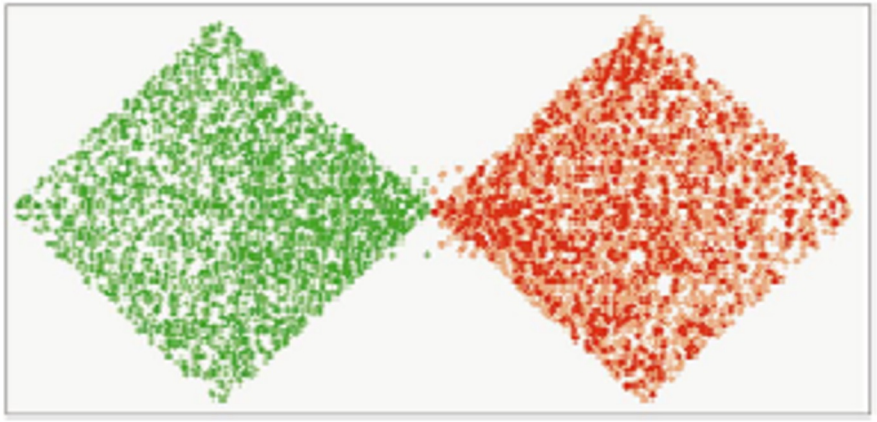
- Clustering is the process of grouping objects into clusters:
  - The same group objects are highly similar.
  - And very different from the subject in the rest of the groups.

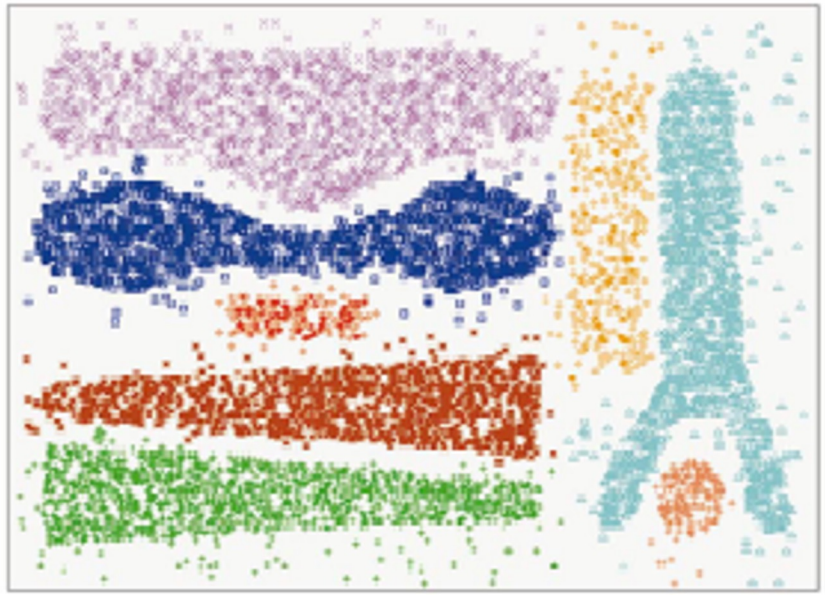


Clustering

- Clustering is an unsupervised learning because the label/class is not pre-defined
- Therefore, clustering is a type of learning by visual rather than learning by examples





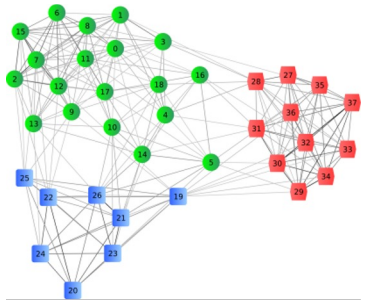


CHAMELEON

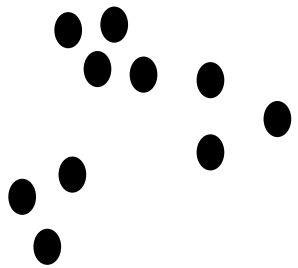
Some applications of clustering

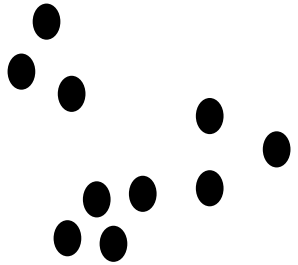
- Grouping of related documents for web browsing
- Groups of genes and proteins have the same function
- Group of stocks with the same volatility
- Groups of areas of the same land type in geography
- Identify homegroups by Home type, value, and geographic location
- Defining a Gaming object group

…

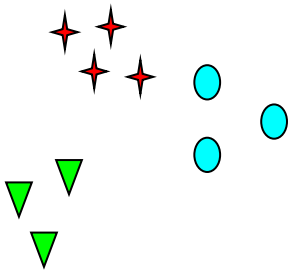


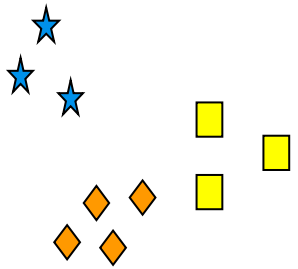
What is a group?



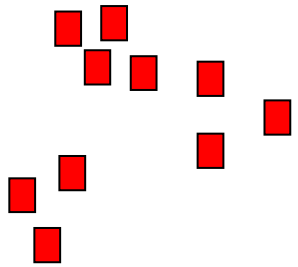


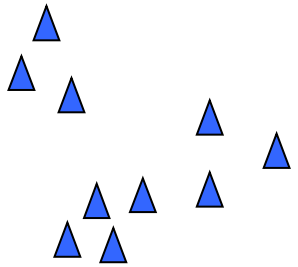
How many groups are there?



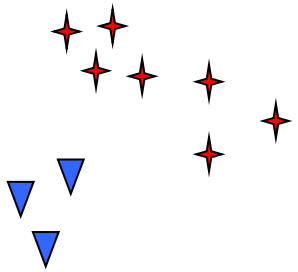


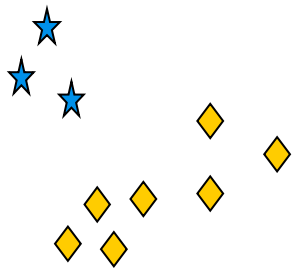
6 groups





2 groups





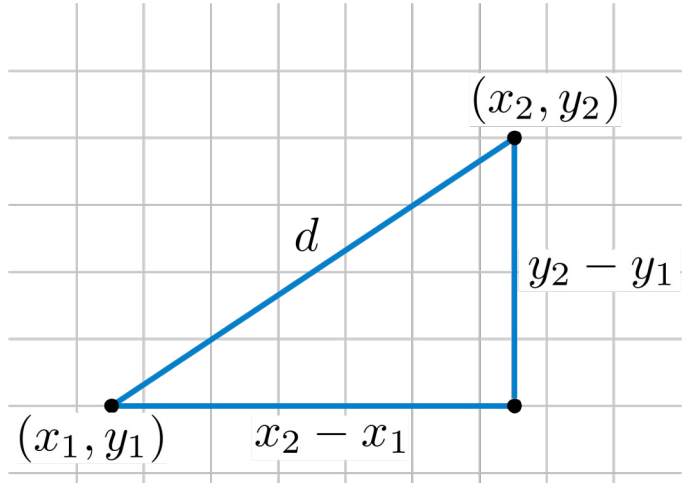
4 groups

A good Clustering?

- A good clustering method will have to create groups of high quality:
  - The same level in the High group.
  - Similar levels to other low-level groups.
- The quality of the clustering depends on:
  - Analog measurement
  - Its enforcement
  - Ability to discover some or all of the underlying patterns

Measure Similarity

- The distance used is mainly to measure the same or not similar level between the two objects.
  - Examples: Euclide gap, Cosin Gap, Minkowski, Mahattan...
- Distance functions differ in value ranges, types, ranks, and variable elements.
- The weights of the dependent variables on the application and the data implications.



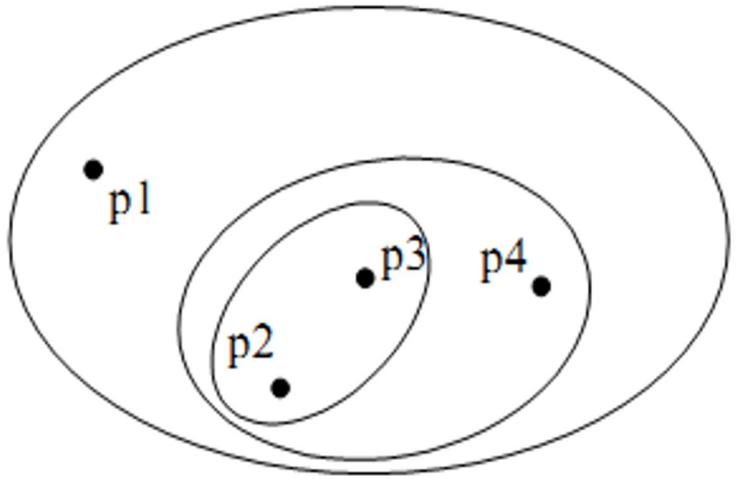
Some clustering methods

- Partitional clustering
  - Formation of partitions and evaluate them based on some criteria
  - Algorithms: K-Means, K-Medoids, CLARANS
- Hierarchical clustering
  - Create the division of Layers

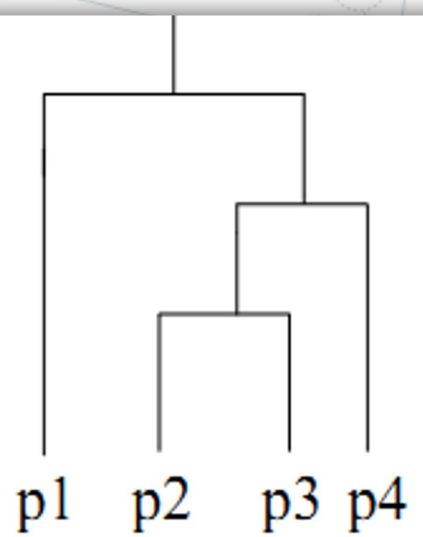
    Algorithms: Diana, Agnes, BIRCH, CAMELEON
- Density-based clustering
  - Based on the jaw and density function

    Algorithms: DBSACN, OPTICS, DenClue

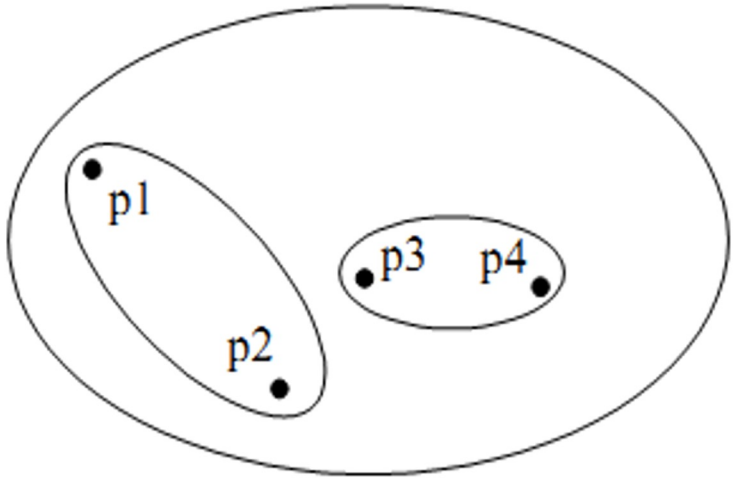
Examples



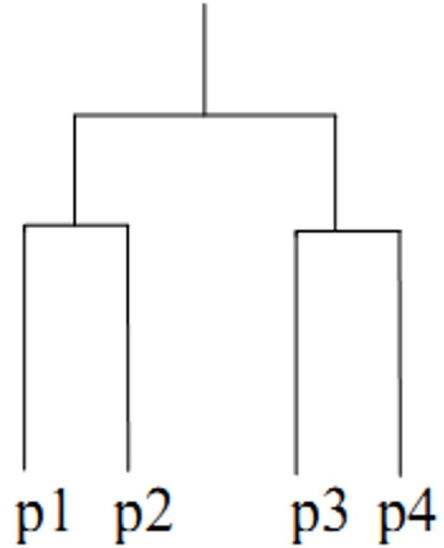
Traditional Hierarchical Clustering



Traditional Dendrogram



Non-traditional Hierarchical Clustering



Non-traditional Dendrogram

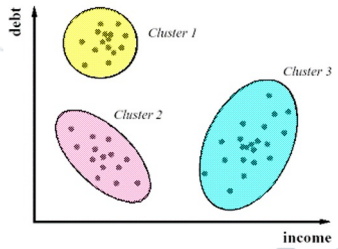
Some clustering methods (2/2)

- Grid-based methods
  - Based on multi-level particle structure

    Algorithms: STING, WaveCluster, CLIQUE
- Model-based methodology
- Frequent pattern-based methods
- Methods based on binding or user guidance
- Link-based method

…

Partitioning Clustering



- Partitioning Clustering Is the simplest and most foundational method of clustering methods
- Idea: The partition of a D database consists of n objects into k Groups so that it optimizes the criteria of the partition.
- Global optimization: Full expression of all groups
- Heuristic algorithm:
  - K-means: each group is performed by the center value of the group
  - K-Medoids, PAM: each group is performed by one of the groups ' objects

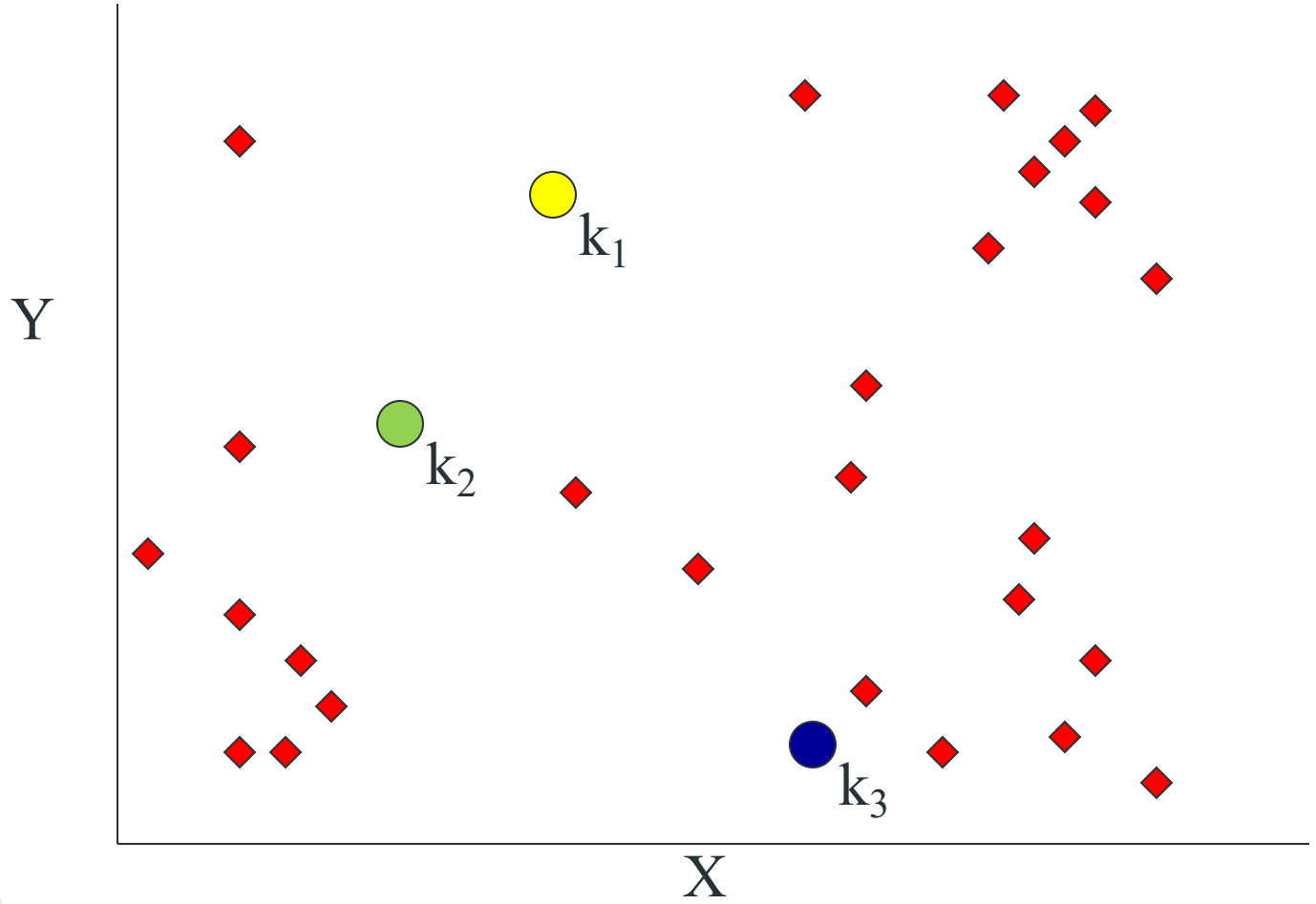
K-Means algorithm (1/2)

- Before number k, each group is performed by the group's centroid value.
  - S1: Select Random K objects as the center of the groups
  - S2: Assigns each remaining object to the closest group based on distance measurement such as Euclide, Cosin analogue, correlation,...
  - S3: Calculating the center value of each group based on newly joined objects.
  - S4: If the group Center has nothing to change or there are only a few points that change the group, stop, back to S2.

K-means example (1/4)

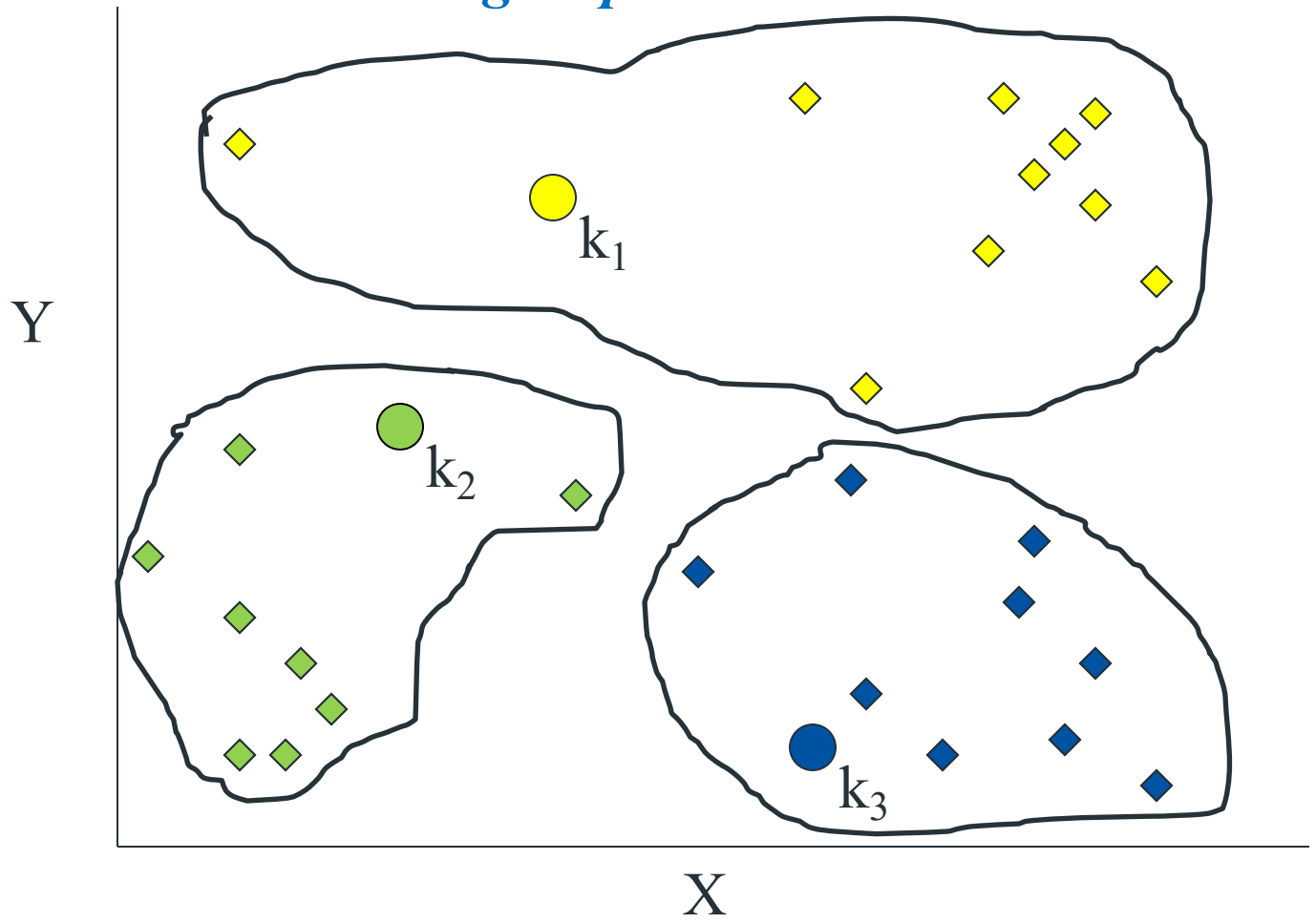
k = 3

S1: Select any of the three centers: k_1, k_2, k_3



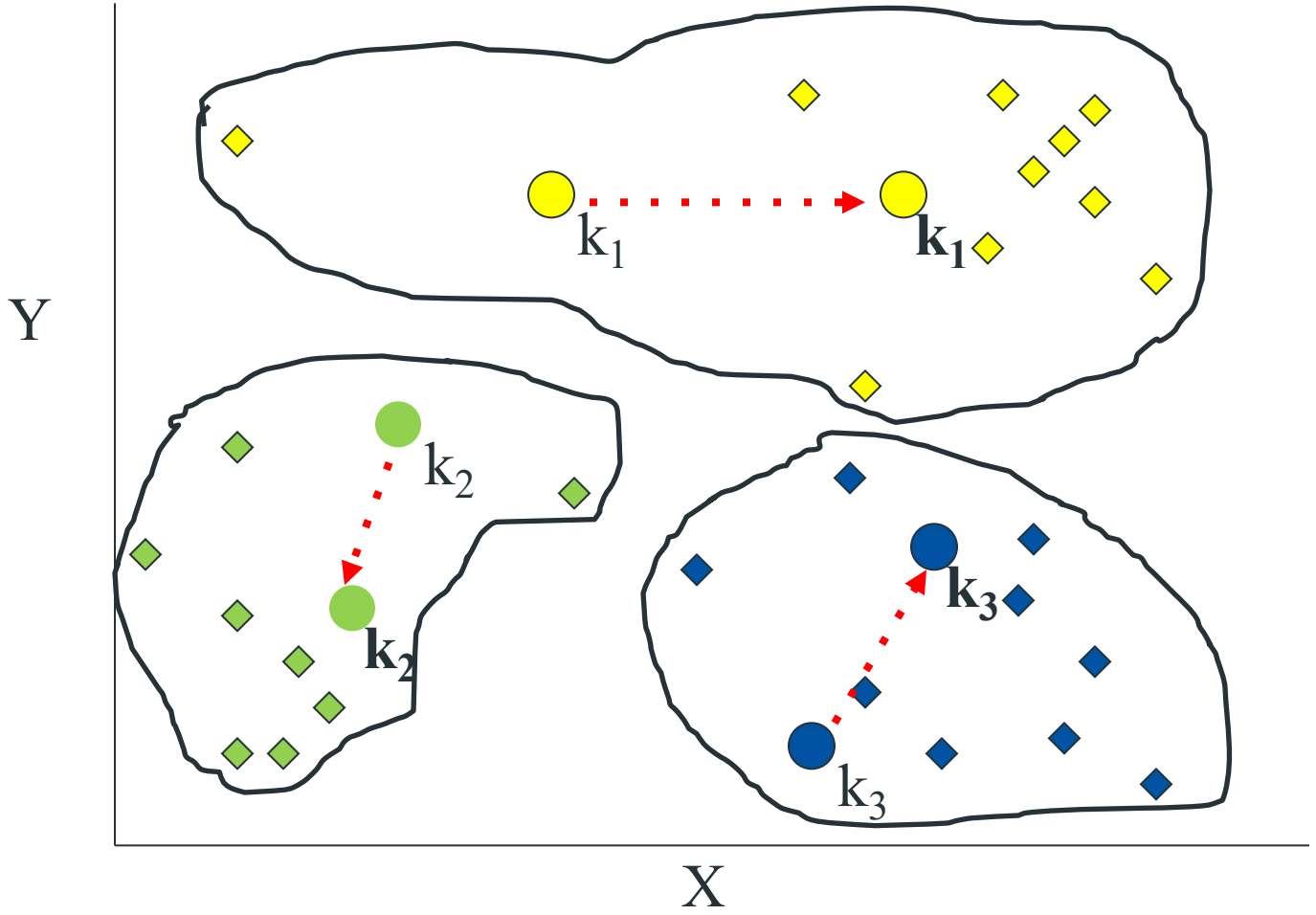
K-means example (2/4)

S2: Assign each point to a group with the closest group center



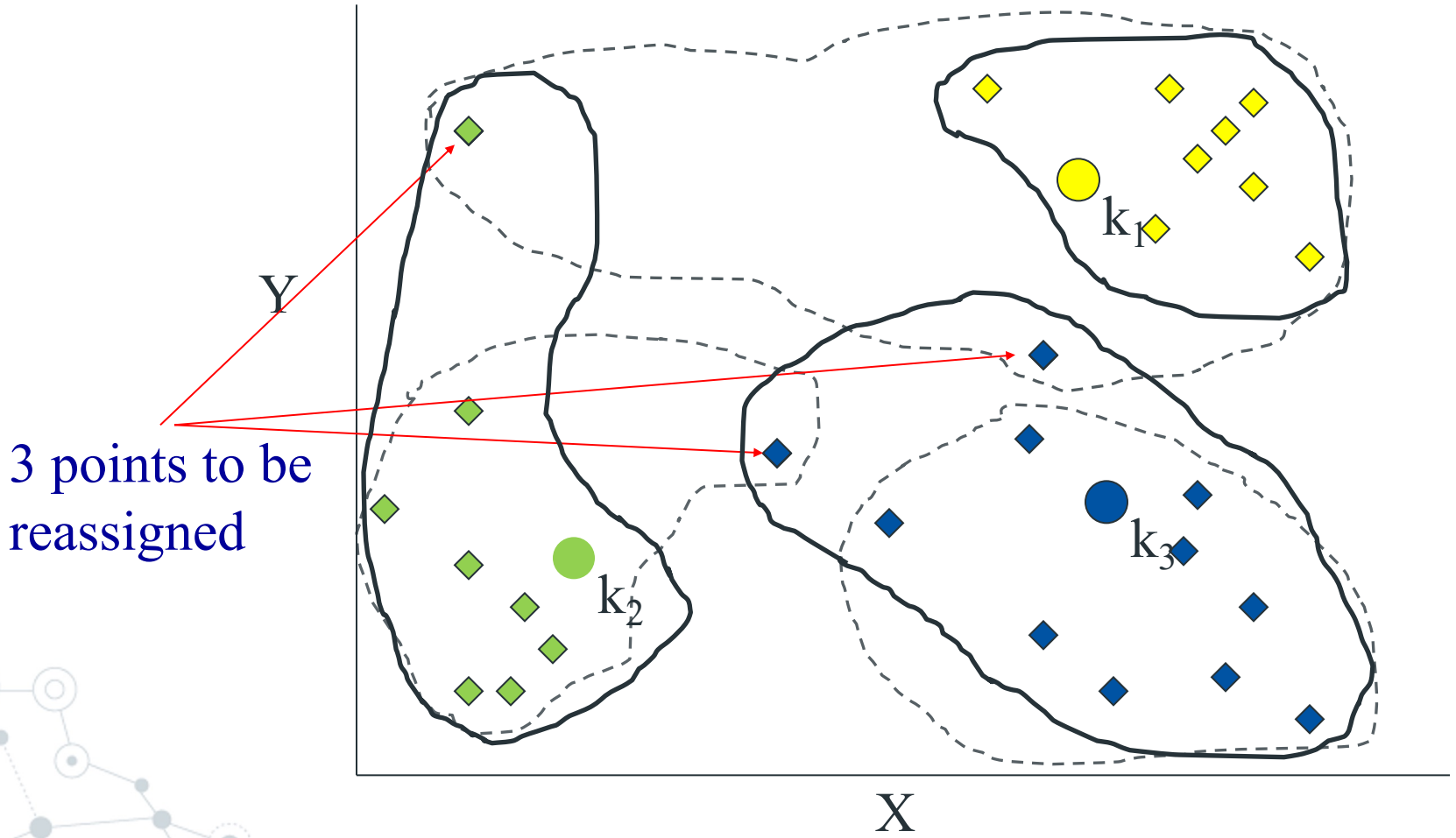
K-means example (3/4)

S3: Move a group center to the group's new average score



K-means example (4/4)

Repeat: Reassign points to close to new group centers...



k-means

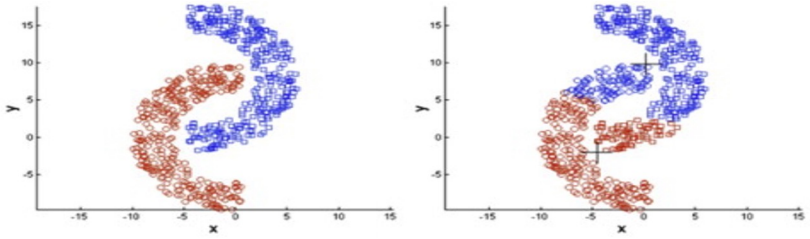
- Pros:
  - Simple, effective. Complexity O (TKN); T, K < < N.

    Achieve local optimization.
- Cons:
  - Only available to objects in a continuous n-dimensional space

    Need to determine the number K group before

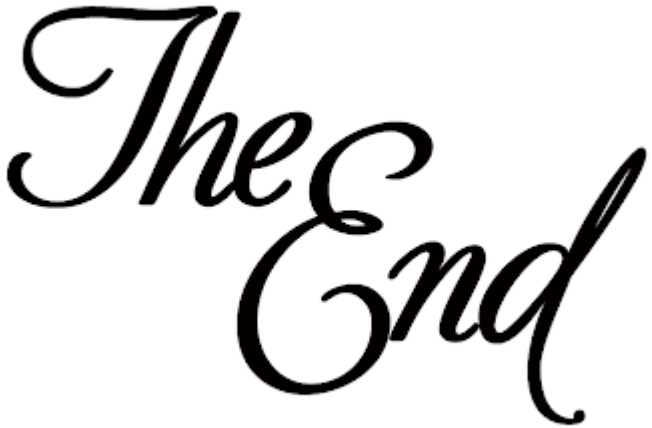
    Sensitivity to noise and personal data

    Not suitable for exploring groups with circles/bridges



Exercise 1

- Use the K-means algorithm and Euclide spacing to Clustering 8 samples into 3 groups:
  A1=(2,10), A2=(2,5), A3=(8,4), A4=(5,8), A5=(7,5), A6=(6,4), A7=(1,2), A8=(4,9).
  Distance matrix based on Euclide is given in the following slide.
  Assuming the initializing seed (center) is k1 = A1, K2 = A4 and K3 = A7. Run K-Means 1 time. Identify groups to be formed. Where is the new center of each group
- Draw in Space 10 x 10 The samples were bundled with the group over 1 run (draw frames) and the center of each group. How many K-means loops will converge (stop)? Illustrate the results at each loop.


The End