The National University of Computer and Emerging Sciences

Introduction to Machine Learnin

Machine Learning for Data Science

Dr. Akhtar Jamil

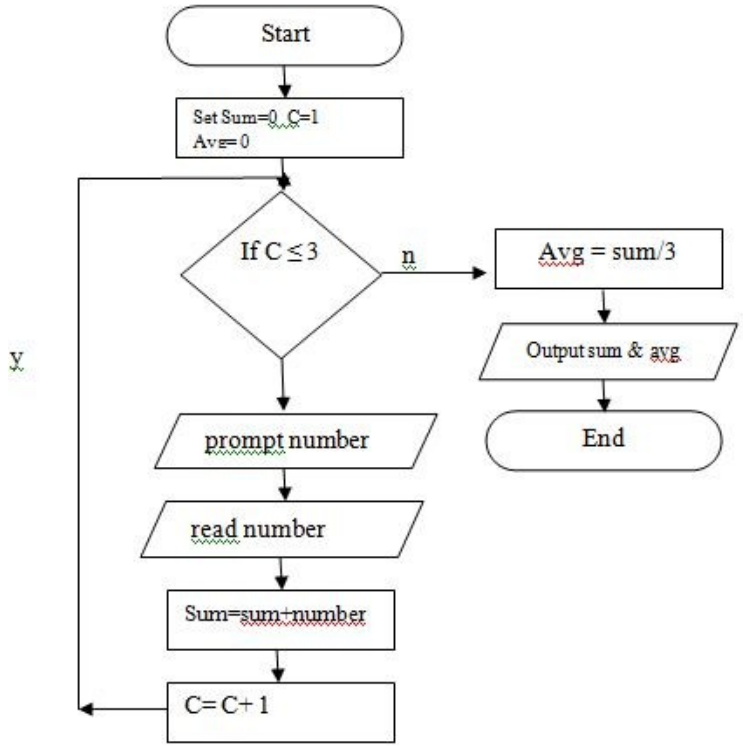
Department of Computer Science

Goals

- What is learning?
- What is machine learning?
- Types of machine learning
  - Classification vs Regression
  - Clustering vs Density Estimation

What is Learning?

- How can we solve a specific problem?
  - We write a program with a set of rules that are useful to solve the problem.
  - Example: Find average of three numbers



What is Learning?

- In many situations it is very difficult to specify those rules to solve a problem.
- For example, given a picture determine whether there is a cat in the image

What is Learning?

- Find face of a specific person?

What is Learning?

Benign vs Malignant tumor

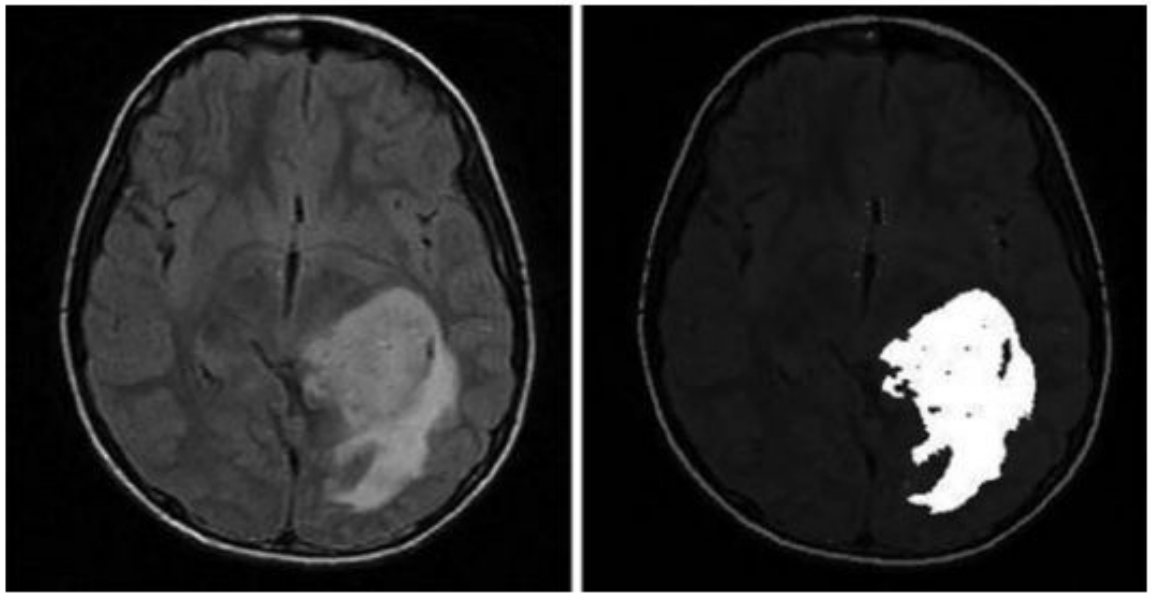
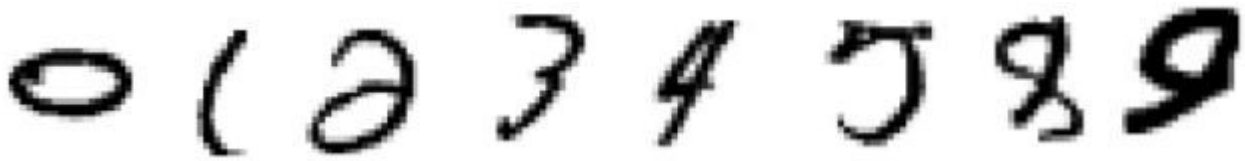


Figure 2. Gradient based genetic algorithm: (i) Original MRI image(ii) Brain tumor segmentation (KumarKole et al., 2012).

What is Learning?

- Any learning systems are not directly programmed using conditions to solve a problem
- Instead it should learn from examples (data)
- From trial-and-error experience trying to solve the problem



What is Machine Learning?

- Machine Learning is the science (and art) of programming computers so they can *learn from data*.

- *[Machine Learning is the] field of study that gi computers the ability to learn without being ex programmed.*
  - Arthur Samuel, 1959

What is Machine Learning?

- Machine learning can be defined as computational methods using experience to improve performance or to make accurate predictions.
- *Experience* refers to the past information.

Mohri et al

What is Machine Learning?

- Definition: “A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E”

Tom M. Mitchel

A checkers learning problem

- **Task T:** playing checkers
- **Performance measure P:** percent of games won against opponents
- **Training experience E:** playing practice games

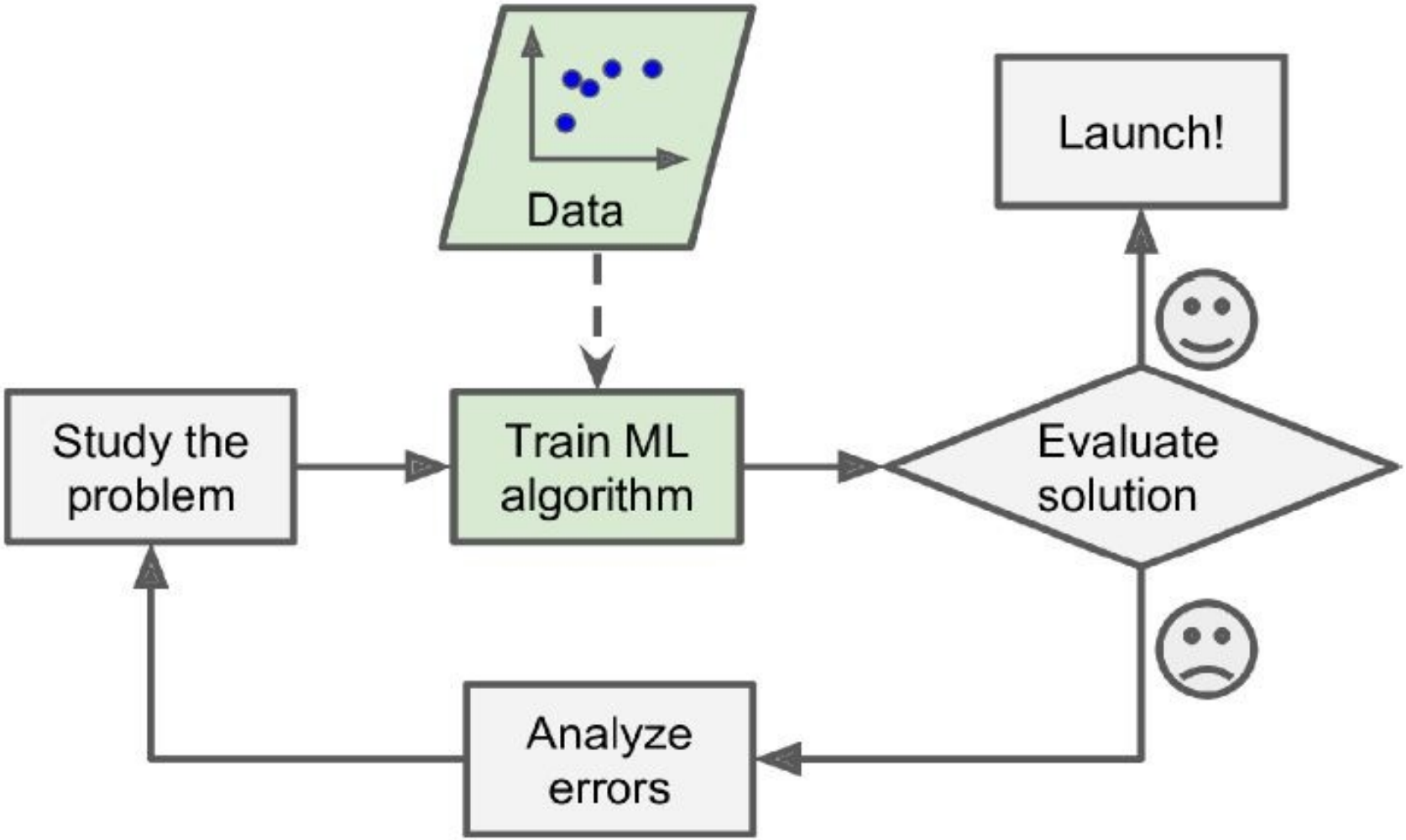
Spam Tagging Problem

- Your spam filter is a Machine Learning program
  - Binary Classification Problem: spam emails or nonspam
- To train a machine learning model, examples of emails that are spam and nonspam should be presented to the model
  - Usually flagged by users
- The examples that the model uses to learn are called the training set.
  - Training instance (or sample).

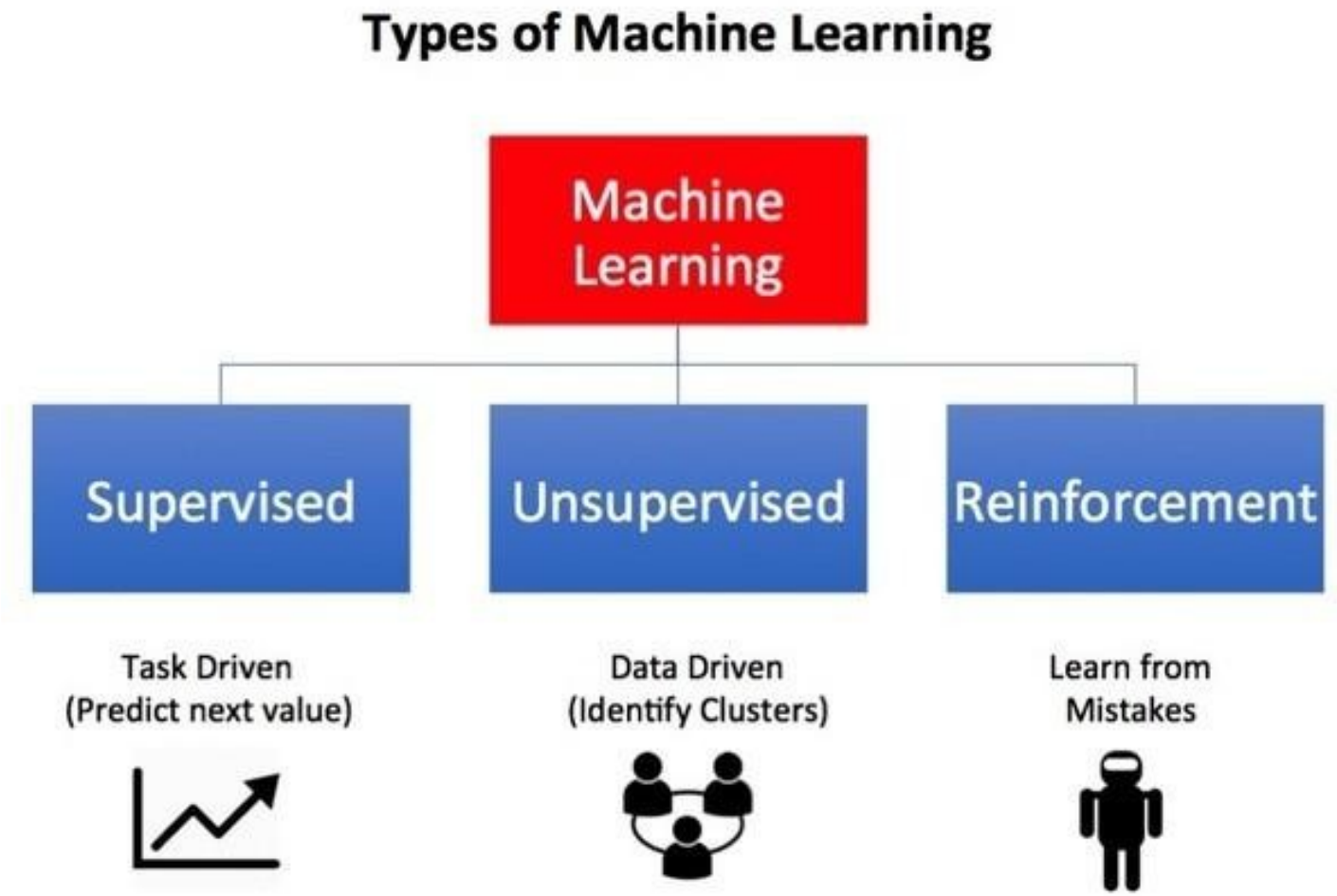
Spam Tagging Problem

- For Spam classification:
  - The task T is to flag spam for new emails
  - The experience *E* is the *training data*
  - The performance measure *P* needs to be defined;
    - Percentage of correctly classified emails (*accuracy)*

General Framework for ML



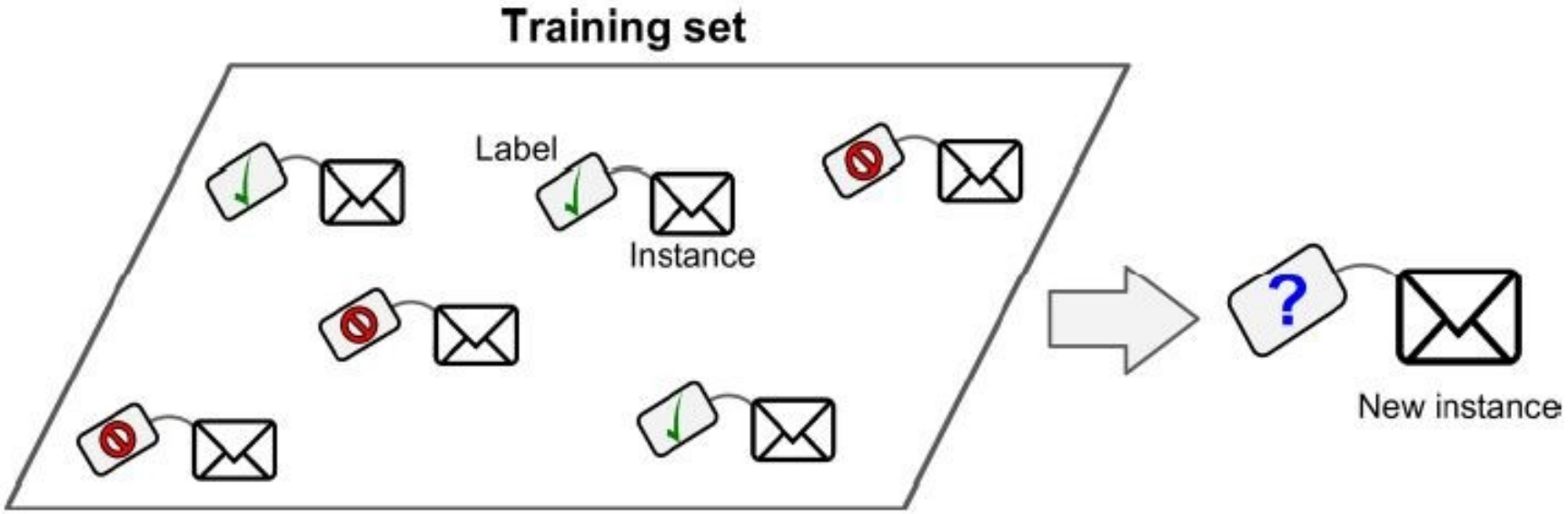
Types of Machine Learning…



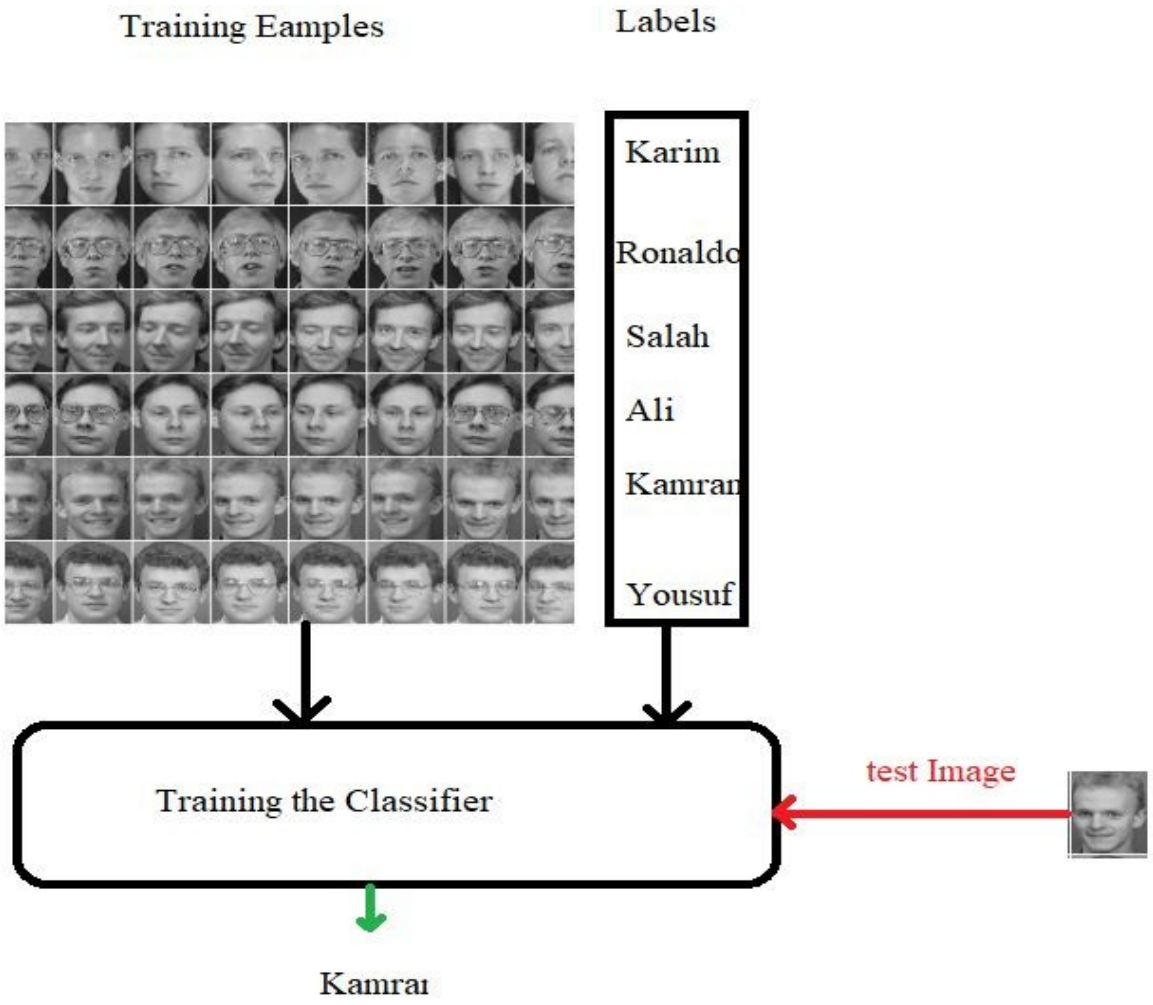
Supervised learning

- For supervised learning, we provide both data and labels for training the algorithm.
- The algorithms learns from the data and labels
- After training, we can pass test samples to check if the algorithm learned the data or not
- Most popular in ML community

Supervised learning: Example



Supervised learning: Example

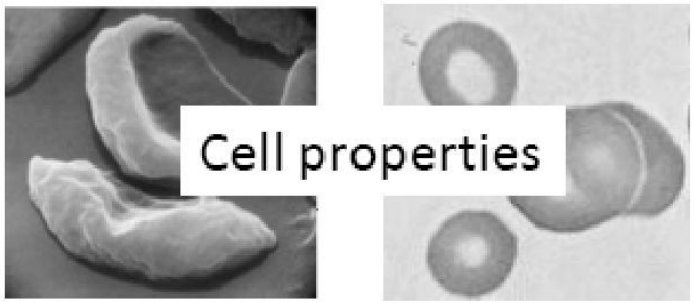


Supervised Learning

Feature Space $\mathcal{X}$

Label Space $\mathcal{Y}$

"Sports"
"News"
"Science"
...



"Anemic cell"
"Healthy cell"

Discrete Labels

Supervised learning

Data: $x = \{x_1, x_2, .., x_n\}$ ***n* examples**

$d_i = < \mathbf{x}_i, y_i >$

$\mathbf{x}_i$ is input vector, and y is desired output (given by a teacher)

Objective: learn the mapping $f : X \to Y$

s.t. $y_i \approx f(x_i)$ for all $i = 1, .., n$

Supervised learning

Two types of problems:

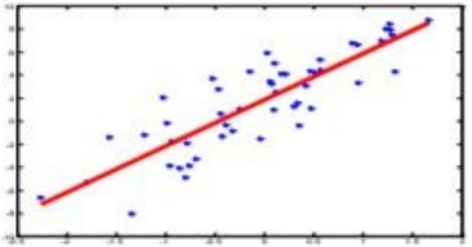
- **Regression:** X discrete or continuous → Y is **continuous**
- **Classification:** X discrete or continuous → Y is **discrete**

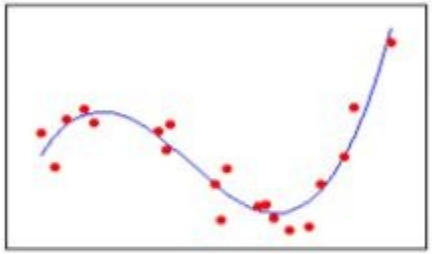
Supervised learning

- **Regression:** Y is **continuous**

Debt/equity
Earnings
Future product orders

→ Stock price





Data:

Debt/equity	Earnings	Future prod orders	Stock price
20	115	20	123.45
18	120	31	140.56
….			

Supervised learning

- **Classification:** Y is **discrete**

Label "3"

Handwritten digit (array of 0,1s)

Data:

image	digit
	3
	7
	5

Supervised learning

- Can regression algorithms be used for classification and vice versa?
  - Yes, some algorithms can be used.
- Logistic Regression is commonly used for classification
  - Predicts probability belonging to a class

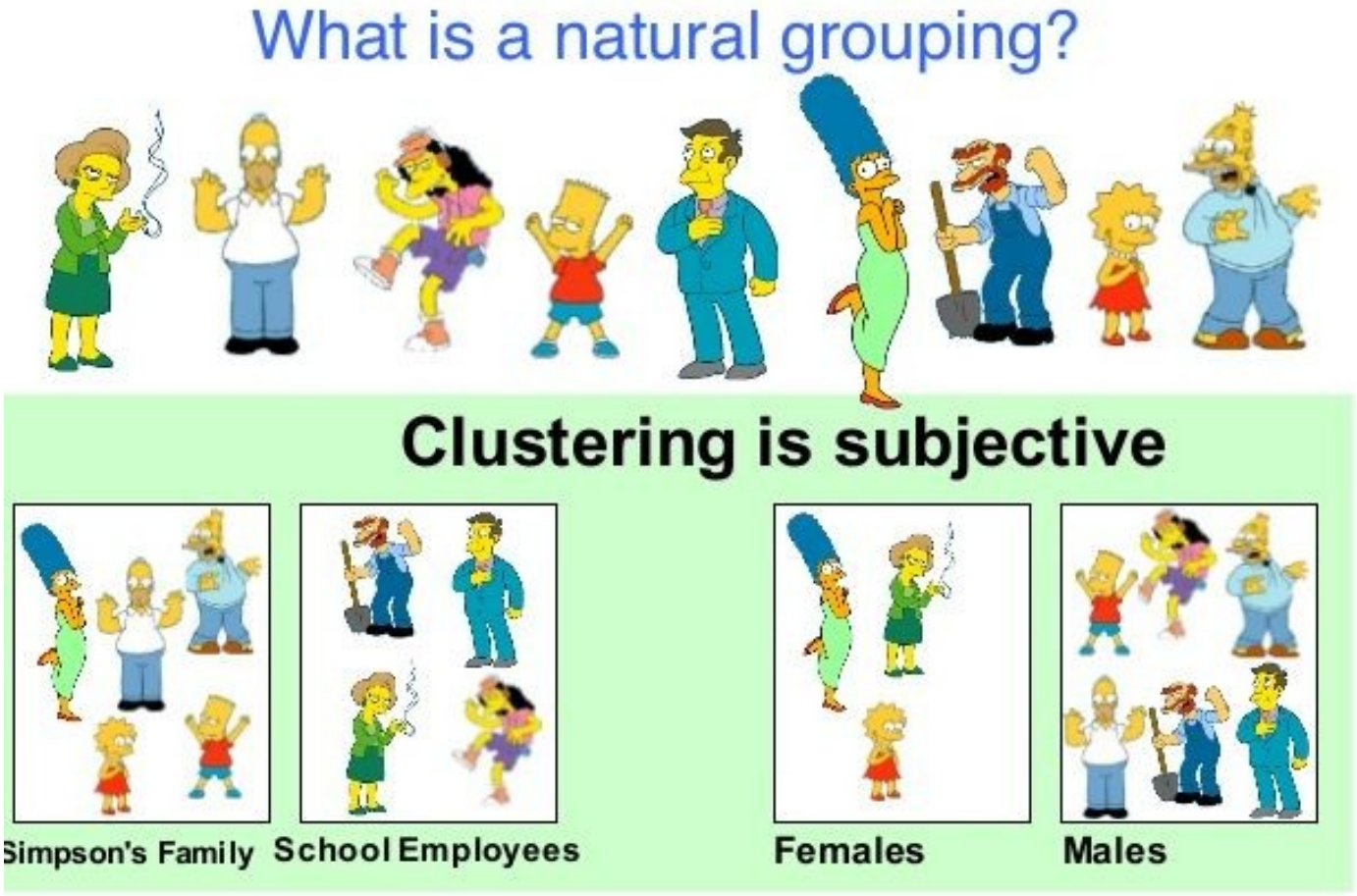
Supervised learning

- Some widely used supervised ML algorithms:
  - Linear Regression
  - Logistic Regression
  - Support Vector Machines (SVMs)
  - Decision Trees and Random Forests
  - Neural networks
  - k-Nearest Neighbors

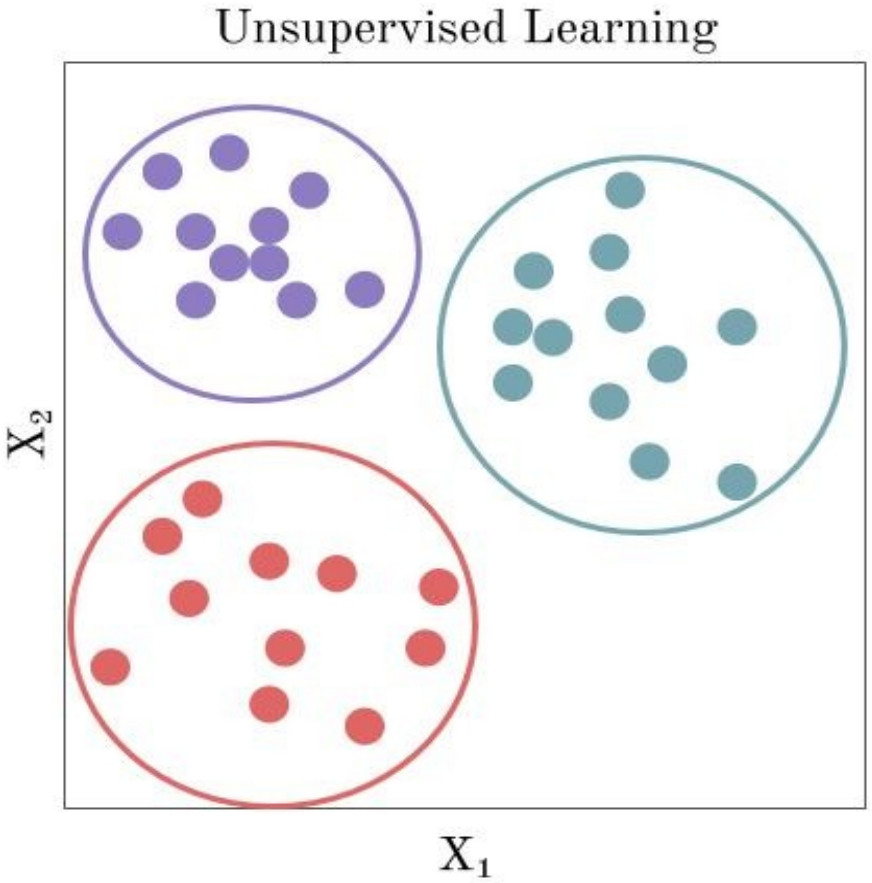
Unsupervised learning

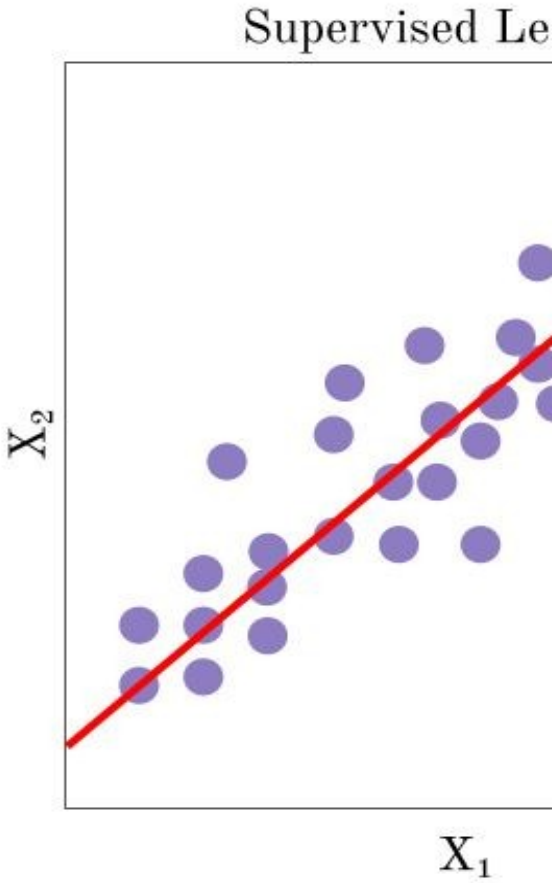
- For unsupervised learning, we provide data but NOT labels for training the algorithm
- The system tries to learn without a teacher.
- Learns relations among data by itself
- Then put the data into different groups/clusters

Unsupervised Learning



Unsupervised Learning





Unsupervised Learning

- Some widely used unsupervised learning algorithms:
  - K-Means
  - Principal Component Analysis (PCA)
  - Apriori
  - Hierarchical Cluster Analysis (HCA)
  - One-class SVM

Usage of Unsupervised Learning

- Data visualization
- Dimensionality reduction
- Clustering
- Anomaly detection

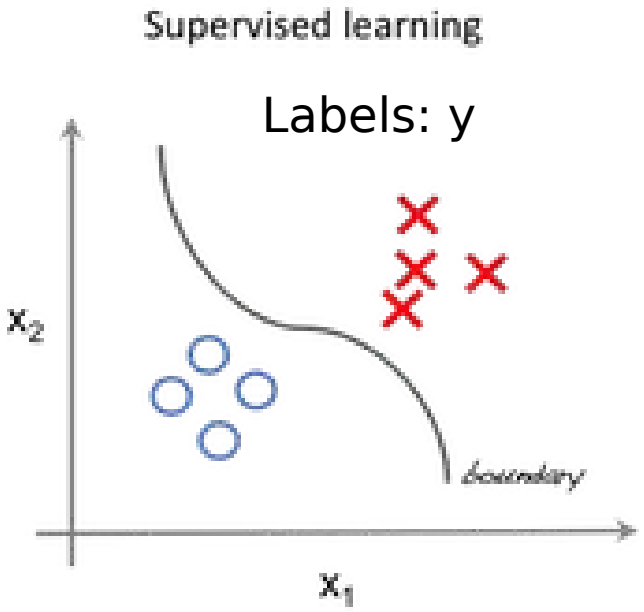
Unsupervised Learning

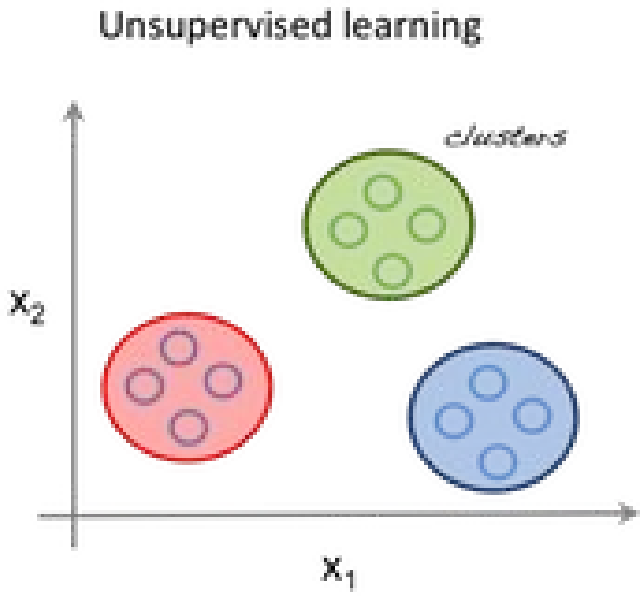
- **Data:** $x = \{x_1, x_2, .., x_n\}$ vector of values

  No target value (output) y

- **Objective:**
  - learn relations between samples, components of samples

Supervised vs Unsupervised Learning





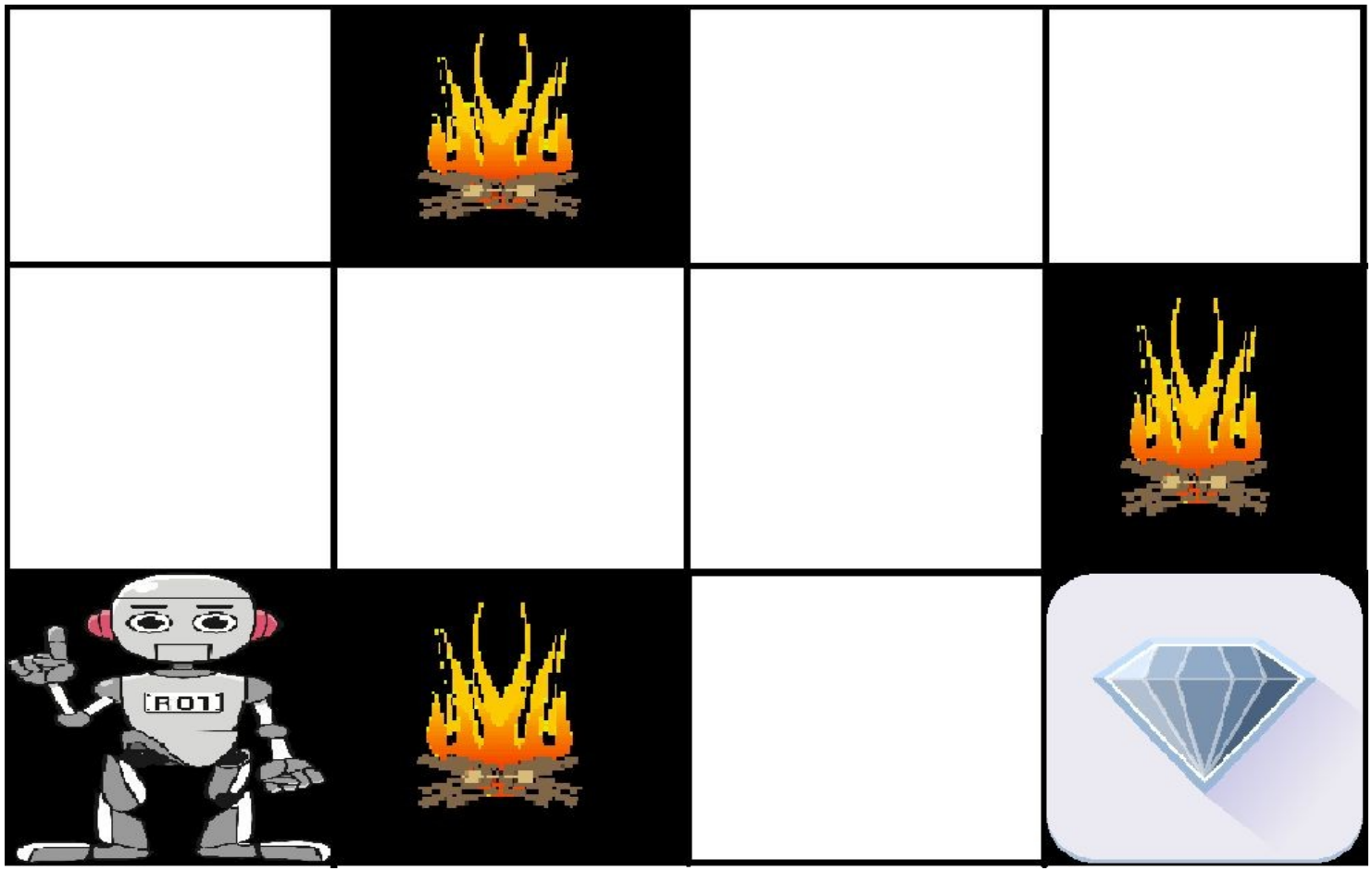
Reinforcement learning

- The learning system, called an agent, can observe the environment, select and perform actions:
  - Get positive rewards for good actions
  - Get negative rewards for wrong action
- Reinforcement learning refers to goal-oriented algorithms, which learn how to attain a complex objective (goal) or maximize

Reinforcement learning

- It must then learn by itself what is the best strategy
  - Policy: best strategy
- A policy defines what action the agent should choose when it is in a given situation.
- Example:
  - Playing games, Robotics
  - Robots learn how to walk.
  - DeepMind's AlphaGo

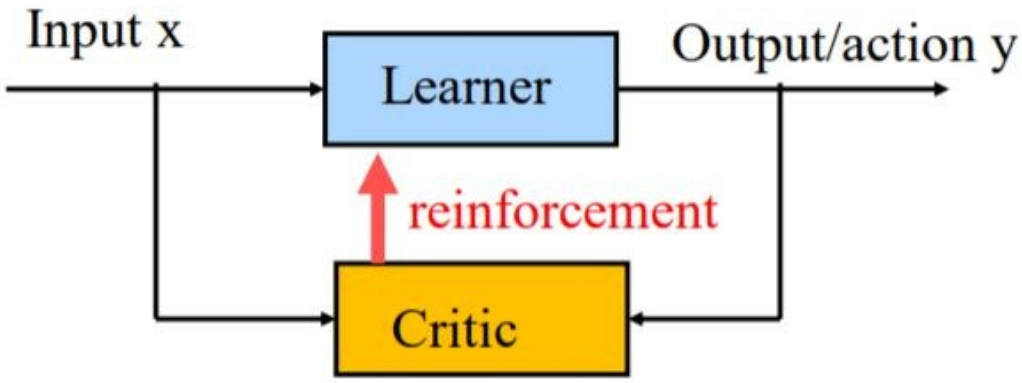
Reinforcement learning



Reinforcement learning

We want to learn: $f : X \rightarrow Y$

- We see examples of inputs **x** but not y
- We select y for observed x from available choices
- We get a feedback (reinforcement) from a **critic** about how good our choice of y was



- The goal is to select outputs that lead to the best reinforcement

Curiosity: Question of the Day

- What if the data is changing, should we retrain the model from scratch?
- Or anything else can be done?

Batch and online Learning

- **Batch learning**
  - The model must be trained using all the available training data.
- Take time and lot of computing resources
- First the system is trained, and then it is deployed into production environment
- No more learning
- This is called *offline learning*.

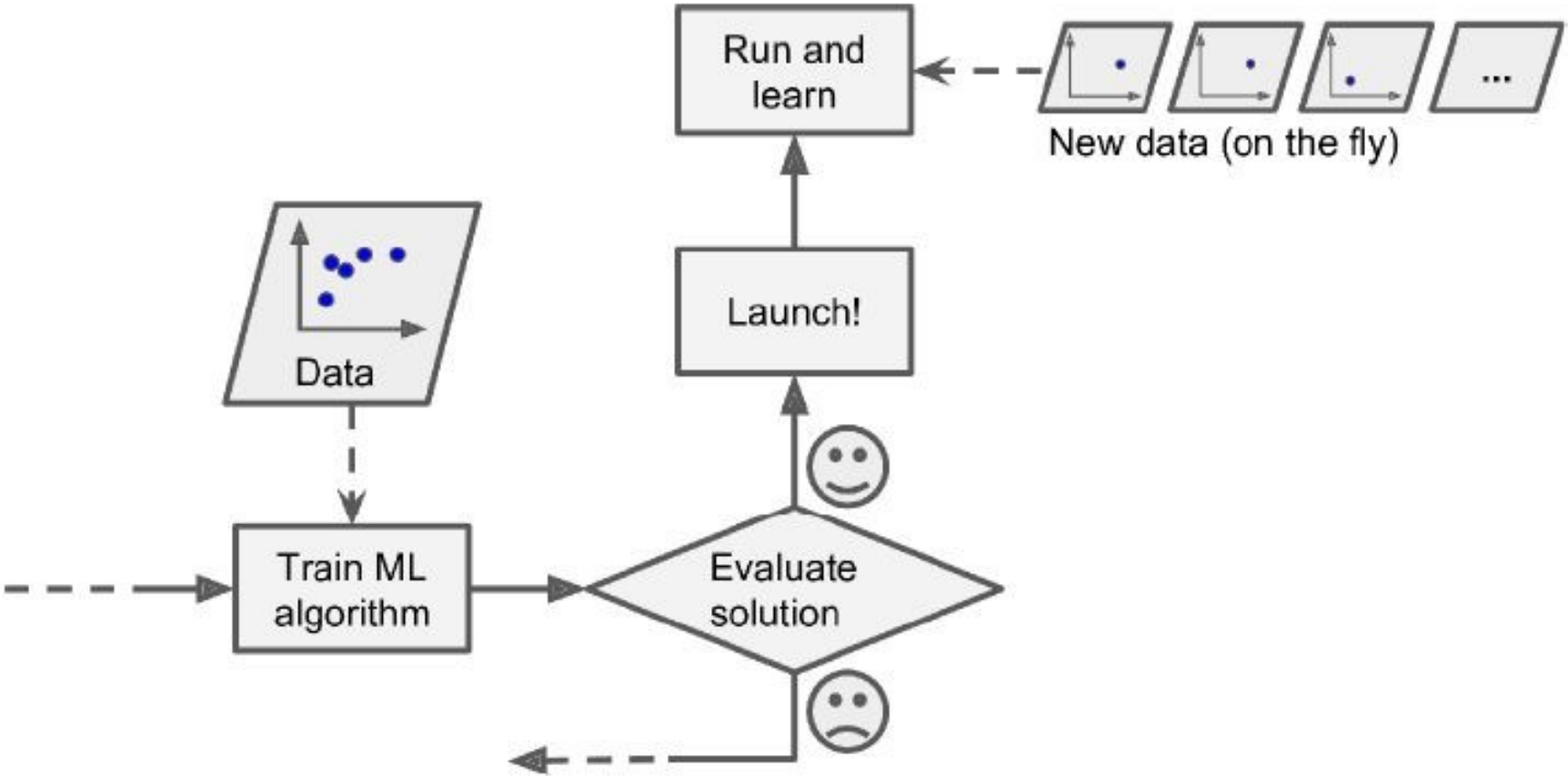
Batch and online Learning

- Batch learning systems can be adopted to new data by training it from scratch on the full dataset
  - new data + old data
- Deploy the new system to production again
- Disadvantage:
  - Training on frequently changing data may be practically infeasible.
  - Time and Computing resources wasted

Batch and online Learning

- **Online learning**
- The system is trained incrementally by feeding data instances sequentially
  - Can be feed data individually or in small groups called minibatches.
- Can perform learning fast
- The system can learn about new data on the fly
- A model is trained and launched into production, and then it keeps learning as new data comes in.

Batch and online Learning



Advantages of Online learning:

- Suitable for data with a continuous flow (e.g., stock prices)
- Adapt to change rapidly or autonomously.
- In case of limited computing resources availability
- Once it learns learned data instances, it does not need them anymore
- Can be used to train systems on huge datasets
  - All data cannot fit in main memory, called *out-of- core* learning.
  - Only part of data is loaded into memory, train model on it, and repeats the process on all of the data

Challenges with online learning

- The performance of the model may gradually decline if low quality data or bad data is feed into it.
  - Data can be corrupted, e.g. hardware malfunctioning like sensor or robot.
- The live system might suffer
- Monitor the model
  - Switch back to old version in case of much decline in performance

Reference

- Read 1st Chapter of Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow- (2019)
- *https://towardsdatascience.com/workflow-of-a-machine-learning-project-ec1dba419b94*
- *https://cloud.google.com/ai-platform/docs/ml-solutions-overview*

Thank You ☺