Machine Learning with Python

Introduction to ML (Part2)

Dr. Aeshah Alsughayyir

Collage of Computer Science and Engineering

Taibah University

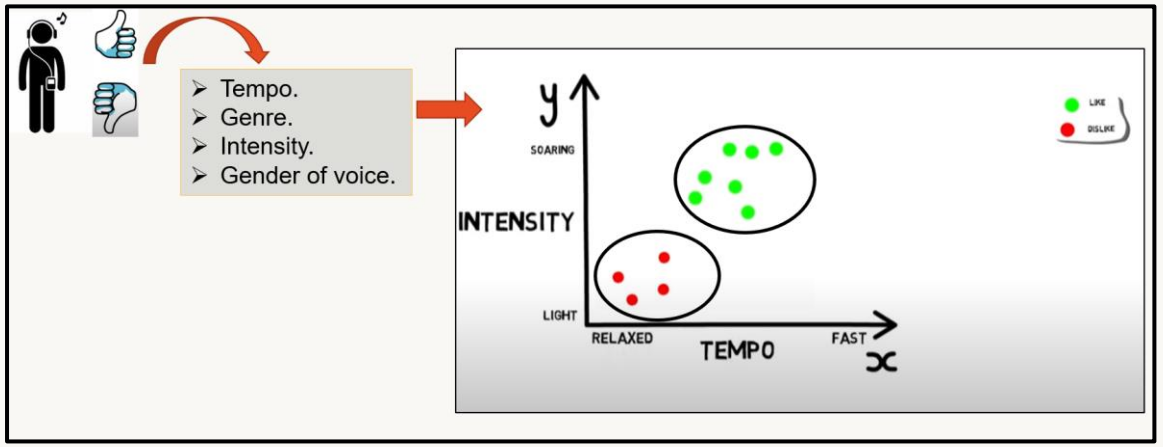
2021-2022

Introduction to Machine Learning (Part 2)

- ✓Machine Learning Definition
- ✓Where is Machine Learning in Computer Science?
- ✓Applications of Machine Learning

- Approaches to Machine Learning
- Learning Process (workflow)

Motivation Example



x → y ←

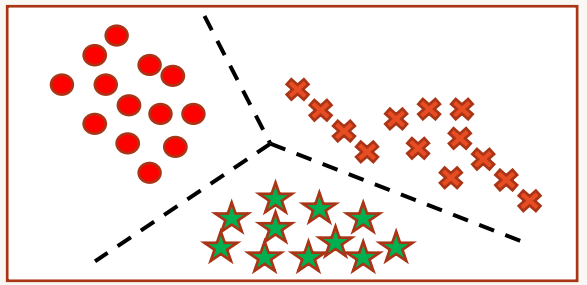
Song ID	Intensity	Tempo	Output (Like/Dislike)
1	Soaring	Fast	Like
2	Soaring	Medium	Like
3	Soaring	Fast	Like
.....	....	....	....
9	Light	Relaxed	Dislike
10	Light	Relaxed	Dislike
11	Light	Relaxed	Dislike

Approaches to Machine Learning

(1) Supervised learning

Data is labeled and comes in pairs *(input, output)*$=(\mathbf{x}, y)$. We need to find a mapping $f(\mathbf{x}) = y$

- If y is a discrete label: classification
- If y is a number: regression

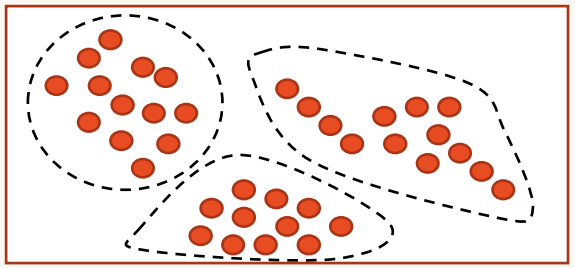


Supervised learning

(2) Unsupervised learning

Data comes without labels. We need to find regularities in the data.

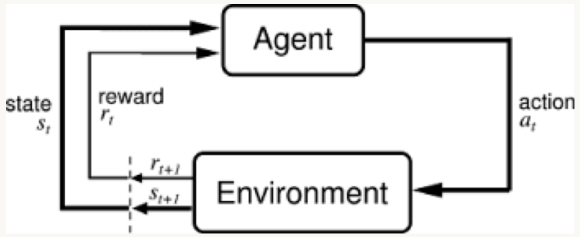
- clustering
- novelty/outlier detection
- dimensionality reduction



Unsupervised learning

(3) Reinforcement learning

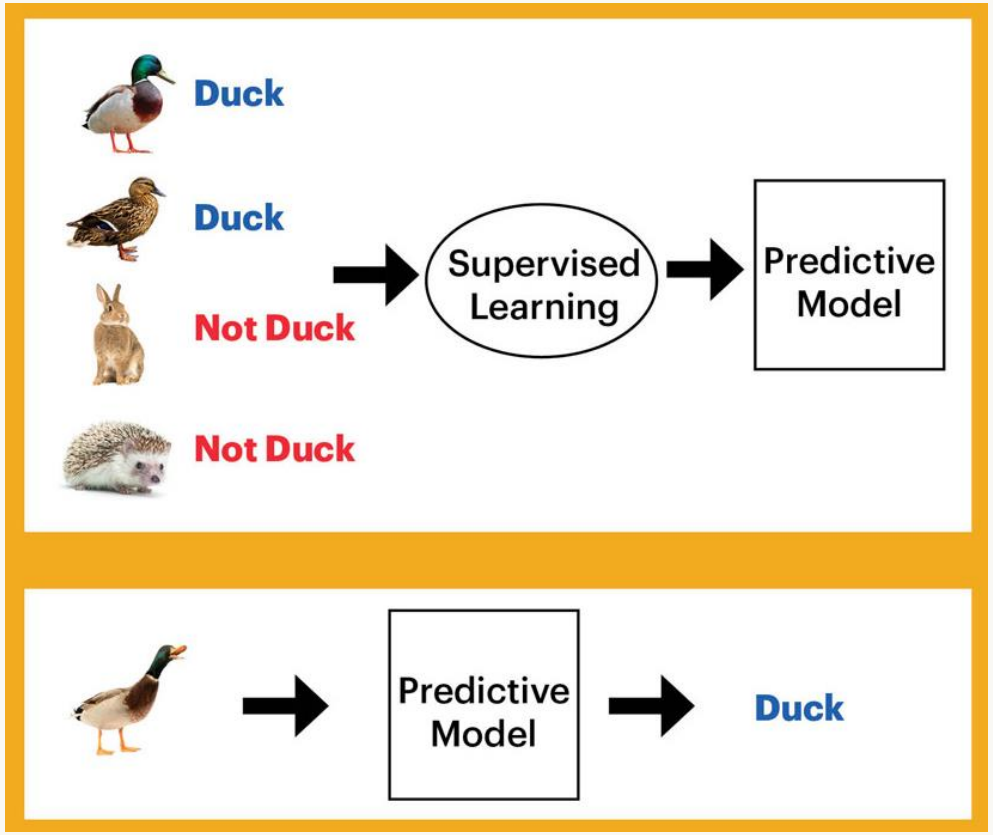
Learning an optimal strategy based on trials with a delayed credit.



Reinforcement learning

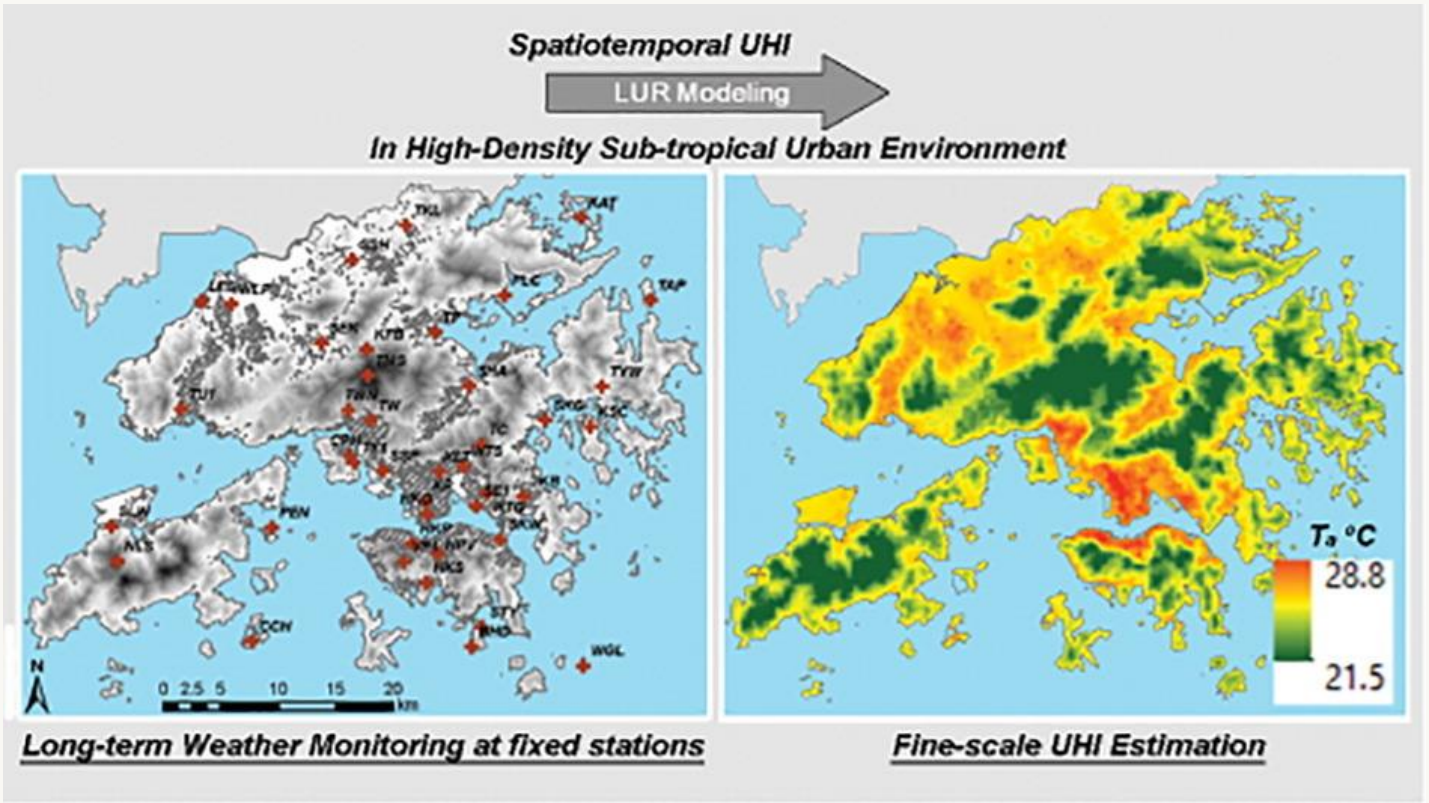
Approaches to Machine Learning (Supervised)

- Classification example:
  *predicting if the input image contains a **Duck***



Approaches to Machine Learning (Supervised)

- Regression example:
  predicting urban heat island (UHI)
  https://www.sciencedirect.com/science/article/pii/S0048969717322738



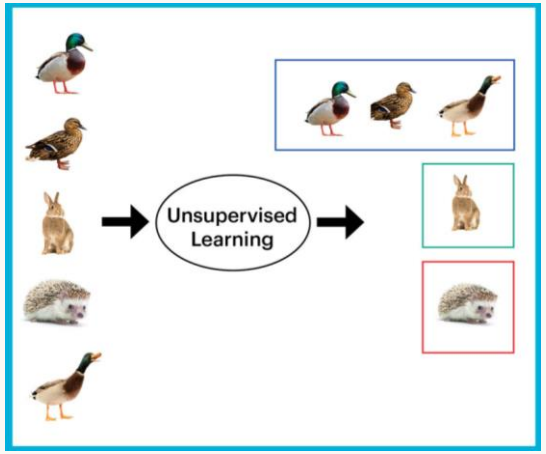
Approaches to Machine Learning (Supervised)

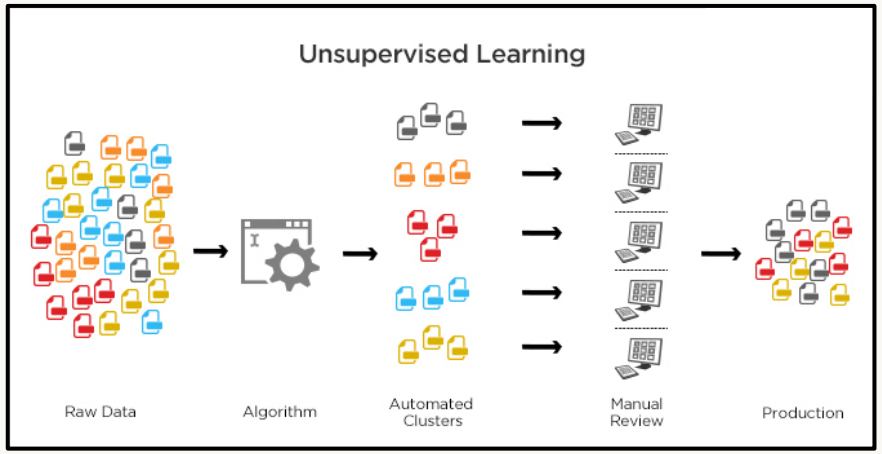
- Classification vs. Regression:

Classification	Regression
• Classification is the task of predicting a discrete class label	• Regression is the task of predicting a continuous quantity
• In a classification problem data is labelled into one of two or more classes	• A regression problem requires the prediction of a quantity
• A classification problem with two classes is called binary, more than two classes is called a multi-class classification	• A regression problem with multiple input variables is called a multivariate regression problem
• Classifying an email as spam or non-spam is an example of a classification problem	• Predicting the price of a stock over a period of time is a regression problem

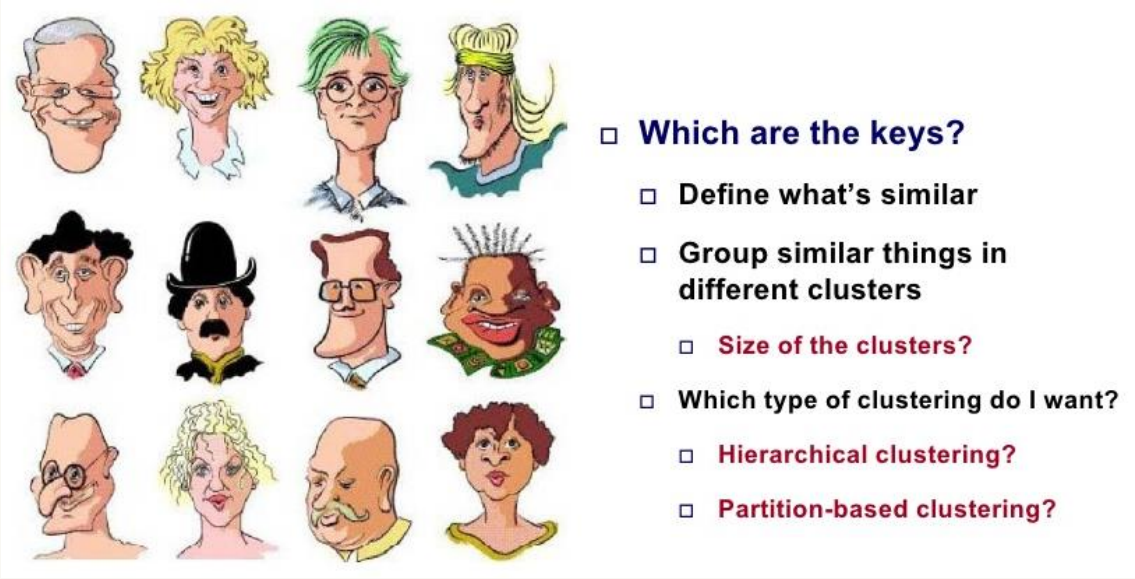
Approaches to Machine Learning (Unsupervised)

- Clustering example:





Put these people in different clustering

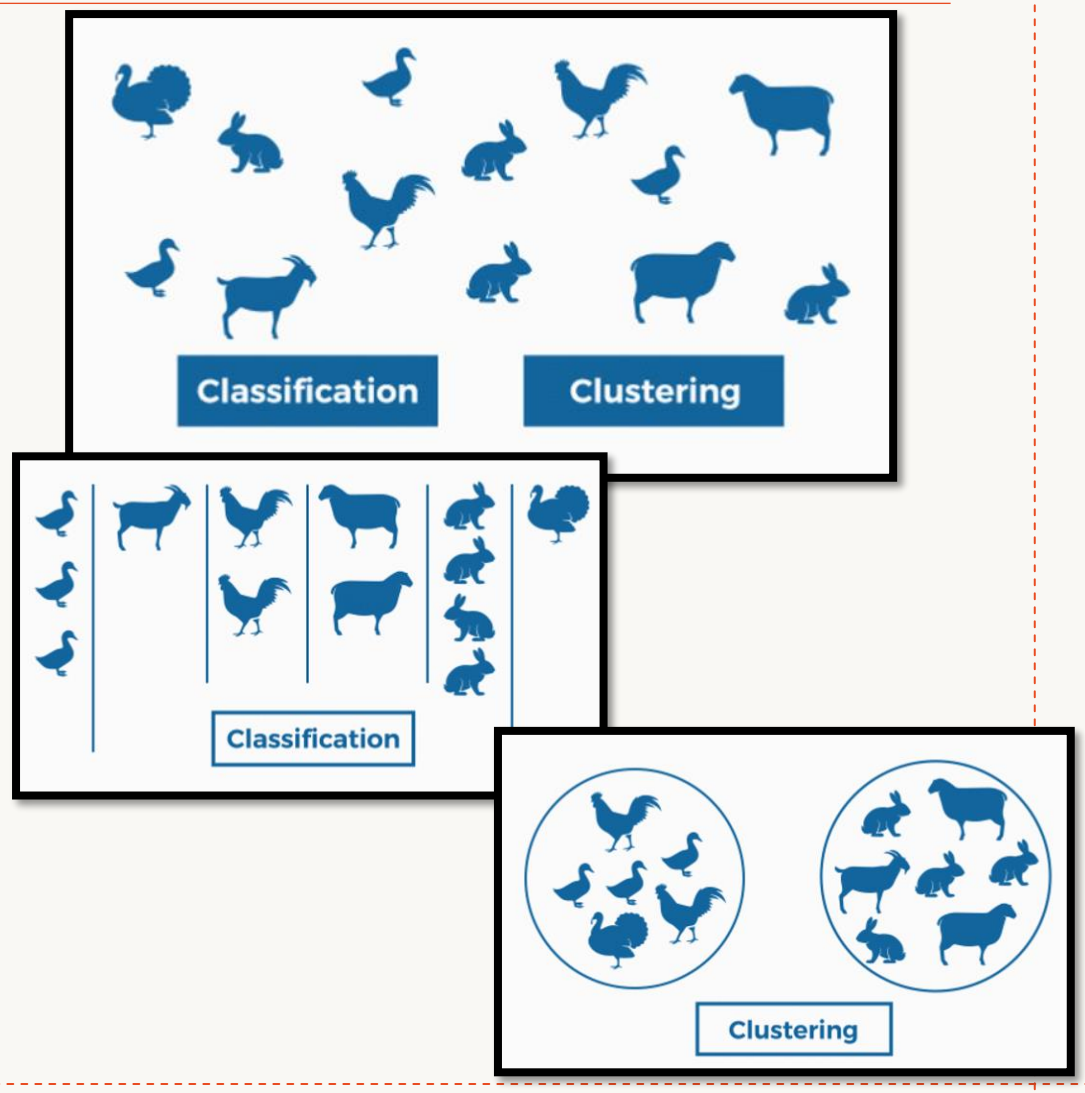


- *Hierarchical clustering* does not require any input parameters, while *partitional clustering* algorithms require the number of clusters (k) to start running.
- *Hierarchical clustering* returns a much more meaningful and subjective division of clusters, but *partitional clustering* results in exactly k clusters.

Classification vs. clustering

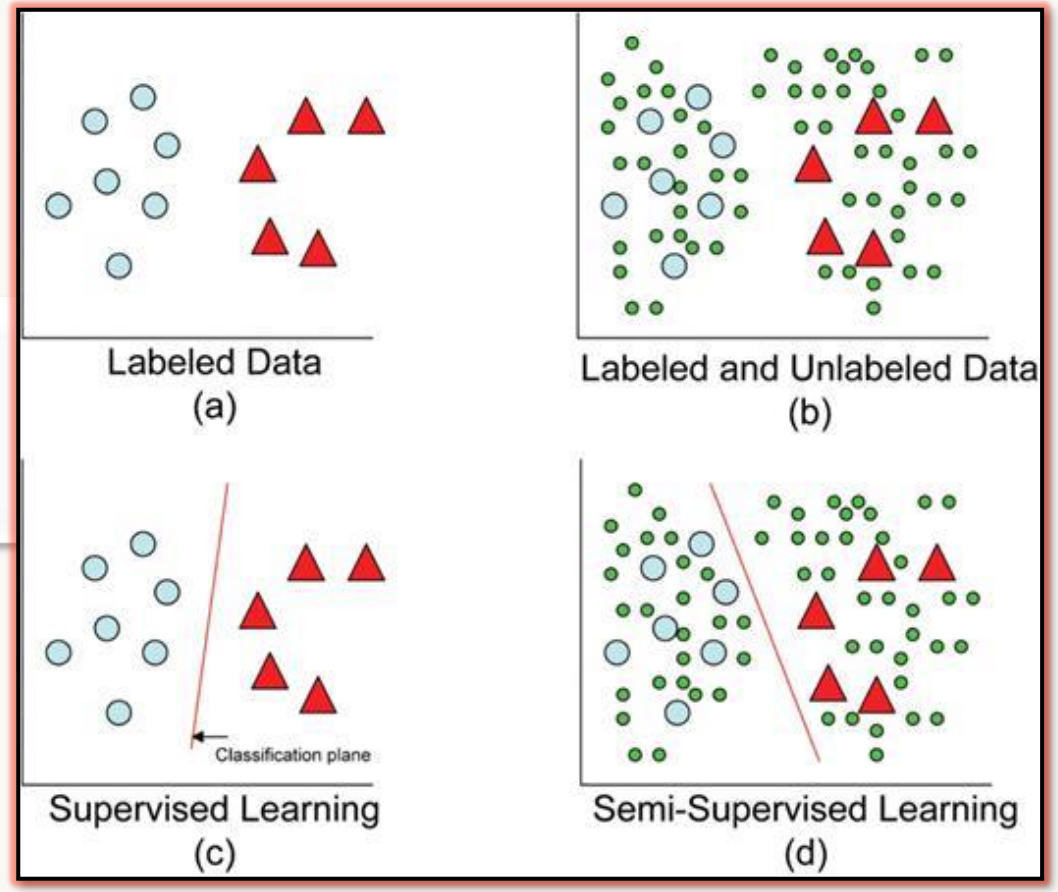
- Classification and clustering are two methods of ***pattern identification used in machine learning***.

- Although both techniques have certain similarities, the difference lies in the fact that classification uses **predefined "classes"** (the data labels) in which objects are assigned, while clustering **identifies similarities between objects**, which it **groups** according to those characteristics in common and which differentiate them from other groups of objects. These groups are known as "**clusters**".



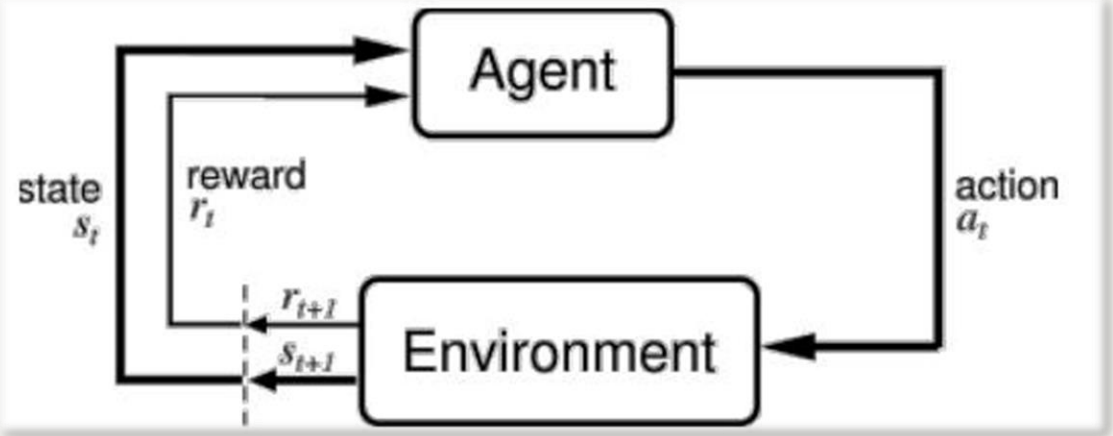
Approaches to Machine Learning (semi-supervised)

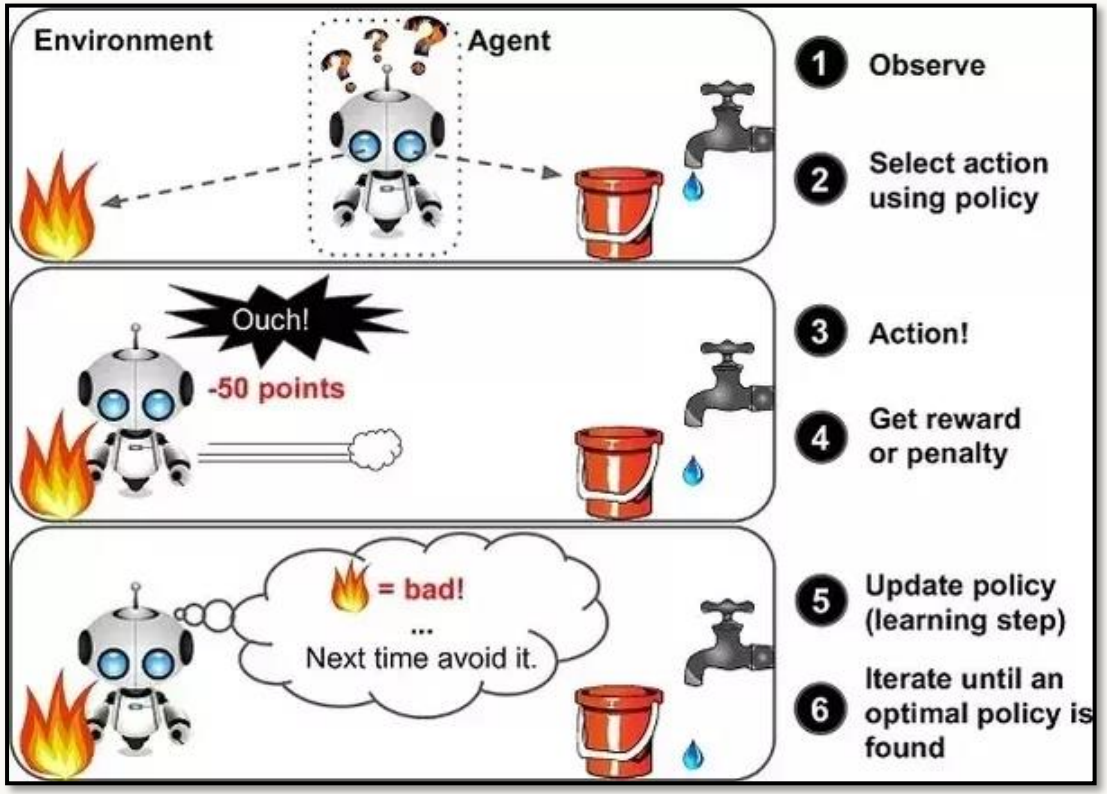
- What is Semi-supervised learning?
  - Supervised learning + Additional unlabeled data
  - Unsupervised learning + Additional labeled data



Approaches to Machine Learning (Reinforcement)

- Basic idea:
  - Receive feedback in the form of rewards
  - Agent's utility is defined by the reward function
  - Must learn to act so as to maximize expected rewards

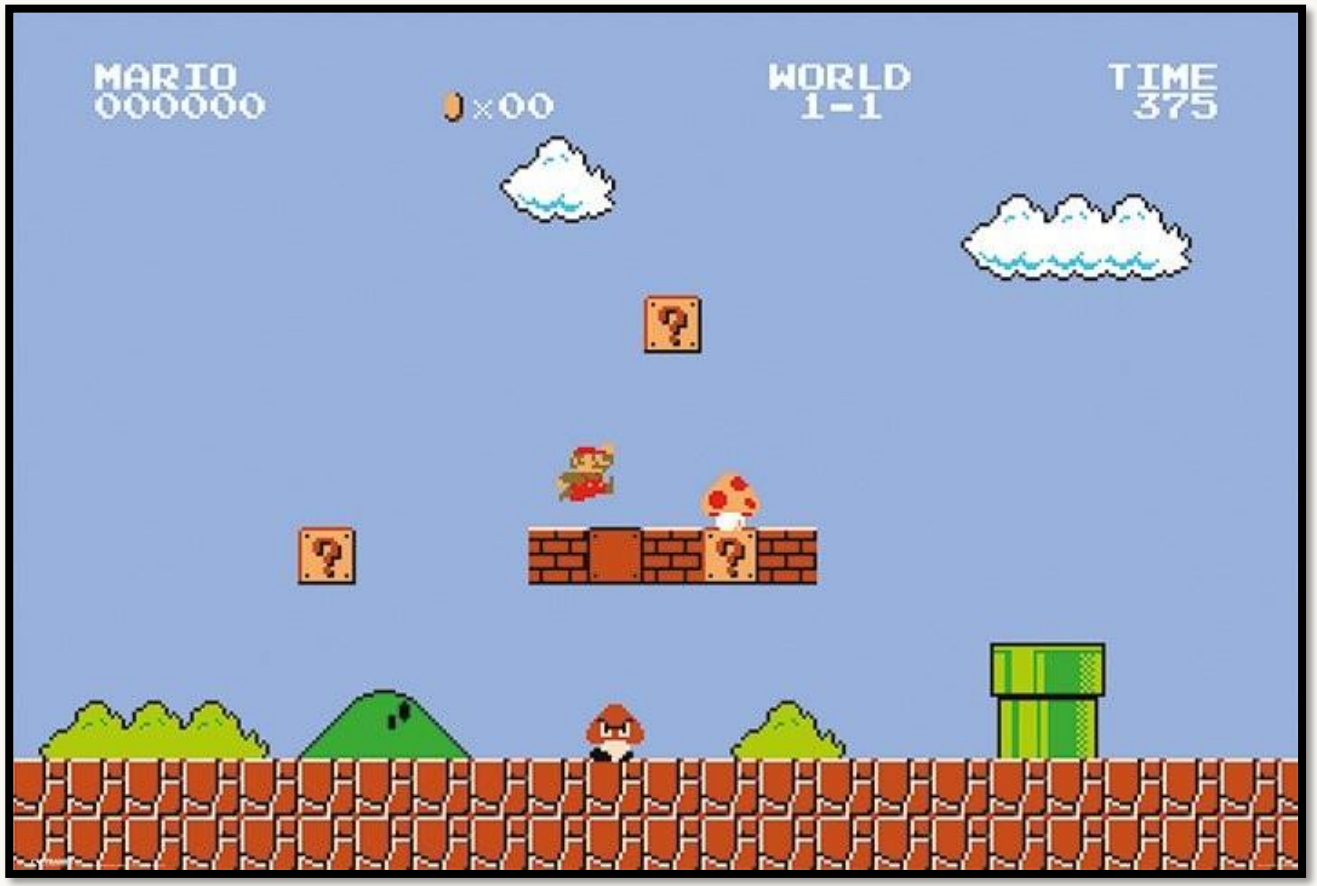




Approaches to Machine Learning (Reinforcement)

- Reinforcement example:

supper Mario



Approaches to Machine Learning (Examples)

Identifying the post-code from handwritten digits on an envelope

Supervised

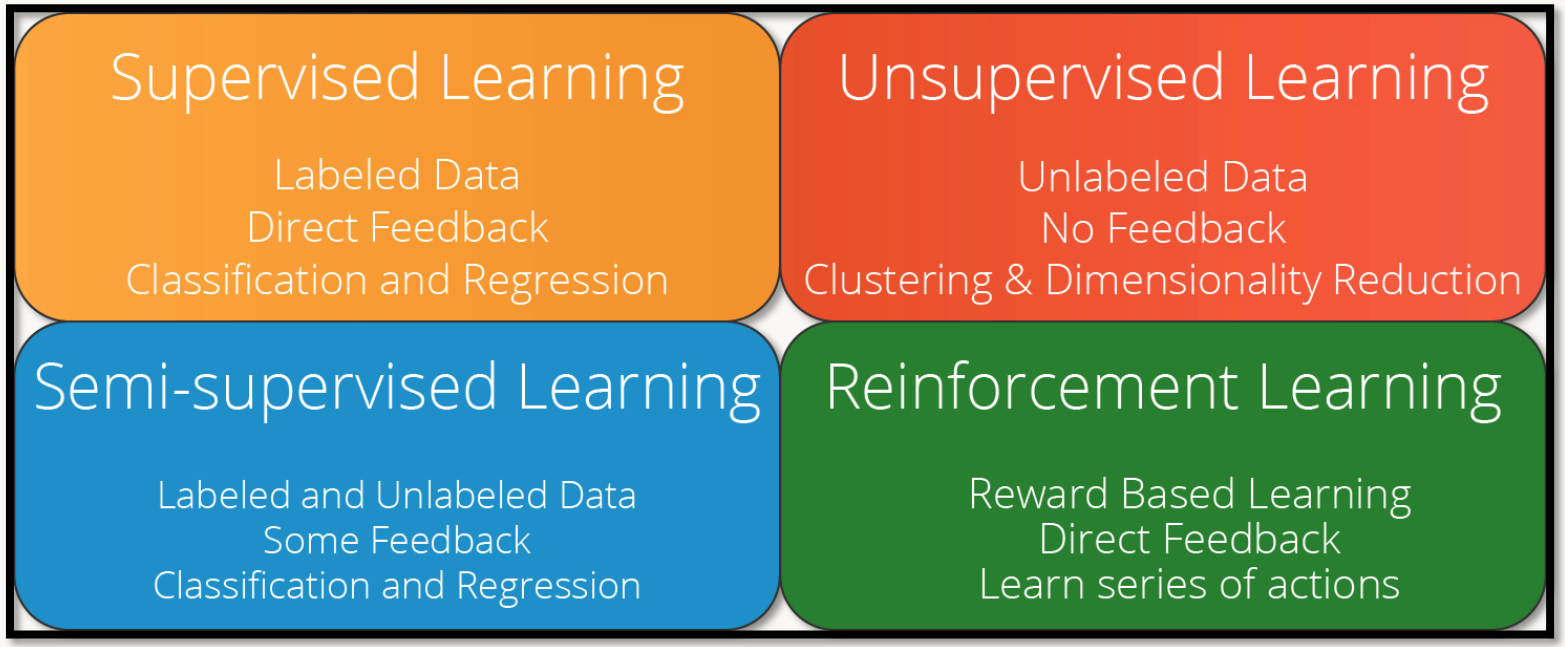
- ✓ Here the input is a scan of the handwriting, and the desired output is the actual digits in the post code.
- ✓ To create a dataset for building a ML model, you need to collect many envelopes.
- ✓ Then you can read the post codes yourself and store the digits as your desired outcomes.

Segmenting customers into groups with similar preferences

Unsupervised

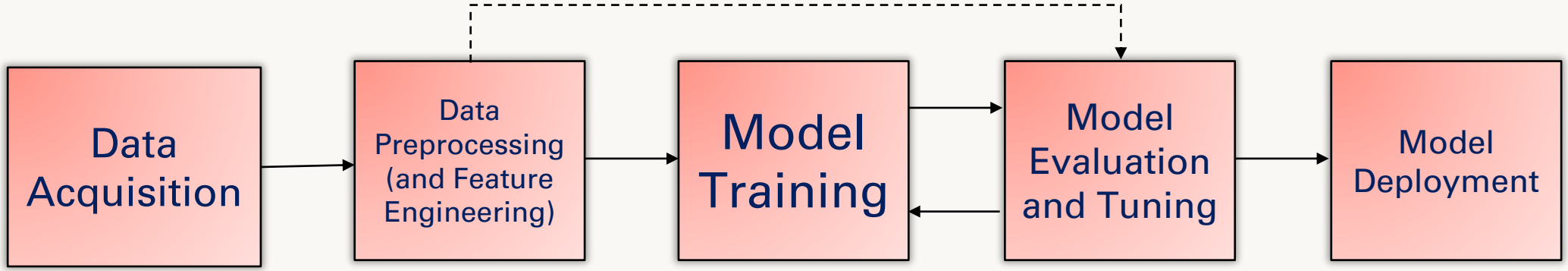
- ✓ Given a set of customer records, you might want to identify which customers are similar, and whether there are groups of customers with similar preferences.
- ✓ For a shopping site, these might be “parents,” “bookworms,” or “gamers.”
- ✓ Because you don’t know in advance what these groups might be, or even how many there are, you have no known outputs.

Summary



Any Question?

Learning Process (Workflow)



➢ Data Acquisition:

- One of the most important part in the ML process is understanding the data you are working with and how it relates to the task you want to solve.

✓ Defining the potential usefulness and accuracy of your project with the quality of the data you collect.

- Having a single dataset can be from one source or aggregated from several sources.
  - Downloading open-source data sets
  - Internet of Things sensors
  - Assorted files, logs, or media.

Data Acquisition (cont.):

- Often it is helpful to think of your data as a table.
- Each data point that you want to reason about (e.g., each transaction) is a row, and each property that describes that data point (e.g., Transaction Date, Customer ID or the Quantity) is a column.

Trans. ID	Transaction Date	Customer ID	Product ID	Shipping Address	Quantity	Unit Price (Rp)	Shipping and Other Expenses (Rp)	Nett Income (Rp)
1	1/12/2017	CUST 01	X1	Self pickup	5	250,000	0	1,250,000
2	1/14/2017	CUST 02	X2	Jakarta	100	35,000	175,000	3,325,000
3	1/14/2017	CUST 02	X3	Self pickup	100	42,000	0	4,200,000
4	1/14/2017	CUST 02	X4	Self pickup	100	33,000	0	3,300,000
5	1/18/2017	CUST 03	X5	Tangerang	30	194,500	115,000	5,720,000
...								
295	5/30/2018	CUST 51	X80	Jakarta	1	125,000	25,000	100,000

Each entity or row here is known as a *sample* (or data point) in machine learning, while the columns—the properties that describe these entities—are called *features*.

➢ Data Preprocessing (and Feature Engineering)

- Pre-processing involves cleaning, verifying, and formatting data into a usable dataset.
- If you are aggregating data from several sources, you need to make sure that:
  - data formats match, data is equally reliable, and remove any potential duplicates.
- No ML algorithm will be able to make a prediction on data for which it has no information (e.g., if the only feature that you have for a patient is their last name).

Feature engineering or *feature extraction*

- It is the process of using domain knowledge to extract features (characteristics, properties, attributes) from raw data.
- The question of how to represent your data best for a particular application is known as Feature Engineering.
- A feature is a property shared by independent units on which analysis or prediction is to be done.
  - It is represented by a column in dataset, and known as dimension
  - It can be *categorical features* (*discrete features)* and *continuous feature*

Data Preprocessing (and Feature Engineering) cont.

- ***Continuous*** is an analog of *Regression,,,* and ***Categorical*** is an analog of Classification,,,

➢ **Example:**

Features

Output (Labels)

Outlook	Temperature	Humidity	Windy	Play
sunny	85	85	false	no
sunny	80	90	true	no
overcast	83	78	false	yes
rain	70	96	false	yes
rain	68	80	false	yes
rain	65	70	true	no
overcast	64	65	true	yes
sunny	72	95	false	no
sunny	69	70	false	yes
rain	75	80	false	yes
sunny	75	70	true	yes
overcast	72	90	true	yes
overcast	81	75	false	yes
rain	71	80	true	no

No.	Outlook	Temperature	Humidity	Windy	Play
1	Sunny	Hot	High	False	No
2	Sunny	Hot	High	True	No
3	Overcast	Hot	High	False	Yes
4	Rain	Mild	High	False	Yes
5	Rain	Cool	Normal	False	Yes
6	Rain	Cool	Normal	True	No
7	Overcast	Cool	Normal	True	Yes
8	Sunny	Mild	High	False	No
9	Sunny	Cool	Normal	False	Yes
10	Rain	Mild	Normal	False	Yes
11	Sunny	Mild	Normal	True	Yes
12	Overcast	Mild	High	True	Yes
13	Overcast	Hot	Normal	False	Yes
14	Rain	Mild	High	True	No

Data Preprocessing (and Feature Engineering) cont.

- Having the right representation of the data is often crucial for supervised or unsupervised learning to succeed.

Note:

- *Representing your data in the right way can have a bigger influence on the performance of a supervised model than the exact parameters you choose.*
- *How you represent your features can have an enormous effect on the performance of machine learning models.*

Data Preprocessing (and Feature Engineering) cont.

- This phase involves breaking processed data into two (or three) datasets—training, validating, and testing:
- Training set—used to initially train the algorithm and teach it how to process information. This set defines model classifications through parameters.
- Validation set—used to estimate the accuracy of the model. This dataset is used to finetune model parameters.
- Testing set—used to assess the accuracy and performance of the models. This set is meant to expose any issues or mistrainings in the model.

➢ Model Training

Once you have datasets, you are ready to train your model.

- This involves feeding your training set to your algorithm so that it can learn appropriate parameters and features used in classification.

> **Training** is the process of feeding an ML algorithm with data to help identify and learn good values for all attributes involved.

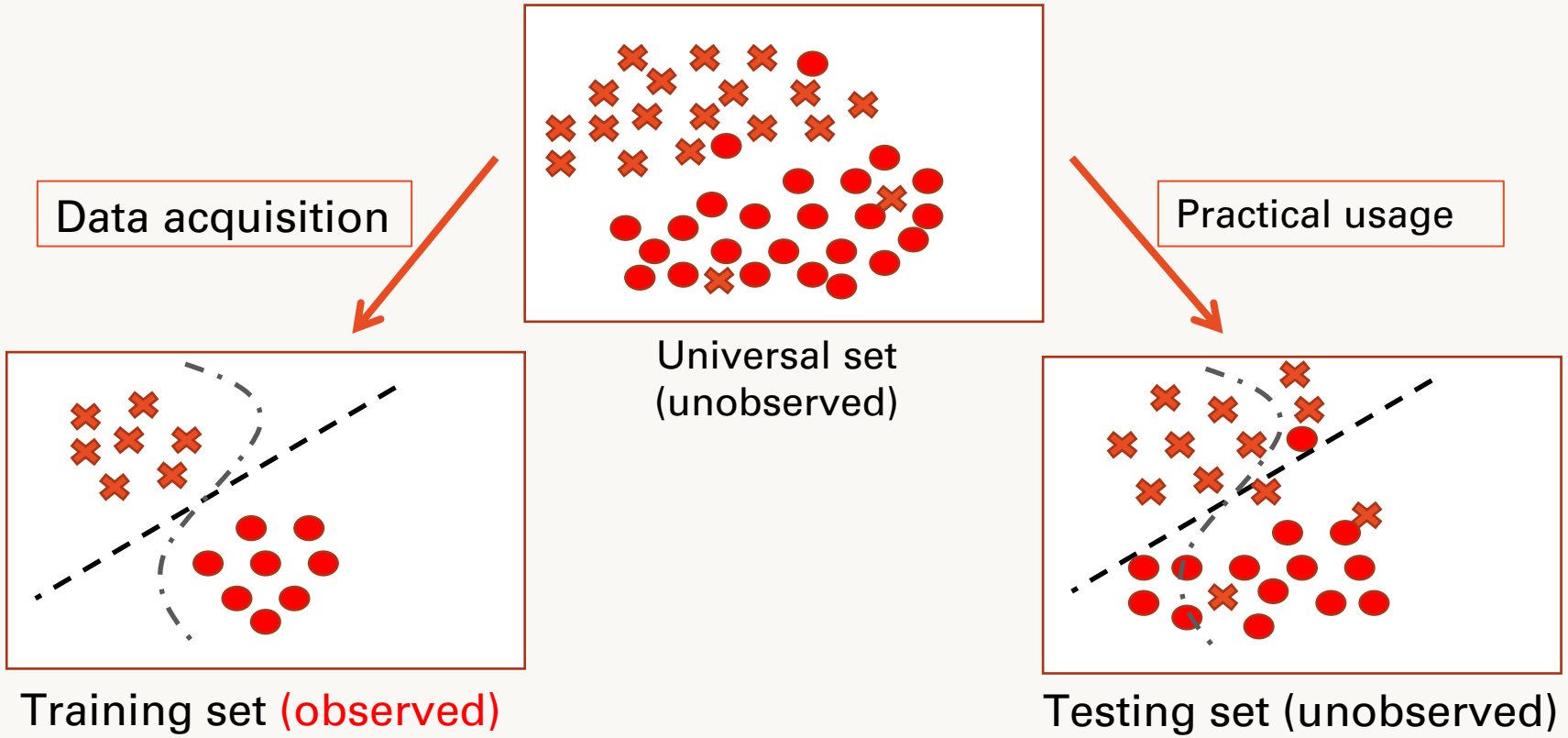
- There are several types of ML models, of which the most common ones are supervised and unsupervised ones.
- Once training is complete, you can then refine the model using your validation dataset.
  - This may involve *modifying* or *discarding variables* and includes a process of tweaking model-specific settings (hyperparameters) until an acceptable accuracy level is reached.

✓ Consistent training can significantly improve the prediction rate of the ML model.
✓ The weights of the model must be initialized randomly.
✓ This way the algorithm will learn to adjust the weights accordingly.

Model Training (cont.)

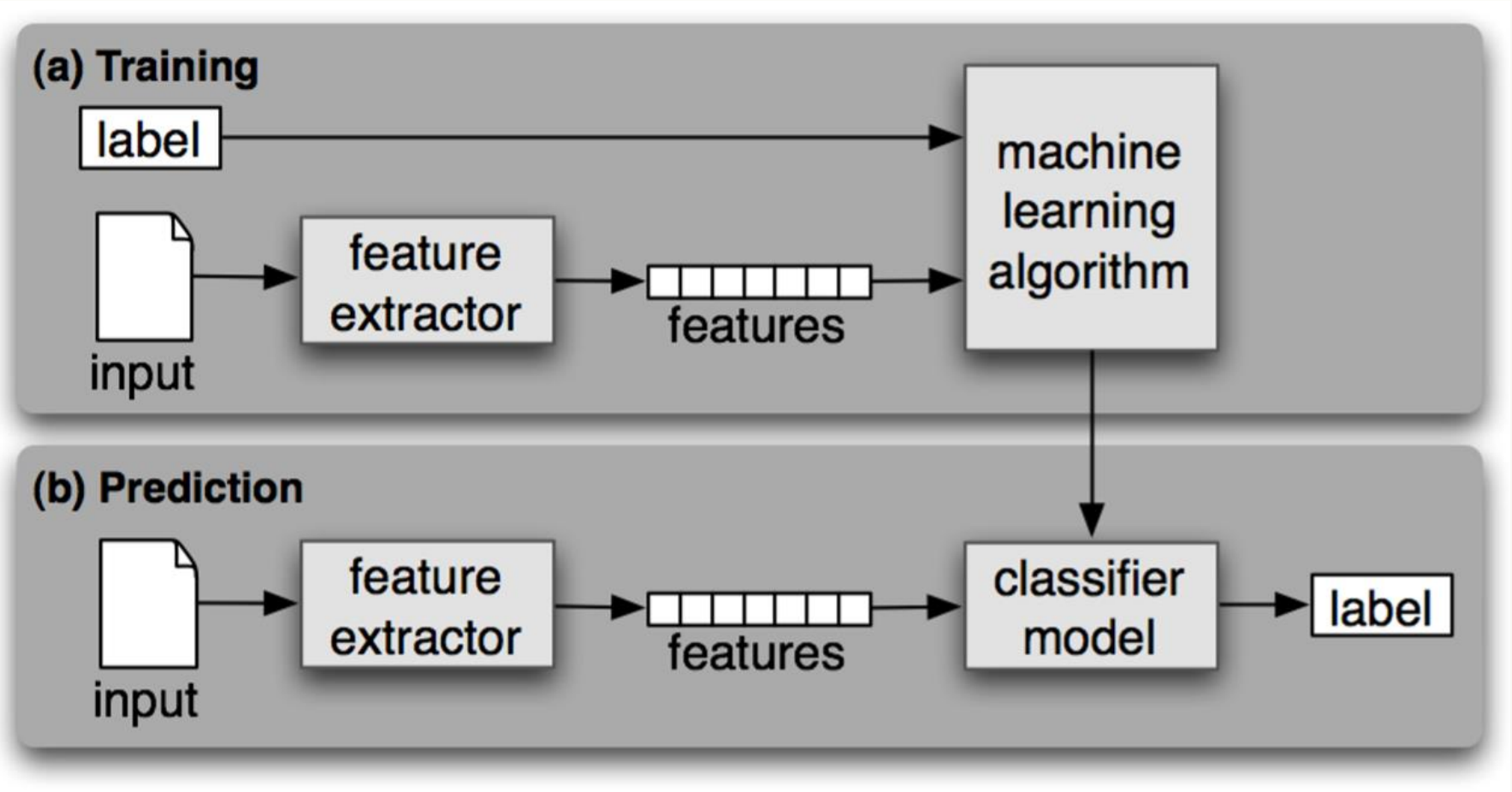
Supervised learning: *Training and prediction*

- **Training** is the process of making the system able to learn.
- **Training set** and **testing set** come from the same distribution



Model Training (cont.)

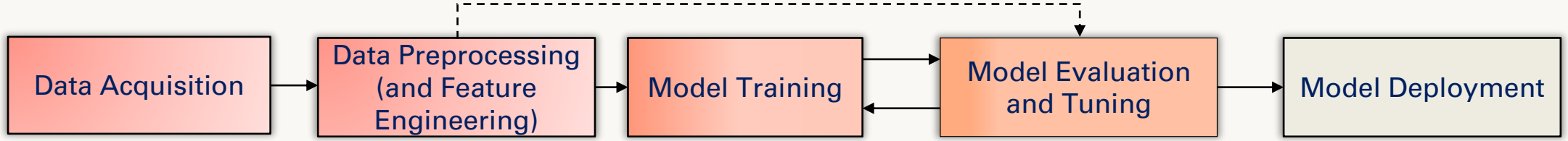
Supervised learning: *Training and prediction* (cont.)



Model Training (cont.)

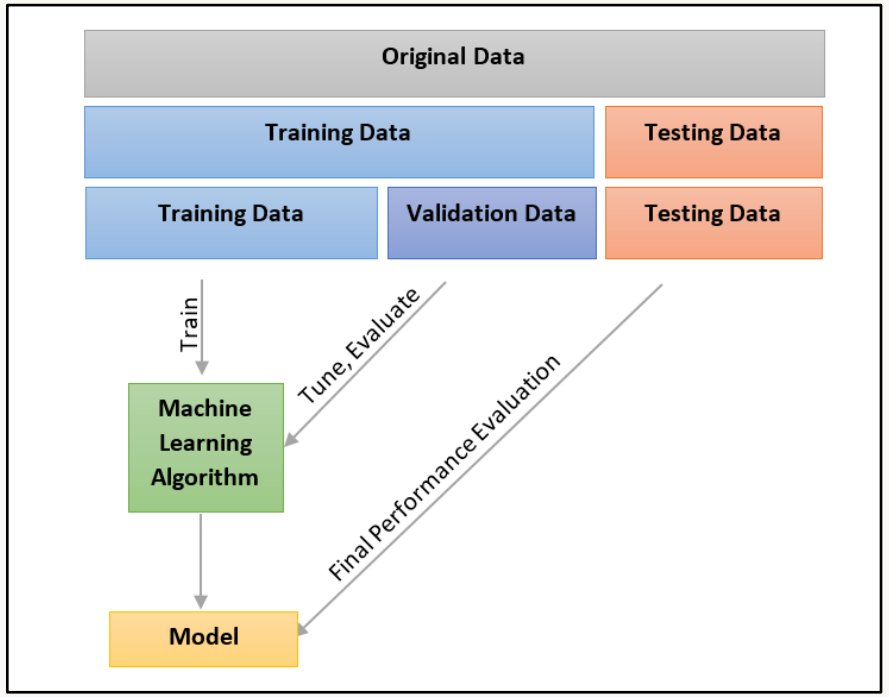
Supervised learning: *Training and prediction* (cont.)

- A supervised model first needs to be trained on labelled data, and after that the model can be used for prediction (*classification or regression*) on previously unseen data.
- Each data instance is represented as a vector of features x, where each feature encodes one aspect of the input (*e.g., height, weight, income, ...*).
- Feature extractor generates the feature vector x for each data instance (this step is identical for both training and prediction).
- During training, the model is fed with data instances x and their labels y, i.e., the pairs (x,y).
- Once the model is trained, it will be used for prediction.
- During prediction, the model will take only data instances x as input, and in turn generates y as output.



Model Evaluation and Tuning

- The *training dataset* used to fit (train) the model and, also, can be further split into a *training set* and a *validation set*.
- This subset of the training dataset, called the *validation set*, can be used to get an early estimate of the skill (accuracy) of the model.
- "The model is fit on the training set, and the fitted model is used to predict the responses for the observations in the validation set. The resulting validation set error rate provides an estimate of the test error rate." (Gareth James, et al., 2013)
  - *typically assessed using Mean Square Error MSE in the case of a quantitative response*.

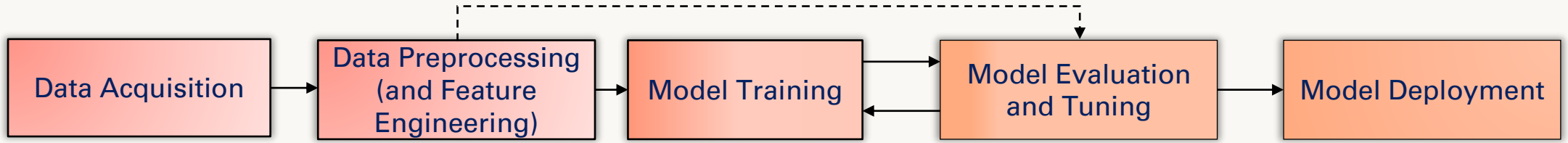


The set of parameters that are selected based on their influence on the model architecture are called *hyperparameters*.

- The process of identifying the hyperparameters by tuning the model is called parameter tuning.
- The parameters for correlation should be clearly defined in a way *the point of diminishing returns* for validation is as close to 100% accuracy as possible.

Model Evaluation and Tuning (cont.):

- Finally, after an acceptable set of hyperparameters is found and your model accuracy is optimized you can test your model.
- Testing uses your test dataset and is meant to verify that your models are using accurate features.
- Based on the feedback you receive you may return to training the model to improve accuracy, adjust output settings, or deploy the model as needed.



➢ Model Deployment

Deployment is the process of integrating a ML model into an existing production environment where it can take in an input and return an output. So, its predictions are available to users or other systems.

There are various approaches to deploy a ML model:

- **One-off** : Trained once or ad-hoc when its needed.
- **Batch training**: Continuously up-to-date version of the model / good with model that used on a consistent basis.
- **Real-time prediction**: When a prompt prediction is required/ using Online ML model: *stochastic gradient descent*.

Example:

- Basic workflow for training a supervised machine learning model:
  - No matter what ML algorithm you use the basic process is the same

- For The project of **predicting home prices**:
  - Collect data for houses sold in one area for a year.

Step 1: Get Data

bedrooms	size	stories	sale_price
5	3800 sq. ft.	2	$400,000
4	2200 sq. ft.	1	$150,000
2	1150 sq. ft.	1	$300,000

Step 2: Clean and Preprocess Data

bedrooms	size	stories	sale_price
5	3800	2	400000
4	2200	1	150000
2	1150	1	300000

Example (cont.)

Step 3: Shuffle the Data

bedrooms	size	stories	sale_price
2	1150	1	150000
4	2200	1	300000
5	3800	2	400000

So, the ML algorithm doesn't pick up a pattern that isn't really there! Because the data in certain order.

Usually with 70% of rows in the data for training the model, and 30% for testing

Step 4: Split into Training/Test Data

Training Data

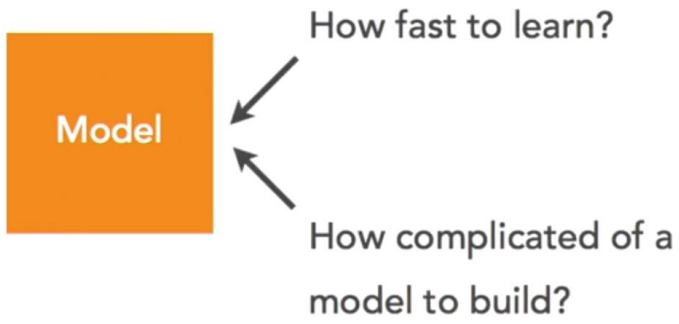
bedrooms	size	stories	sale_price
2	1150	1	150000
4	2200	1	300000

Test Data

bedrooms	size	stories	sale_price
5	3800	2	400000

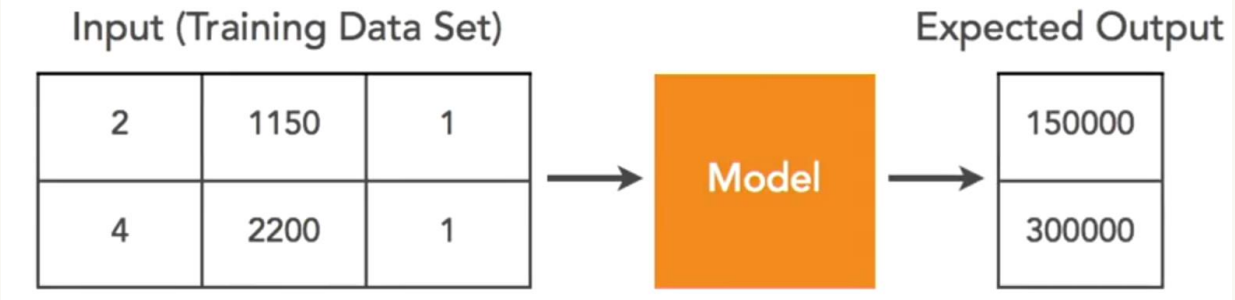
Example (cont.)

Step 5: Set Model Hyperparameters



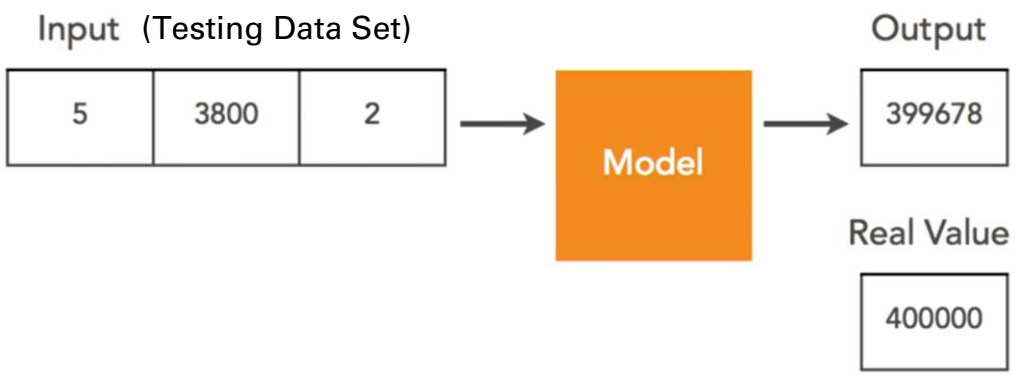
Each ML algorithm has a few parameters we have to set. These parameters will control things like how fast the model should learn data and how complex of a pattern to look for .

Step 6: Train the Model



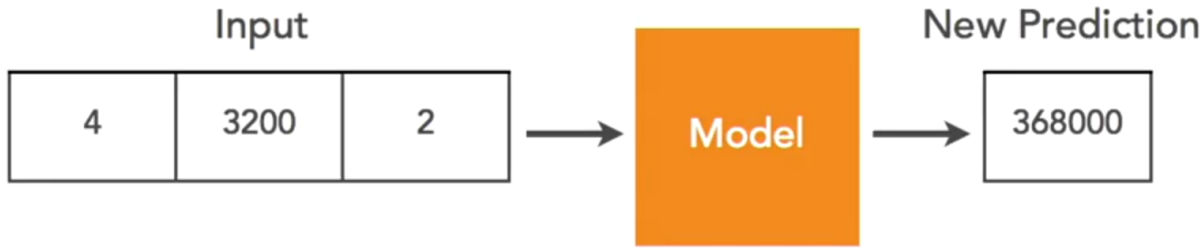
Example (cont.)

Step 7: Evaluate the Model

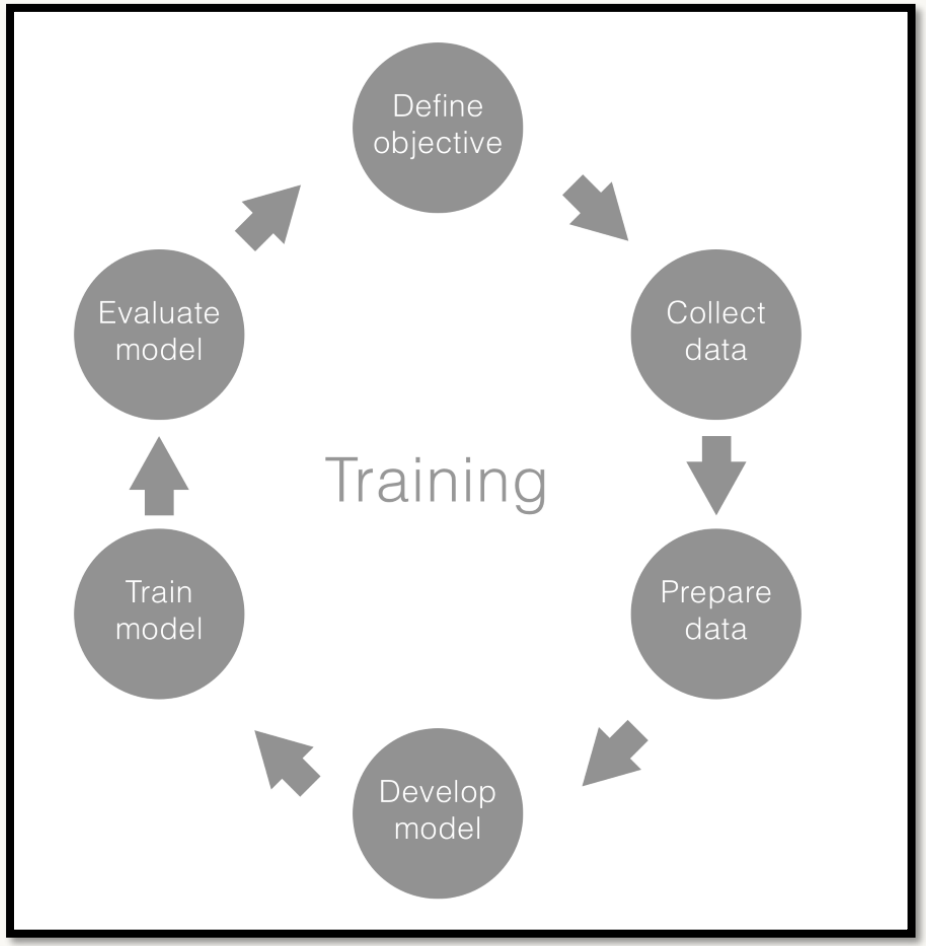


We need to check the accuracy of the model against the test data set.. This proves if the model works well on new data or if it only works on the training data.

Step 8: Use the Model



Summary



Any Question?