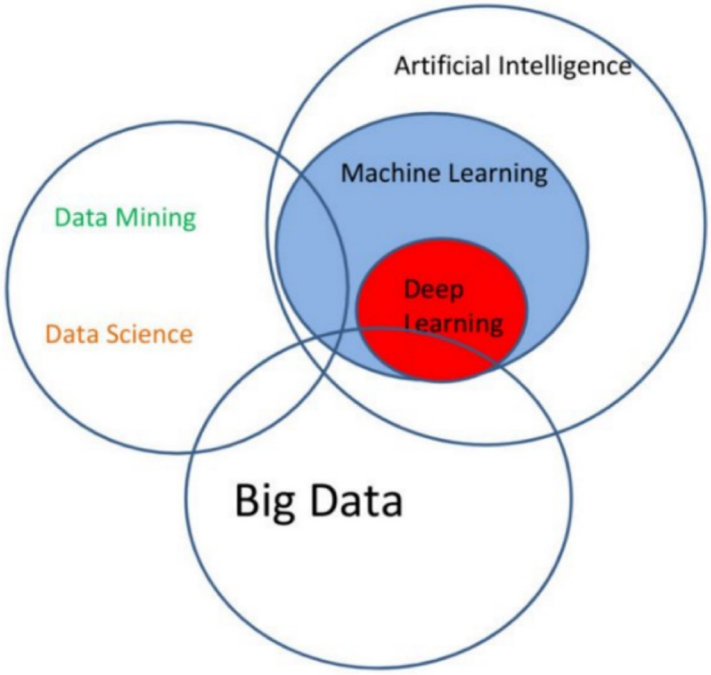
1 Introduction

Some terminology and high level concepts to get you started.

1.1 Machine Learning



Machine Learning Paradigms

Machine Learning (ML): a branch of artificial intelligence (AI) devoted to developing and understanding methods that "learn", i.e. that…

> … leverage data to make predictions or decisions (act like humans) without being explicitly programmed to do so

(Arthur Samuel, Wikipedia).

Model: A model is a logical, mathematical or probabilistic relationship between several variables.

Learning (also called *training*): Machine Learning employs adaptive models, which are configured and parameterised automatically based on the training data.

The computational methods in Machine learning are used to discover patterns in the data and/or derive a corresponding generating process to

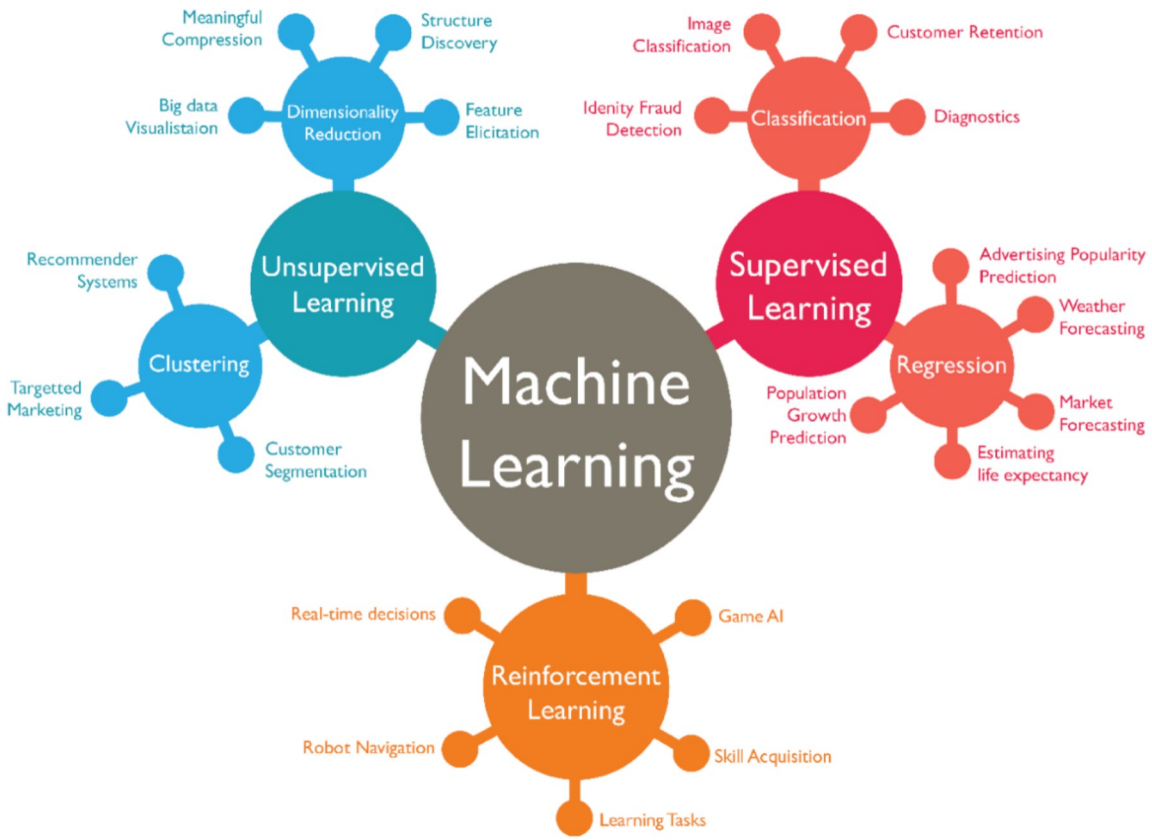
- gain insights and
- predict events

in order

- to provide a quantitative basis for decisions (actionable insights), e.g. determine target segment for marketing campagne, and
- to influence the underlying process of the data, e.g. adapt the user features of an app.

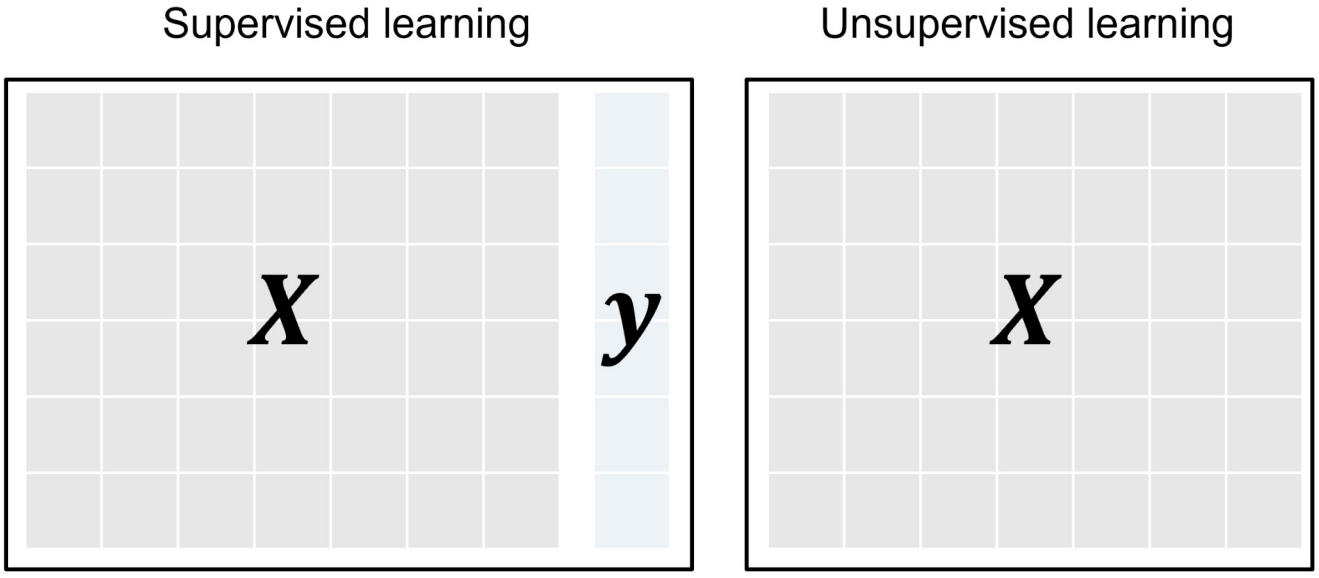
Deep Learning is a a subset of machine learning that employs *deep learning networks* - models inspired by the neurons in the human brain.

1.2 Machine Learning Paradigms



The three main machine learning paradigms are supervised, unsupervised, and reinforcement learning.

1.3 Supervised Learning



Let's consider a dataset characterised by

M: Number of training samples

N: Number of features
Dimension X: $M \times N$
Dimensions y: M

In **supervised learning** the training data consists of input samples $X_{m,:}$ (rows in the design matrix $\mathbf{X}$) and their associated output values $y^{(m)}$ (note that we denote single samples with superscript $^{(m)}$).

In supervised learning we try to find a function f which systematically produces the output values y_m associated with the input values $\boldsymbol{X}_{m,:}$ per sample m: $f(\boldsymbol{X}_{m,:}) \rightarrow y^{(m)}$. Supervised learning aims to find a mapping from input data to their corresponding outputs.

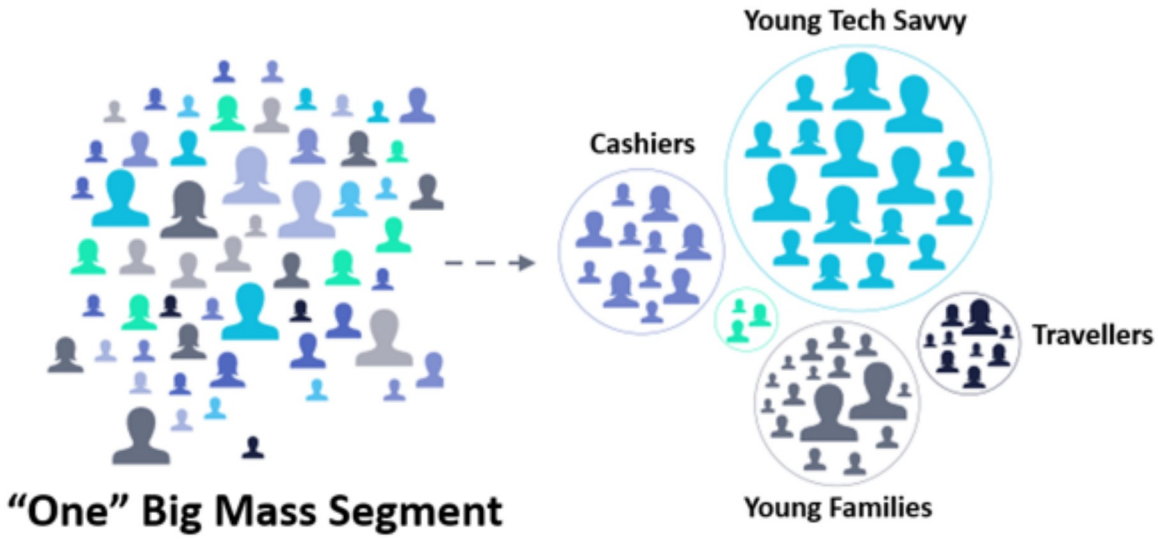
The columns of $\mathbf{X}$ represent the independent variables, also called features, predictors, attributes or covariates. Inputs can vary widely depending on the application: can be tabular data (properties of a car like mileage, age, brand), text (product reviews), images (object detection) which are converted to numerical format. The associated output values $y^{(m)}$ are individual observations of the target variable, also called response variable. In classification settings these values are also called labels.

1.4 Unsupervised Learning

In **unsupervised learning** the training data does not contain any output values. The goal is to model the underlying distribution of the data $\mathbf{X}$ (describe the structure of the data), in order to explain it and to apply the model to new data. This brings additional challenges compared to supervised learning:

- The problem statement is much fuzzier
- The evaluation is more difficult without test data including expected output values

Example: **Clustering** is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters)



1.5 Reinforcement Learning

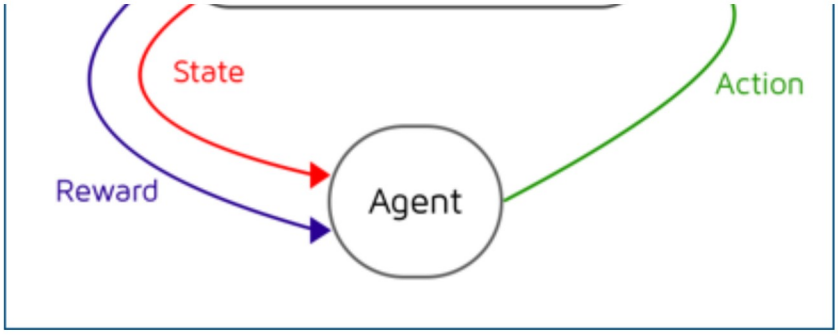


Figure 1.1: Reinforcement learning

In **Reinforcement Learning**, the learning system, called an agent in this context, can observe the environment, select and perform actions, and get rewards in return (or penalties in the form of negative rewards), as shown in Figure 1.1. It must then learn by itself what is the best strategy, called a policy, to get the most reward over time. A policy defines what action the agent should choose when it is in a given situation.

1.6 Details of Supervised Learning

We now discuss some typical methods and technologies of **supervised learning**.

1.6.1 Classification vs. Regression

Supervised learning problems are distinguished regarding the type of the target variable $\mathbf{y}$.

In **classification** the target variable is **categorical**, typically on a nominal scale (the classification algorithms discussed in this course treat ordinally scaled data on a nominal scale). The output values are assumed to belong to a set of discrete classes $y^{(m)} \in \{C_1, C_2, \ldots, C_K\}$

A spam filter is a good example of this: it is trained with many example emails along with their corresponding labels pertaining to two classes, nemaley spam and ham, and the goal is to learn to correctly classify (assign labels to) new emails as spam or ham.

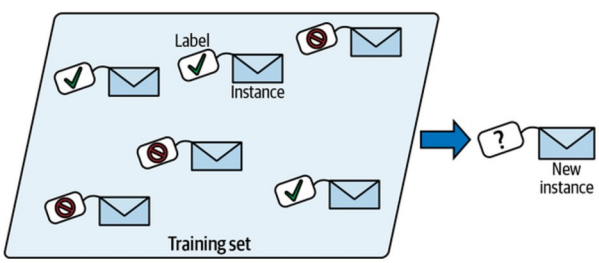
Another typical task is to predict a **numeric** (continuous) target value, i.e. $y^{(m)} \in \mathbb{R}$, such as the price of a car, given a set of input features (mileage, age, brand, etc.). This type of task is called **regression**.

Note that some machine learning methods have variants for both types of tasks, classification and regression.

Classification

Target variable y: categorical

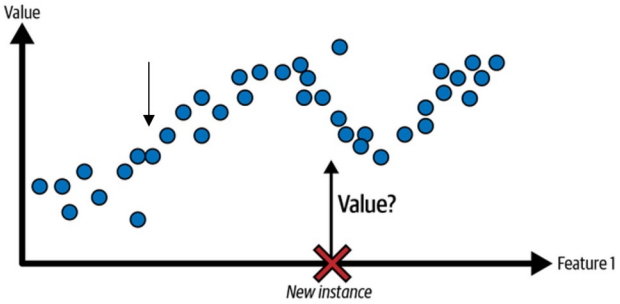
$y_m \in \{C_1, C_2, \ldots, C_K\}$



Regression

Target variable y: numerical - continuous

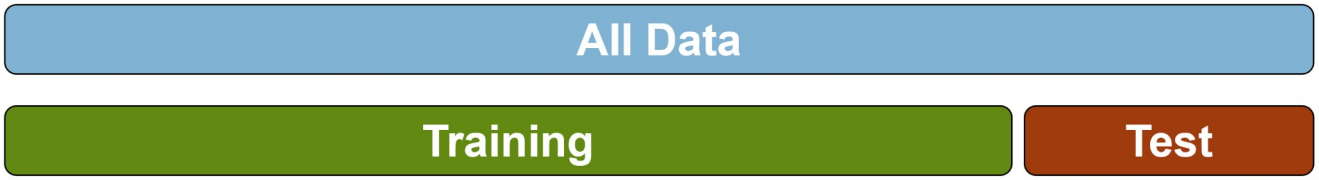
$y_m \in \mathbb{R}$



Classification vs. regression

1.6.2 Evaluating Supervised Machine Learning Models

The ultimate goal in supervised machine learning is to generate predictions for new input data samples beyond the data available for training. Therefore, the quality of a machine learning model is assessed in terms of its generalization ability or, in other words, the generalisation error. To calculate the generalisation error exactly is obviously impossible at the time when training the model. At best, it can be estimated by means of an independent test set that is separated from the training data before the training process has begun. Once the training is completed, the accuracy of the trained model is assessed on the independent test set. This mimicks the situation of the model being applied to new “unseen” data not included in the training set. Since the test set also consists of input values and associated expected output values, the output values predicted by the model can be compared with the expected output values and quality metrics, that quantitatively measure, how well the model’s predictions compare with the expected output values, can be calculated. For regression and classification problems different quality metrics are employed.

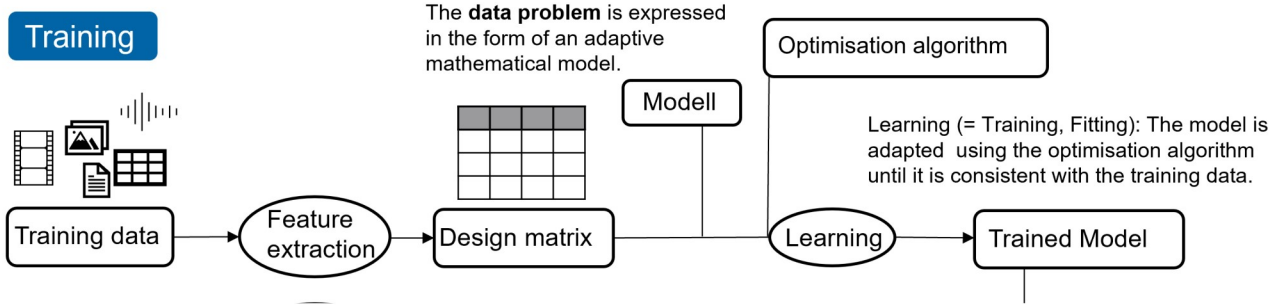


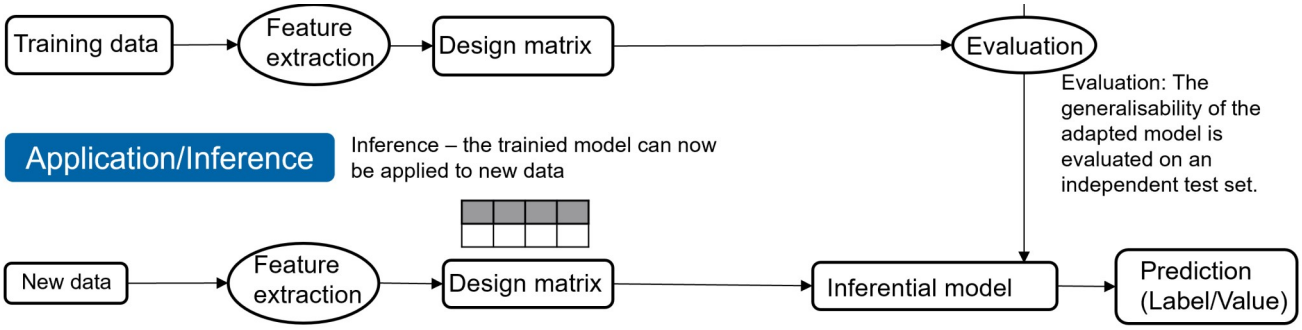
Training-test split of the data

The evaluation of unsupervised machine learning models is much more difficult due to the more complex nature of the problem.

1.6.3 Supervised Learning Pipeline

Given a problem (e.g. ham/spam classification) and training data a typical supervised learning pipeline starts with representing the data in a structured format (numerical design matrix). Then a model type is selected and trained (learned) on the available training data. More specifically, the model represents a family of functions determined by a set of parameters and these parameters are adjusted using an optimization algorithm (in an iterative process) to obtain a consistent mapping between the inputs and their corresponding outputs in the training set. The generalization ability of the final, adapted model (the function determined by the parameter values selected in the learning process) is quantified by computing the estimate of the generalization error on an independet set (unseen during the model training). The final model can then be applied in practice to predict (infer) the unknown output values for given inputs (e.g. is the email I just received spam or ham). In practice the process above is typically iterated several times, on the one hand, to obtain an ML model that gives predictions with sufficient quality to be useful, and on the other hand, to update a deployed model if its performance has dropped due to changes in the data.




Training data
Feature extraction
Design matrix
Evaluation
Evaluation: The generalisability of the adapted model is evaluated on an independent test set.
Application/Inference
Inference – the trainied model can now be applied to new data
New data
Feature extraction
Design matrix
Inferential model
Prediction (Label/Value)