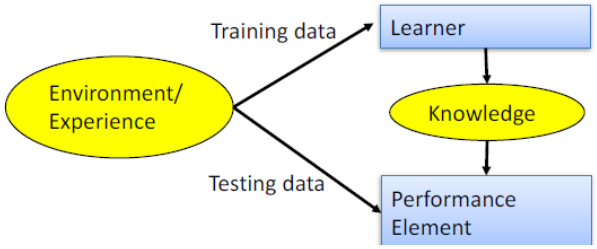
Lecture 7

Designing a Learning System

• Choose the training experience
• Choose exactly what is to be learned
– i.e. the ***target function*** ***(*** A **target function** maps data to its **target** value such as if you have picture of a digit, the **function** then receives that picture as the input and spit the digit value as the output. The process of finding that **function** is called learning or training.***)***
• Choose how to represent the target function
• Choose a learning algorithm to infer the target function from the experience



Machine Learning model

The names like *Linear Regression*, *Logistic Regression*, *Decision Trees* etc. are just the names of the algorithms. Those are just **theoretical concepts** that describe what to do in order to achieve the specific effect. Model is a **mathematical formula** which is a result of Machine Learning **algorithm implementation**. It has measurable parameters that can be used for prediction. Models can be trained by modifying their parameters in order to achieve better results. It is possible to say that models are representations of what a Machine Learning system has learned from the training data.

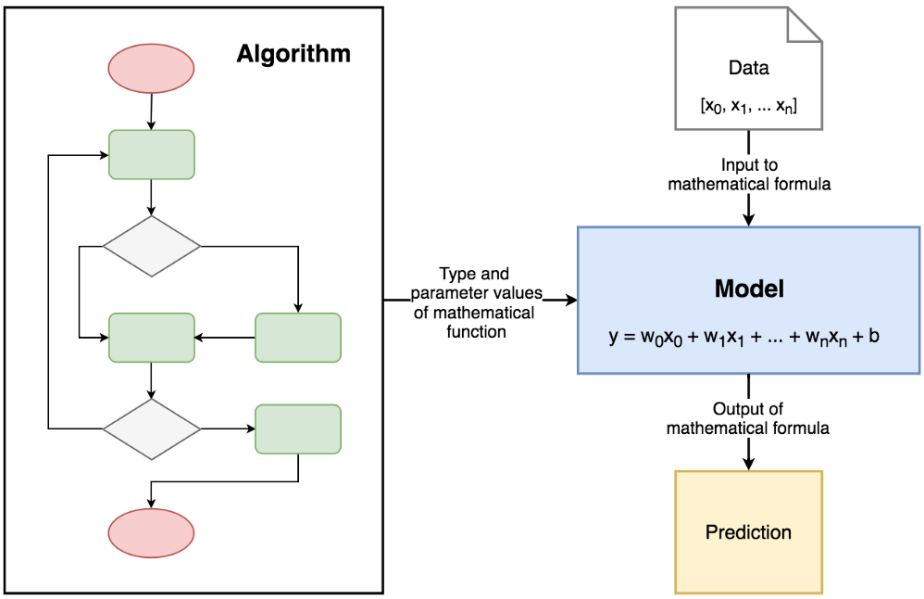
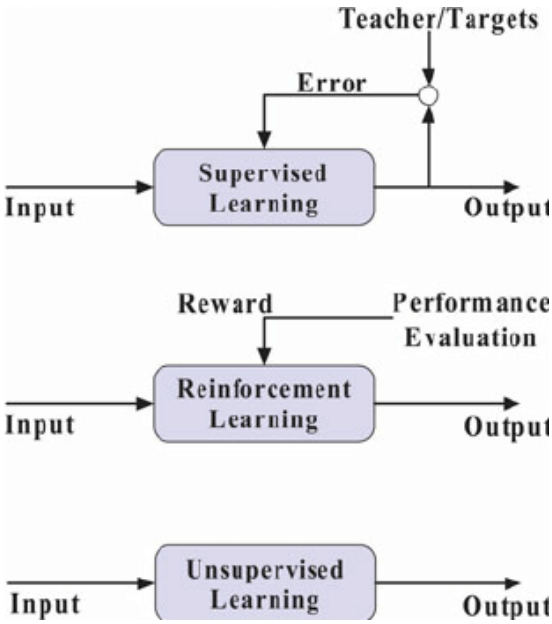


Diagram visualizing difference between Machine Learning Algorithm and Machine Learning Model.

Basic structures/model of the three learning paradigms: supervised learning, reinforcement learning, and unsupervised learning.



Basic learning algorithms

	Supervised Learning	Unsupervised Learning
Discrete	classification or categorization	clustering
Continuous	regression	dimensionality reduction

Supervised Learning Model

The group of algorithms that **require dataset which consists of example input-output pairs**. Each pair consists of **data sample** used to make prediction and expected outcome called **label**. Word "supervised" comes from a fact that labels need to be assigned to data by the **human supervisor**.

In training process, samples are being iteratively fed to the model. For every sample, the model uses the current state of parameters and returns a prediction. Prediction is compared to label, and the difference is called an error. **The error is a feedback for the model of what went wrong and how to update itself in order to decrease the error in future predictions.** This means that model will change the values of its parameters according to the algorithm based on which it was created.

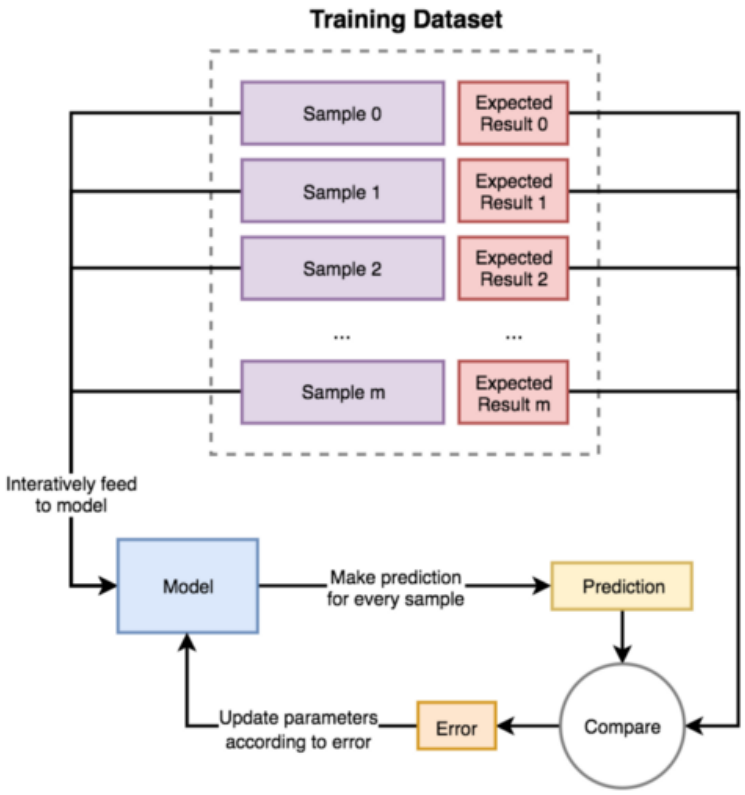


Diagram demonstrating how Supervised Learning works.

Supervised learning problems can be grouped into **Regression** and **Classification** problems. Both problems have as goal the construction of a succinct model that can predict the value of the dependent attribute from the attribute variables. The difference between the two tasks is the fact that the dependent attribute is numerical for regression and categorical for classification.

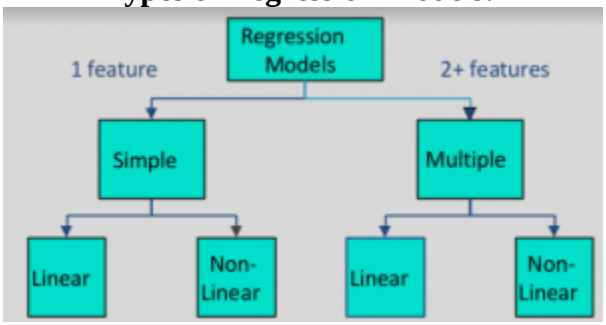
 Classification— A classification problem is when the output variable is a category, such as "red" or "blue" or "disease" and "no disease". A classification model attempts to draw some conclusion from observed values. Given one or more inputs a classification model will try to predict the value of one or more outcomes.

For example, when filtering emails “spam” or “not spam”, when looking at transaction data, “fraudulent”, or “authorized”. In short Classification either predicts categorical class labels or classifies data (construct a model) based on the training set and the values (class labels) in classifying attributes and uses it in classifying new data. There are a number of classification models. Classification models include logistic regression, decision tree, random forest, gradient-boosted tree, multilayer perceptron, one-vs-rest, and Naive Bayes. So classification is the process of *assigning category to input* data sample. Example usages: predicting whether a person is ill or not, detecting fraudulent transactions, face classifier.

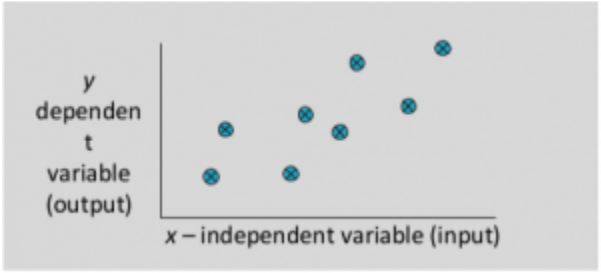
 Regression – A regression problem is when the output variable is a real or continuous value, such as “salary” or “weight”. Many different models can be used, the simplest is the linear regression. It tries to fit data with the best hyper-plane which goes through the points.

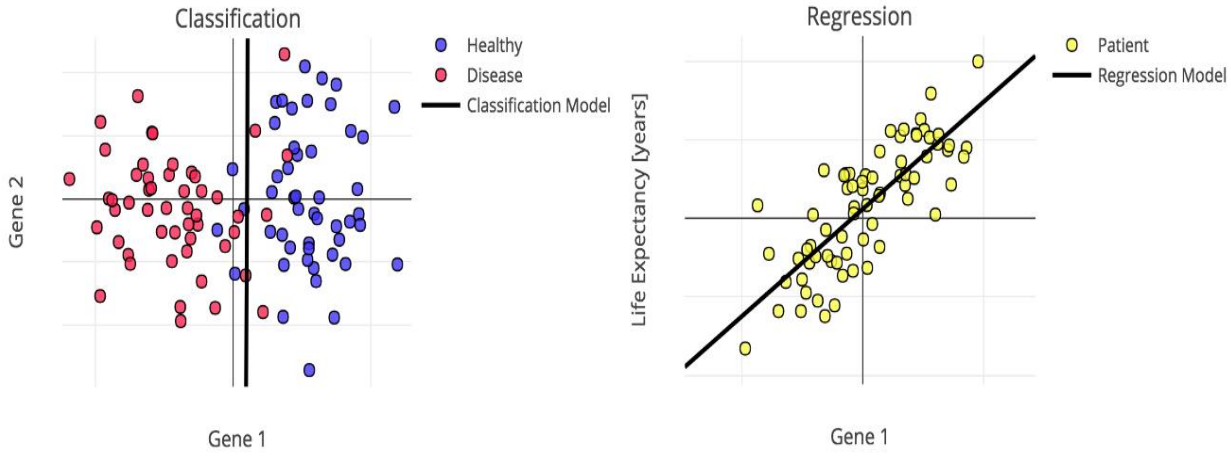
So it’s the process of *predicting a continuous, numerical value* for input data sample. Example usages: assessing the house price, forecasting grocery store food demand, temperature forecasting.

Types of Regression Models:



Supervised Learning models are **trying to find parameter values that will allow them to perform well on historical data**. Then they are **used for making predictions on unknown data**, that was not a part of training dataset.



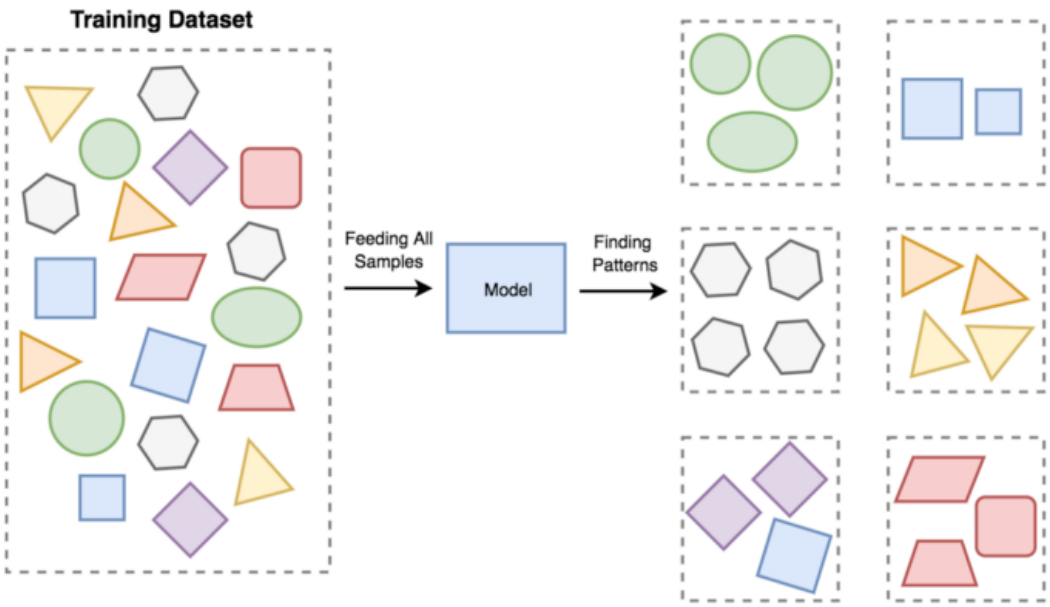


Example of Classification and Regression models.

Classification trees have dependent variables that are categorical and unordered. **Regression** trees have dependent variables that are continuous values or ordered whole values. **Regression** means to predict the output value using training data. **Classification** means to group the output into a class.

Unsupervised Learning Model

Group of algorithms that try to **draw inferences from non-labeled data** (without reference to known or labeled outcomes). In Unsupervised Learning, **there are no correct answers**. Models based on this type of algorithms can be used for discovering unknown data patterns and data structure itself.



Example of Unsupervised Learning concept. All data is fed to the model and it produces an output on it's own based on similarity between samples and algorithm used to create the model.

Two common use-cases for unsupervised learning are clustering and dimensionality reduction:

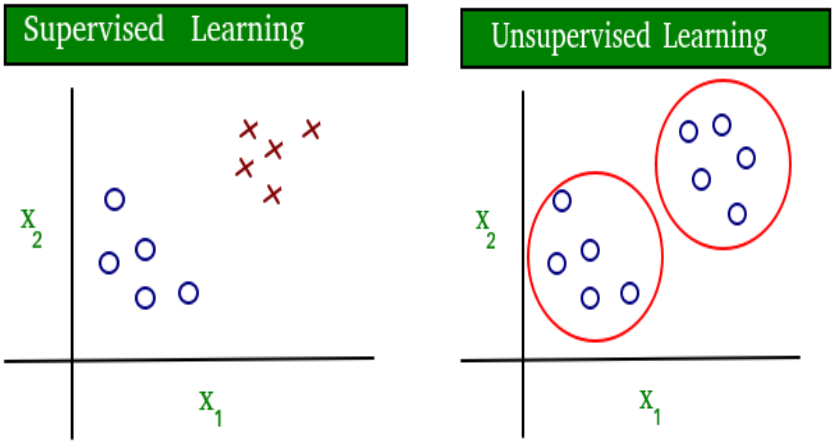
- **Clustering** is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups. It is basically a collection of objects on the basis of similarity and dissimilarity between them. It's the process of *dividing and grouping similar data* samples together. Groups are usually called clusters. Example usages: segmentation of supermarkets, user base segmentation, signal denoising.

  Applications of Clustering in different fields
  1. **Marketing :** It can be used to characterize & discover customer segments for marketing purposes.
  2. **Biology :** It can be used for classification among different species of plants and animals.
  3. **Libraries :** It is used in clustering different books on the basis of topics and information.
  4. **Insurance :** It is used to acknowledge the customers, their policies and identifying the frauds.
  5. **City Planning :** It is used to make groups of houses and to study their values based on their geographical locations and other factors present.
  6. **Earthquake studies :** By learning the earthquake affected areas we can determine the dangerous zones.

- **Reducing data dimensionality** - Data dimension is the number of features needed to describe data sample. Transform an initial representation of items into a lower-dimensional representation of these items while preserving some properties of the initial representation.

Dimensionality reduction is a *process of compressing features into so-called principal values* which conveys similar information concisely. By selecting only a few components, the amount of features is reduced and a small part of the data is lost in the process. Example usages: speeding up other Machine Learning algorithms by reducing numbers of calculations, finding a group of most reliable features in data.

A simple diagram which clears the concept of supervised and unsupervised learning is shown below:

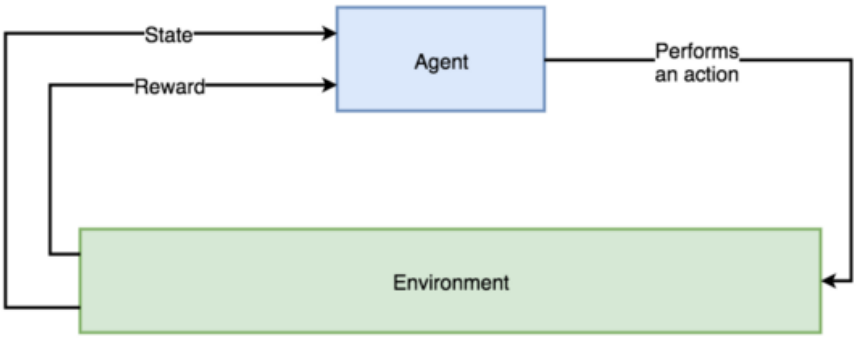


Reinforcement Learning Model

Branch of Machine Learning algorithms which produces so-called **agents**. The agent role is slightly different than classic model. It's to **receive information from the environment and react to it** by performing an **action**. The information is fed to an agent in form of numerical data, called **state,** which is stored and then used for choosing right action. As a result, an agent receives a **reward** that can be either positive or negative. The reward is a feedback that can be used by an agent to update its parameters.

Training of an agent is a process of **trial and error**. It needs to find itself in various situations and get punished every time it takes the wrong action in order to learn. The goal of optimization can be set in many ways depending on Reinforcement Learning approach e.g. based on *Value Function, Gradient Policy* or *Environment Model*.

* The value function represent how good is a state for an agent to be in. It is equal to expected total reward for an agent starting from states. The value function depends on the policy by which the agent picks actions to perform.



Interaction between Agent and Environment.