1 Introduction to Machine Learning

Contents

Abstract

In 1959, Arthur Samuel, whom created one of the world's first self-learning program, define machine learning as the "Field of study that gives computer the ability to learn *without* being explicitly programmed". In this chapter, we introduce the four[1] main forms of machine learning algorithms. We also outline the general ingredients for any form of machine learning algorithm: LEARNING = REPRESENTATION + EVALUATION + OPTIMISATION

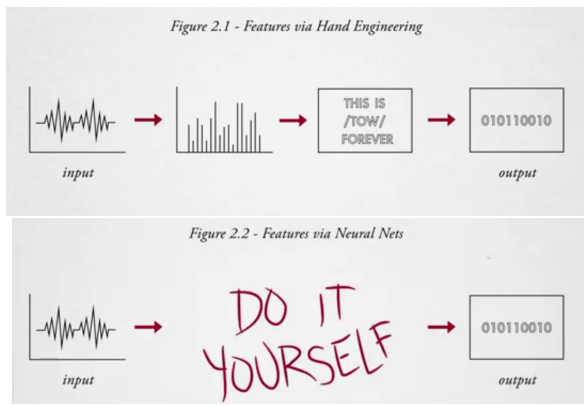

Figure 1.1: Explicitly programming the process (top) vs. machine learned engineering (bottom).

1.1 Machine learning algorithms

1.1.1 Supervised Learning

Definition 1.1. In **supervised learning** we teach the computer how to solve some problem, and then let it use its new found knowledge to solve similar problems. In particular, we choose a *model,* f, to map input, $\mathbf{x}$, to the *predicted* output, $y = f(\mathbf{x} : \boldsymbol{\theta})$, where $\boldsymbol{\theta}$:=set of parameters. We then adjust the parameters, $\boldsymbol{\theta}$, to reduce the difference between y and the *target* output, t.

Remark. There are two types of supervised learning:

1. **Regression** The target output is a *real number or a vector of real numbers*, and the aim is to *get as close as you can to this target.*
2. **Classification** The target output is a *(discrete) class label.*

Example. ***Regression vs. classification***

[1] The Coursera machine learning only discusses *supervised* and *unsupervised* learning, but I have included brief details on semi-supervised and reinforcement learning for completeness.

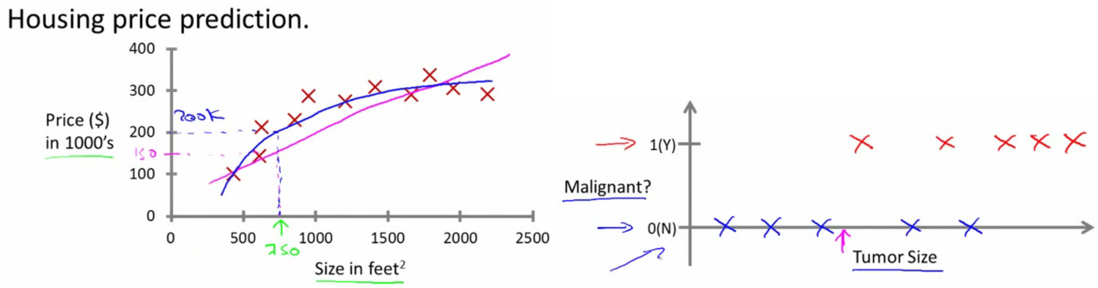

Figure 1.2: Left: ***regression*** *problem (**continuous valued output**)*: What is the best predictor of house price - straight or quadratic?
Right: ***classification*** *problem (**discrete valued output**)*: What is the likelihood of my friend with a tumour of size x having a malignant or benign cancer?

Example. ***Classification under multiple attributes***

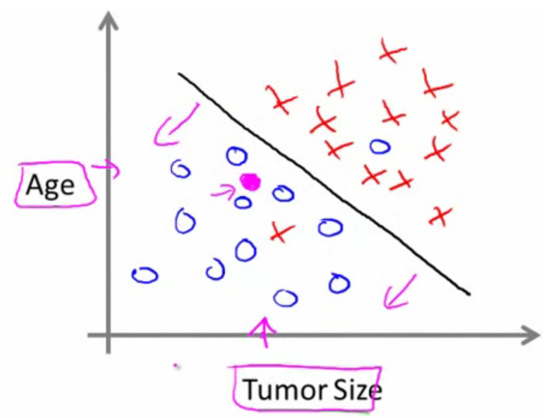

Figure 1.3: What is the likelihood of my friend with a tumour of size x **and** of age y having a malignant of benign cancer?

Note. We can deal with problems that have an infinite number of attributes (see support vector machine in chapter).

1.1.2 Unsupervised Learning

Definition 1.2. In **unsupervised** learning, we let the computer determine its own structure and patterns in the data.

Example 1.1. ***Clustering (*****relationships among the variables in the data*****)***

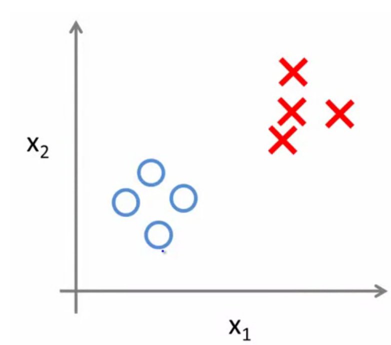

Figure 1.4: Given an **unknown** data set, can you find some structure to the data? E.g. Google News - given a set of new articles, group them into cohesive groups.

1.1.3 Semi-supervised Learning

Definition 1.3. **Semi-supervised** learning is a mixture of supervised and unsupervised learning, where we typically have only a small amount of labelled data, and a large amount of unlabelled data.

1.1.4 Reinforcement Learning (aka. trial and error learning)

Definition 1.4. In **reinforcement learning**, an agent learns from the consequences of its actions on some environment, and selects its actions on basis of its past experiences to maximise some numerical reward.

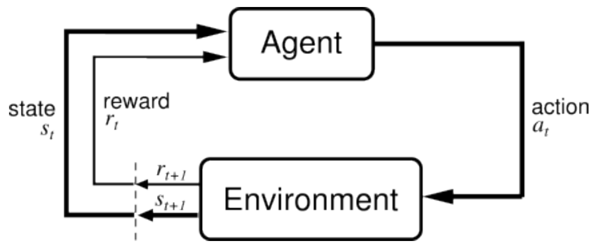

Figure 1.5: Reinforcement learning (RL) is learning by interacting with an environment to maximise some reward. An agent using the state of the system (which is identifies using observations) and the reward at time t, to choose some action at time t, which in turn updates the state of the environment and the reward value, at time $t + 1$.

Remark. The environment is typically "noisy" - you never have full information about what is going on.

Remark. This type of learning is typically seen in biology (c.f. evolution - in humans it is the dopamine system in our brains that implements reinforcement learning[2]).

Remark. The agent choose its actions using a balance of exploration (of uncharted territory) and exploitation (of current knowledge); randomly selecting actions, without reference to an estimated probability distribution, is known to give rise to very poor performance.

1.2 LEARNING = REPRESENTATION + EVALUATION + OPTIMISATION[3]

There are literally thousands of machine learning algorithms available, and hundreds more are published each year. The key to not getting lost in this huge space is to realise that it consists of combinations of just three components:

1. **Representation:** The model must be represented in some formal language that the computer can handle. In particular, how should we represent the inputs (or features) to be used, as well as the mapping from features to predicted output.
2. **Evaluation:** An evaluation function is needed to distinguish good mapping from inputs to outputs.
3. **Optimisation:** what learner best maps the inputs to the target output (i.e. which parameters minimises the cost function).

[2]`https://youtu.be/rbsqaJwpu6A?t=7m35s`
[3]Information obtained from `https://homes.cs.washington.edu/~pedrod/papers/cacm12.pdf`

Representation	Evaluation	Optimization
Instances	Accuracy/Error rate	Combinatorial optimization
K-nearest neighbor	Precision and recall	Greedy search
Support vector machines	Squared error	Beam search
Hyperplanes	Likelihood	Branch-and-bound
Naive Bayes	Posterior probability	Continuous optimization
Logistic regression	Information gain	Unconstrained
Decision trees	K-L divergence	Gradient descent
Sets of rules	Cost/Utility	Conjugate gradient
Propositional rules	Margin	Quasi-Newton methods
Logic programs		Constrained
Neural networks		Linear programming
Graphical models		Quadratic programming
Bayesian networks		
Conditional random fields		

Figure 1.6: Common examples of each of the 3 components of learning algorithms.