

Module 12

Machine Learning

Version 2 CSE IIT, Kharagpur

12.1 Instructional Objective

- The students should understand the concept of learning systems
- Students should learn about different aspects of a learning system
- Students should learn about taxonomy of learning systems
- Students should learn about different aspects of a learning systems like inductive bias and generalization
- The student should be familiar with the following learning algorithms, and should be able to code the algorithms
 - Concept learning
 - Decision trees
 - Neural networks
- Students understand the merits and demerits of these algorithms and the problem domain where they should be applied.

At the end of this lesson the student should be able to do the following:

- Represent a problem as a learning problem
- Apply a suitable learning algorithm to solve the problem.

Lesson 33

Learning : Introduction

12.1 Introduction to Learning

Machine Learning is the study of how to build computer systems that adapt and improve with experience. It is a subfield of Artificial Intelligence and intersects with cognitive science, information theory, and probability theory, among others.

Classical AI deals mainly with *deductive* reasoning, learning represents *inductive* reasoning. Deductive reasoning arrives at answers to queries relating to a particular situation starting from a set of general axioms, whereas inductive reasoning arrives at general axioms from a set of particular instances.

Classical AI often suffers from the knowledge acquisition problem in real life applications where obtaining and updating the knowledge base is costly and prone to errors. Machine learning serves to solve the knowledge acquisition bottleneck by obtaining the result from data by induction.

Machine learning is particularly attractive in several real life problem because of the following reasons:

- Some tasks cannot be defined well except by example
- Working environment of machines may not be known at design time
- Explicit knowledge encoding may be difficult and not available
- Environments change over time
- Biological systems learn

Recently, learning is widely used in a number of application areas including,

- Data mining and knowledge discovery
- Speech/image/video (pattern) recognition
- Adaptive control
- Autonomous vehicles/robots
- Decision support systems
- Bioinformatics
- WWW

Formally, 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 .

Thus a learning system is characterized by:

- task T
- experience E , and
- performance measure P

Examples:

Learning to play chess

T: Play chess

P: Percentage of games won in world tournament

E: Opportunity to play against self or other players

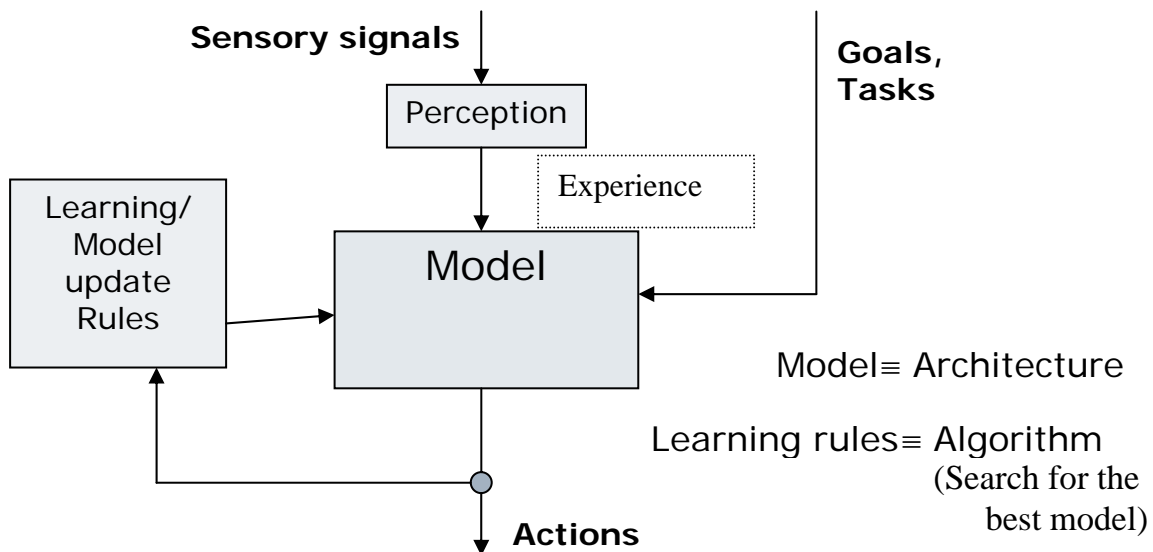
Learning to drive a van

T: Drive on a public highway using vision sensors

P: Average distance traveled before an error (according to human observer)

E: Sequence of images and steering actions recorded during human driving.

The block diagram of a generic learning system which can realize the above definition is shown below:



As can be seen from the above diagram the system consists of the following components:

- *Goal*: Defined with respect to the task to be performed by the system
- *Model*: A mathematical function which maps perception to actions
- *Learning rules*: Which update the model parameters with new experience such that the performance measures with respect to the goals is optimized
- *Experience*: A set of perception (and possibly the corresponding actions)

12.1.1 Taxonomy of Learning Systems

Several classification of learning systems are possible based on the above components as follows:

Goal/Task/Target Function:

Prediction: To predict the desired output for a given input based on previous input/output pairs. E.g., to predict the value of a stock given other inputs like market index, interest rates etc.

Categorization: To classify an object into one of several categories based on features of the object. E.g., a robotic vision system to categorize a machine part into one of the categories, spanner, hammer etc based on the parts' dimension and shape.

Clustering: To organize a group of objects into homogeneous segments. E.g., a satellite image analysis system which groups land areas into forest, urban and water body, for better utilization of natural resources.

Planning: To generate an optimal sequence of actions to solve a particular problem. E.g., an Unmanned Air Vehicle which plans its path to obtain a set of pictures and avoid enemy anti-aircraft guns.

Models:

- Propositional and FOL rules
- Decision trees
- Linear separators
- Neural networks
- Graphical models
- Temporal models like hidden Markov models

Learning Rules:

Learning rules are often tied up with the model of learning used. Some common rules are gradient descent, least square error, expectation maximization and margin maximization.

Experiences:

Learning algorithms use experiences in the form of perceptions or perception action pairs to improve their performance. The nature of experiences available varies with applications. Some common situations are described below.

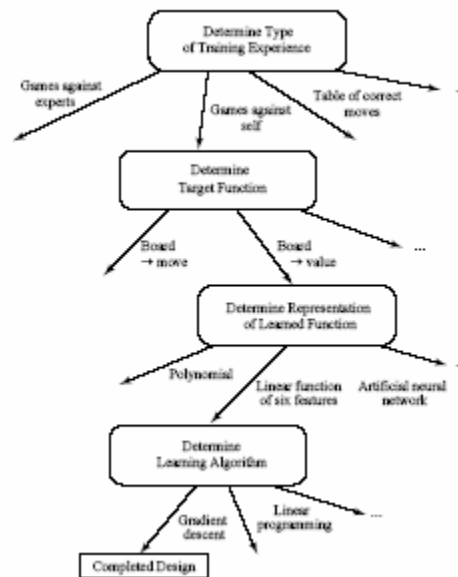
Supervised learning: In supervised learning a teacher or oracle is available which provides the desired action corresponding to a perception. A set of perception action pair provides what is called a training set. Examples include an automated vehicle where a set of vision inputs and the corresponding steering actions are available to the learner.

Unsupervised learning: In unsupervised learning no teacher is available. The learner only discovers persistent patterns in the data consisting of a collection of perceptions. This is also called exploratory learning. Finding out malicious network attacks from a sequence of anomalous data packets is an example of unsupervised learning.

Active learning: Here not only a teacher is available, the learner has the freedom to ask the teacher for suitable perception-action example pairs which will help the learner to improve its performance. Consider a news recommender system which tries to learn an users preferences and categorize news articles as interesting or uninteresting to the user. The system may present a particular article (of which it is not sure) to the user and ask whether it is interesting or not.

Reinforcement learning: In reinforcement learning a teacher is available, but the teacher instead of directly providing the desired action corresponding to a perception, return reward and punishment to the learner for its action corresponding to a perception. Examples include a robot in a unknown terrain where its get a punishment when its hits an obstacle and reward when it moves smoothly.

In order to design a learning system the designer has to make the following choices based on the application.



12.1.2 Mathematical formulation of the inductive learning problem

- Extrapolate from a given set of examples so that we can make accurate predictions about future examples.
- **Supervised versus Unsupervised learning**
 Want to learn an unknown function $f(\mathbf{x}) = \mathbf{y}$, where \mathbf{x} is an input example and \mathbf{y} is the desired output. Supervised learning implies we are given a set of (\mathbf{x}, \mathbf{y}) pairs by a "teacher." Unsupervised learning means we are only given the \mathbf{x} s. In either case, the goal is to estimate f .

Inductive Bias

- Inductive learning is an inherently conjectural process because any knowledge created by generalization from specific facts cannot be proven true; it can only be proven false. Hence, inductive inference is **falsity preserving**, not truth preserving.
- To generalize beyond the specific training examples, we need constraints or **biases** on what f is best. That is, learning can be viewed as searching the **Hypothesis Space** H of possible f functions.
- A bias allows us to choose one f over another one
- A completely unbiased inductive algorithm could only memorize the training examples and could not say anything more about other unseen examples.
- Two types of biases are commonly used in machine learning:
 - **Restricted Hypothesis Space Bias**
 Allow only certain types of f functions, not arbitrary ones

- **Preference Bias**
Define a metric for comparing f s so as to determine whether one is better than another

Inductive Learning Framework

- Raw input data from sensors are preprocessed to obtain a **feature vector**, \mathbf{x} , that adequately describes all of the relevant features for classifying examples.
- Each \mathbf{x} is a list of (attribute, value) pairs. For example,

$\mathbf{x} = (\text{Person} = \text{Sue}, \text{Eye-Color} = \text{Brown}, \text{Age} = \text{Young}, \text{Sex} = \text{Female})$

The number of attributes (also called features) is fixed (positive, finite). Each attribute has a fixed, finite number of possible values.

Each example can be interpreted as a *point* in an n -dimensional **feature space**, where n is the number of attributes.