UNIT - IV

LEARNING From Examples

Here we describe agents that can improve their behaviour through different

Forms of Learning

Any Component of an agent can be improved by learning from data.

The improvements & techniques used to make them depend on 4 major factors

- Which Component is to be improved
- What prior knowledge the agent already has
- What representation is used for data & Component
- What feedback is available to learn from.

Components to be learned

Components of these agents include:

1) A direct mapping from conditions on current state to actions.

2) A means to infer relevant properties of world from percept sequence

3) Information about the way the world evolves and about the results of possible actions the agent can take.

4) Utility information indicating desirability of world states

5) Action value information indicating the desirability of actions

6) Goals that describe classes of states whose achievement maximizes the agent's utility.

An agent training to become a taxi driver. Everytime instructor ~~bec~~ shouts "Brake!" the agent might learn a condition - action rule for when to brake (component 1).

the agent also learns every time the instructor doesn't shout.

By seeing many camera images that it is told contain buses, it can learn to recognize them (2).

By trying actions & observing the results - like braking hard on a wet road - it can learn the effects of its actions (3).

Then, when it receives no tip from passengers who have been thoroughly shaken up during the trip it can learn a useful component of its overall utility function (4).

Representation and prior knowledge.

We have seen propositional & first-order logical sentences for components in a logical agent; Bayesian n/ws for inferential components.

Here we cover i/ps that form a factored representation - a vector of attribute values - & o/ps that can be either a continuous numerical value or a discrete value.

the agent also learns every time the instructor doesn't shout.

By seeing many camera images that it is told contain buses, it can learn to recognize them (2).

By trying actions & observing the results - like braking hard on a wet road - it can learn the effects of its actions (3).

Then, when it receives no tip from passengers who have been thoroughly shaken up during the trip, it can learn a useful component of its overall utility function (4).

Representation and prior knowledge.

We have seen propositional & first-order logical sentences for components in a logical agent; Bayesian n/ws for inferential components.

Here we cover i/ps that form a factored representation - a vector of attribute values - & o/ps that can be either a continuous numerical value or a discrete value.

- There is another way to look at various types of learning.
- We say learning a general function or rule from specific i/p - o/p pairs is called inductive learning.
- Analytical or deductive learning - going from a known general rule to a new rule, i.e. logically entailed but useful b'coz it allows more efficient processing.

Feedback to learn from

- There are three types of feedback that determine three main types of learning:
- In unsupervised learning the agent learns patterns in the i/p even though no explicit feedback is supplied.
- Most common unsupervised learning task is clustering.
- Clustering: detecting potentially useful clusters of i/p examples.

A taxi agent might gradually develop a concept of "good traffic days" & "bad traffic days" without ever being given labeled examples of each by a teacher.

In reinforcement learning the agent learns from a series of reinforcements - rewards & punishments.

Ex:- * lack of a tip gives agent an indication that it did something wrong.

* two pts for a win at end of a chessgame tells agent it did something right.

So it is up to agent to decide which of the actions prior to reinforcement were most responsible for it.

In Supervised learning agent observes some example i/p - o/p pairs and learns a function that maps from i/p to o/p.

In component 1, the i/ps are percepts & o/p are provided by instructor who says "Brake!" or "Turn left"

Classification of Input D...
Techniques.

In component 2 i/p's are Camera images and o/p again come from a instructor who says "that

In 3 theory of braking is a function from st & braking actions to stopping distance in feet in this case. Here, o/p is available directly from agent percepts, the environment is instructor.

In Semi-Supervised learning

we are given a few labeled examples. & must make what we can of a large collection of unlabeled examples.

→ If we are trying to build a System to guess a person's age from a photo, we gather some labeled examples by Snapping pictures of people and asking their age. That's Supervised learning

→ But in reality Some people lied about their age. It's not just that there is a random noise in data. rather inaccuracies are Systematic & to uncover them is an unsupervised learning problem involving image

Self-reported ages & true ages.

So, both noise & lack of tables labels create a continuum b/w Supervised & unsupervised learning.

Supervised learning

Task:

Given a training set of N example i/p-o/p pairs.

$(x_1, y_1), (x_2, y_2) \dots (x_N, y_N)$

where each y_i was generated by an unknown function. $y = f(x)$, discover a function h that approximates the true function f.

- x & y can be any value, need not be numbers.
- function h is a hypothesis.
- learning is a search through the space of possible hypotheses for one that will perform well, even on new examples beyond the training set.
- To measure the accuracy of a hypothesis we give it a test set of examples that are distinct from training set.

- Sometimes the function f is stochastic – it is not strictly a function of x,
- we have to learn conditional probability distribution $P(y|x)$

- when the o/p y is one of a finite set of values (such as sunny, cloudy & rainy) the learning problem is called classification.
- when y is a ~~no~~ number (such as tomorrow's temperature) the learning problem is called regression.

-