SNS COLLEGE OF TECHNOLOGY

Coimbatore-35

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade

Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF AI&ML

FOUNDATIONS OF ARTIFICIAL INTELLIGENCE

II YEAR - III SEM

UNIT 5 – Inductive learning

Inductive learning

Need for Inductive Learning

- There are basically two methods for knowledge extraction firstly from domain experts and then with machine learning.
- For a very large amount of data, the domain experts are not very useful and reliable.
- So we move towards the machine learning approach for this work.

Inductive Learning

- Also called as Deterministic Supervised Learning
- In this first input x, (the verified value) given to a function f, and the output is f(x).
- Then we can give different set of inputs (raw inputs) to the same function f, and verify the output f(x).
- By using the outputs we generate (learn) the rules.

Inductive Learning

- Inductive learning, also known as discovery learning, is a process where the learner discovers rules by observing examples.
- We can often work out rules for ourselves by observing examples. If there is a pattern; then record it.
- We then apply the rule in different situations to see if it works.
- With inductive language learning, tasks are designed specifically to guide the learner and assist them in discovering a rule.

- Inductive learning: system tries to make a "general rule" from a set of observed instances.
- Example:
- Mango -> f(Mango) - > sweet (e1)
- Banana -> f(Banana) -> sweet (e2)
- ...
- Fruits -> f(Fruits) -> sweet (general rule)
- Supervised learning:
  - Learning algorithm is given the correct value of the function for particular inputs, (verified output)
  - Then changes its representation of the function to try to match the information provided by the feedback.

Example

- Suppose an example set having attributes - Place type, weather, location, decision and seven examples.
- Our task is to generate a set of rules that under what condition what is the decision.

EXAMPLE NO.	PLACE TYPE	WEATHER	LOCATION	DECISION
I)	hilly	winter	kullu	Yes
II)	mountain	windy	Mumbai	No
III)	mountain	windy	Shimla	Yes
IV)	beach	windy	Mumbai	No
V)	beach	warm	goa	Yes
VI)	beach	windy	goa	No
VII)	beach	warm	Shimla	Yes

- **finally we get the rule set :-**
- **Rule Set**
- **Rule 1:** IF the weather is warm THEN the decision is yes.
- **Rule 2:** IF place type is hilly THEN the decision is yes.
- **Rule 3:** IF location is Shimla THEN the decision is yes.
- **Rule 4:** IF location is Mumbai THEN the decision is no.
- **Rule 5:** IF place type is beach AND the weather is windy THEN the decision is no.

Text Book Example

- **example** is a pair *(x, f (x))*,
- where x is the input and *f(x)* is the output of the function applied to *x*.
- The task of **pure inductive inference** (or **induction)** is this:
  - Given a collection of examples of *f*, return a function *h* that approximates f .
  - Where The function h is called a **hypothesis.**
  - A good hypothesis will **generalize** well-that is, will predict unseen examples correctly.
- This is the fundamental **problem of induction.**

examples are *(x, f (x))* pairs...

- Fitting a function of a single variable to some data points.
- The examples are *(x, f (x))* pairs, where both x and f (x) are real numbers.
- We choose the **hypothesis space** H, which are the set of hypotheses we will consider,-to be the set of polynomials of degree at most ***k***.

examples are *(x, f (x))* pairs...

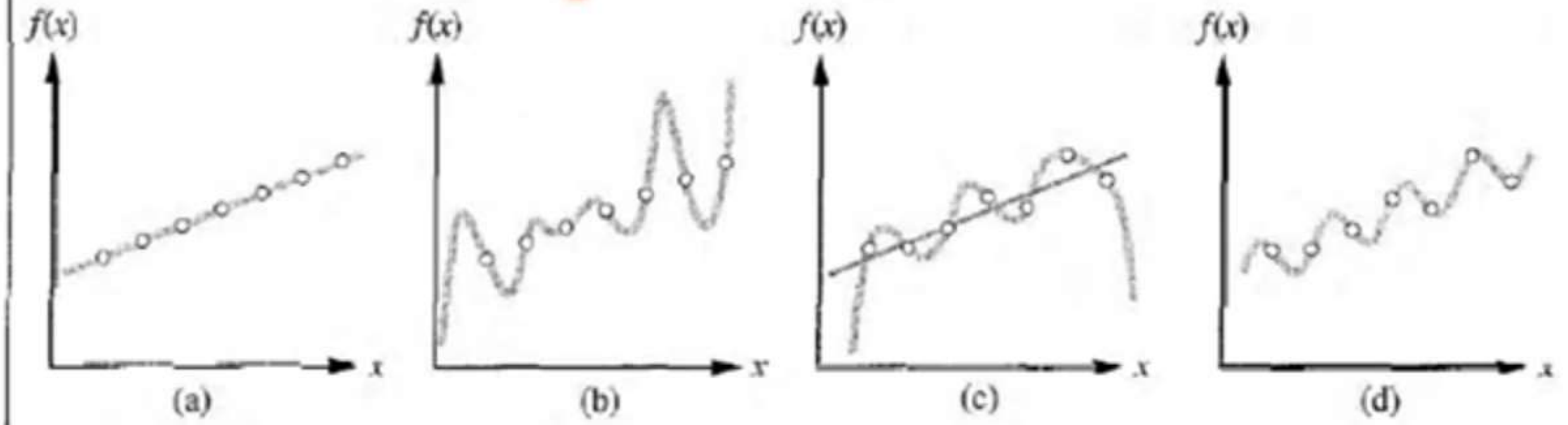

Figure 18.1 (a) Example (x, f (x)) pairs and a consistent, linear hypothesis. (b) A consistent, degree-7 polynomial hypothesis for the same data set. (c) A different data set that admits an exact degree-6 polynomial fit or an approximate linear fit. (d) A simple, exact sinusoidal fit to the same data set.

examples are *(x, f (x))* pairs...

- *There is a tradeoff between the expressiveness of a hypothesis space and the complexity of finding simple, consistent hypotheses within that space.*
- For example,
  - fitting straight lines to data is very easy;
  - fitting high-degree polynomials is harder; and
  - fitting Turing machines is very hard indeed because determining whether a given Turing machine is consistent with the data is not even decidable in general.
- Prefer a simple hypothesis spaces,
  - in which the resulting hypotheses may be simpler to use
  - It is faster to compute *h(x)* when h is a linear function than when it is an arbitrary Turing machine program.

T H A N K S