UNIT - V

5 Analytical Learning

5.1 : Introduction to Analytical Learning

[JNTU : Dec-17, Marks 2]

Q.1 What is analytical learning ?

Ans. : In analytical learning, the input to the learner includes the same hypothesis space H and training examples D as for inductive learning. In addition, the learner is provided an additional input: A **domain theory B** consisting of background knowledge that can be used to explain observed training examples. The desired output of ,the learner is a hypothesis h from H that is consistent with both the training examples D and the domain theory B.

Q.2 What is inductive learning ?

Ans. : In inductive learning, the learner is given a hypothesis space H from which it must select an output hypothesis, and a set of training examples $D = \{(x_1, f(x_1)). .,. (x_n, f(x_n))\}$ where $f(x_i)$ is the target value for the instance x_i. The desired output of the learner is a hypothesis h from H that is consistent with these training examples.

Q.3 What is the difference between inductive and analytical learning methods ?

Ans. :

Parameters	Inductive learning	Analytical learning
Goal	Hypothesis fits data	Hypothesis fits domain theory
Justification	Statistical inference	Deductive inference
Merit	Requires little prior knowledge	Learns from scarce data
Demerit	• Scarce data, • Incorrect bias	Imperfect domain theory

Q.4 What is domain theory ?

Ans. : A domain theory is said to be correct if each of its assertions is a truthful statement about the world. A domain theory is said to be complete with respect to a given target concept and instance space, if the domain theory covers every positive example in the instance space.

5.2 : Learning with Perfect Domain Theories : PROLOG-EBG

Q.5 What is prolog ?

Ans. : • Prolog is a logic programming language associated with artificial intelligence and computational linguistics.

- Prolog has its roots in first-order logic, a formal logic, and unlike many other programming languages, Prolog is intended primarily as a declarative programming language: the program logic is expressed in terms of relations, represented as facts and rules. A computation is initiated by running a query over these relations.

What are the main properties of PROLOG-EBG algorithm ? Is it deductive or inductive ? Justify your answer.

[JNTU : Dec-17, Marks 5]

s. :
- PROLOG-EBG is a sequential covering algorithm.
- PROLOG-EBG computes the most general rule that can be justified by the explanation, by computing the weakest pre-image of the explanation.
- PROLOG-EBG constructs intermediate features after analyzing examples.
- It is deductive learning system, which assume that domain knowledge is correct and complete.
- PROLOG-EBG produces justified general hypotheses by using prior knowledge to analyze individual examples
- PROLOG-EBG implicitly assumes that the domain theory is correct and complete. If the domain theory is incorrect or incomplete, the resulting learned concept may also be incorrect.
- The generality of the learned Horn clauses will depend on the formulation of the domain theory and on the sequence in which training examples are considered.
- In its pure form, PROLOG-EBG is a deductive, rather than inductive, learning process. That is, by calculating the weakest pre-image of the explanation it produces a hypothesis h that follows deductively from the domain theory B, while covering the training data D.

Discuss Prolog-EBG algorithm.

[JNTU : Dec-16, Marks 5]

s. : **PROWG-EBG(TargetConcept, TrainingExamples, DomainTheory)**

LearnedRules ← { }

Pos ← the positive examples from **TrainingExamples**

for each **PositiveExample** in Pos that is not covered by **LearnedRules**, do

1. **Explain** : **Explanation** ← an explanation (proof) in terms of the **DomainTheory** that **PositiveExample** satisfies the **TargetConcept**
2. **Analyze : SufficientConditions** ← the most general set of features of **PositiveExample** sufficient to satisfy the **TargetConcept** according to the **Explanation**.
3. **Rejine : LearnedRules** ← **LearnedRules** + **NewHornClause**, where **NewHornClause** is of the form

   TargetConcept ← **SufficientConditions**

   Return **LearnedRules**

5.3 : Remarks on Explanation Based Learning

What is explanation-based learning ?

Ans. :

- Explanation-based learning is a form of analytical learning in which the learner processes each novel training example by explaining the observed target value for this example in terms of the domain theory, analyzing this explanation to determine the general conditions under which the explanation holds refining its hypothesis to incorporate these general conditions.
- An Explanation-based Learning (EBL) system accepts an example (i.e. a training example) and explains what it learns from the example.
- The EBL system takes only the relevant aspects of the training. This explanation is translated into particular form that a problem solving program can understand. The explanation is generalized so that it can be used to solve other problems.
- The EBL module uses the results from the problem-solving trace (ie. Steps in solving problems) that were generated by the central problem solver.
- It constructs explanations using an axiomatized theory that describes both the domain and the architecture of the problem solver.
- The results are then translated as control rules and added to the knowledge base. The control knowledge that contains control rules is used to guide the search process effectively.

Q.9 List and explain Explanation-based Learning phases.

Ans. : EBL phases are as follows :

1. Problem solving

- From the example of the concept and the domain theory a solution that explains the concept is obtained
- From this resolution we are interested on all the actions performed
- These action will be the trace of the resolution, and will be used during the generalization process

2. Resolution trace analysis and filtering

- The domain determines the operational criteria that tells which are the primitive actions for the problem
- The relevance criteria will also be defined , this will allow to decide what parts of the resolution are important
- Using these two criteria the parts that need to be generalized from the resolution trace will be determined
- The filtered resolution trace will be the explanation of the example.

3. Generalization of the explanation

- The generalization of the explanation requires the substitution of constants by variables in a way that preserves the original explanation
- The usual mechanism for the generalization id the goal regression algorithm
- This algorithm consists on the variabilization of the goal and the propagation of the substitution in the explanation

4. Building the new knowledge

- The explanation has to be expressed using the primitive predicates of the domain
- The knowledge has to be translated to the representation formalism of the domain theory
- This knowledge can be new definition of the domain predicates or control rules that represent how the knowledge has to be used to solve new problems

5. Incorporating the new knowledge

- Sometimes is not enough to add the knowledge to the domain theory
- If we do not want the theory to degrade, the new knowledge has to be transformed so it can be used efficiently
- Their use can also be evaluated so it can be eliminated if it is not used frequently enough

Q.10 Explain elements of Explanation-based Learning.

Ans. : EBL elements are as follows :

1. Domain theory: Information about the specificc domain of the problem
2. Goal concept: Concept we want to obtain an operational definition

3. Example : Positive example of the concept we want to learn
4. New domain Theory : The initial theory plus the new definition learned for the goal concept from the example

Q.11 Explanation determines feature relevance." Substantiate this statement with respect to explanation based learning. **[JNTU : Dec-16, Marks 5]**

Ans. :

- Choosing good features to represent objects can be crucial to the success of supervised machine learning algorithms
- Explanation-based learning (EBL) is a method of dynamically incorporating prior domain knowledge into the learning process by explaining training examples.
- In classical EBL, an "explanation" is a logical proof that shows how the class label of a particular labeled example can be derived from the observed inputs
- Unlike inductive methods, PROLOG-EBG produces justified general hypotheses by using prior knowledge to analyze individual examples.
- The explanation of how the example satisfies the target concept determines which example attributes are relevant, those mentioned by the explanation.
- The further analysis of the explanation, regressing the target concept to determine its weakest pre-image with respect to the explanation, allows deriving more general constraints on the values of the relevant features

Q.12 What is knowledge level learning ? Explain.

Ans. :

- The knowledge level is a level of description for computer systems
- Example of knowledge-level analytical learning is provided by considering a type of assertions known as determinations.
- Determinations assert that some attribute of the instance is fully determined by certain other attributes, without specifying the exact nature of the dependence.
- For example, consider learning the target concept "people who speak Hindi," and imagine we are given as a domain theory the single determination assertion "the language spoken by a person is determined by their nationality."
- Taken alone, this domain theory does not enable us to classify any instances as positive or negative.
- However, if we observe that "Jon, a 23- year-old left-handed US, speaks Hindi," then we can conclude from this positive example and the domain theory that "all US speak Hindi."

5.4 : Using Prior Knowledge to Alter the Search Objective

Q.13 What is prior knowledge ?

Ans. : Prior knowledge refers to all information about the problem available in addition to the training data.

Q.14 Describe TANGENT PROP algorithm ?

Ans. :

- Tangent Propagation is the name of a learning technique of an artificial neural network (ANN) which enforces soft constraints on first order partial derivatives of the output vector.
- It accommodates domain knowledge expressed as derivatives of the target function with respect to transformations of its inputs.

- The TANGENTPROP algorithm assumes various training derivatives of the target function are also provided.
- For example, if each instance xi is described by a single real value, then each training example may be of the form $\left(x_i, f(x_i), \left.\frac{\partial f(x)}{\partial x}\right|_{x_i} \right)$.
- Here $\left.\frac{\partial f(x)}{\partial x}\right|_{x_i}$ denotes the derivative of the target function f with respect to x evaluated at the point $x = x_i$.
- To develop an intuition for the benefits of providing training derivatives as well as training values during learning, consider the simple learning task depicted in Fig. Q.14.1.

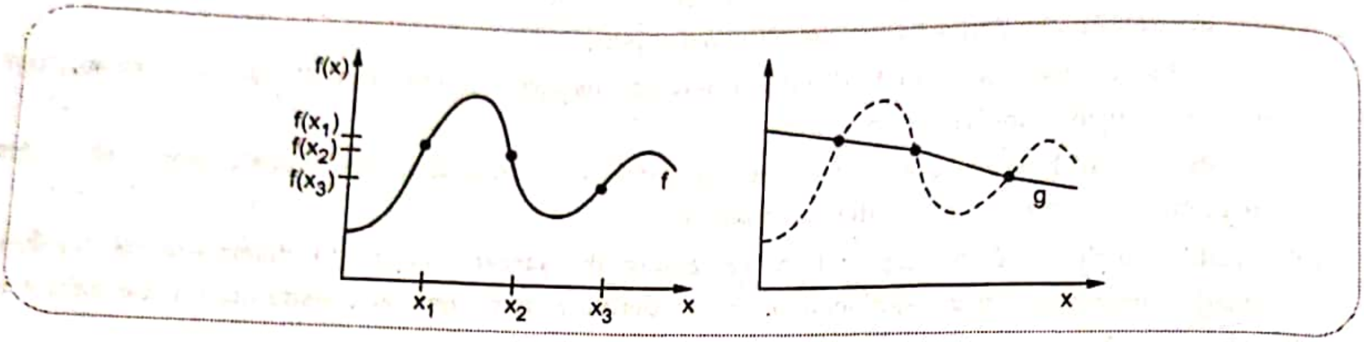

Fig. Q.14.1.

- The task is to learn the target function f shown in the leftmost plot of the figure, based on the three training examples shown: $(x_1, f(x_1))$, $(x_2, f(x_2))$, and $(x_3, f(x_3))$.
- Given these three training examples, the BACKPROPAGATATION algorithm can be expected to hypothesize a smooth function, such as the function g depicted in the middle plot of the figure.
- In TANGENTPROP an additional term is added to the error function to penalize discrepancies between the trainin4 derivatives and the actual derivatives of the learned neural network function f.

5.5 : Using Prior Knowledge to Augment Search Operators

Q.15 What is difference between first-order inductive learner (FOIL) and First Order Combined Learner (FOCL) ?

Ans. : FOIL generates each candidate specialization by adding a single new literal to the clause preconditions. FOCL uses this same method for producing candidate specializations, but also generates additional specializations based on the domain theory.

.16 What is FOCL ? Explain in detail.

ns. :

- FOCL uses the domain theory to increase the number of candidate specializations considered at each step of the search for a single Horn clause.
- FOCL expands its current hypothesis h using the following two operators:

1. For each operational literal that is not part of h, create a specialization of h by adding this single literal to the preconditions.

2. Create an operational, logically sufficient condition for the target concept according to the domain theory. Add this set of literals to the current preconditions of h.

- FOCL first selects one of the domain theory clauses whose head (postcondition) matches the target concept.
- If there are several such clauses, it selects the clause whose body (preconditions) have the highest information gain relative to the training examples of the target concept.
- FOCL learns Horn clauses of the form (c O_i ^ O_b^ O_f)

  where c is the target concept,

  O_i is an initial conjunction of operational literals added one at a time by the first syntactic operator,

  O_b is a conjunction of operational literals added in a single step based on the domain theory, and

  O_f is a final conjunction of operational literals added one at a time by the first syntactic operator.
- Any of these three sets of literals may be empty.
- FOCL uses both a syntactic generation of candidate specializations and a domain theory driven generation of candidate specializations at each step in the search.
- The algorithm chooses among these candidates based solely on their empirical support over the training data.
- Thus, the domain theory is used in a fashion that biases the learner, but leaves final search choices to be made based on performance over the training data.

5.6 : Combining Inductive and Analytical Learning

Q.17 Write specific properties of a learning method.

Ans. :

Properties include:

a. Given no domain theory, it should learn at least as effectively as purely inductive methods.

b. Given a perfect domain theory, it should learn at least as effectively as purely analytical methods.

c. Given an imperfect domain theory and imperfect training data, it should combine the two to outperform either purely inductive or purely analytical methods.

d. It should accommodate an unknown level of error in the training data.

e. It should accommodate an unknown level of error in the domain theory.

5.7 : Using Prior Knowledge to Initialize the Hypothesis

Q.18 Write KBANN algorithm to explain usage of prior knowledge to reduce complexity.

[JNTU : Dec-17, Marks 5]

Ans. :

- KBANN(Domain-Theory, Training_Examples)
- Domain-Theory: Set of propositional, nonrecursive Horn clauses.
- Training example: Set of (input output) pairs of the target function.
- Analytical step: Create an initial network equivalent to the domain theory.

1. For each instance attribute create a network input.
2. For each Horn clause in the Domain-Theory, create a network unit as follows:
   - Connect the inputs of this unit to the attributes tested by the clause antecedents.
   - For each non-negated antecedent of the clause, assign a weight of W to the corresponding sigmoid unit input.
   - For each negated antecedent of the clause, assign a weight of - W to the corresponding sigmoid unit input.
   - Set the threshold weight wo for this unit to -(n - 0 .5)W, where n is the number of non-negated antecedents of the clause.
3. Add additional connections among the network units, connecting each network unit at depth i from the input layer to all network units at depth i + 1. Assign random near-zero weights to these additional connections.