SA:

ML

Illustrate find S algorithm for EnjoySport training instances given:-

Ans:- • The FIND-S algo illustrates one way in which the more general than partial ordering can be used to organize the search for an acceptable hypothesis.

• The search moves from hypothesis to hypothesis, searching from the most specific to progressively more general hypothesis along one chain of partial ordering.

• At each stage, the hypothesis is the most specific hypothesis consistent with the training exs observed upto that point (hence name FIND-S).

The algorithm.

1. Initialize h to the most specific hypothesis in H
2. For each +ve training instance x
   - For each attribute contraint a_i in h,
     If a_i is satisfied by x, then do nothing
     Else
     replace ai in h by next more general constraint satisfied by x.
3. Output hypothesis h.

Step 1:- Initialize ....

$$h_0 \Leftarrow \{<\phi, \phi, \phi, \phi, \phi, \phi>\}$$

Step 2: None of "ϕ" constraints in 'h' are satisfied, so each is replaced by next more general constraint that fits this example.

$$x_1 = <S, W, N, S, W, S>, +$$

$$h_1 \Leftarrow \{<S, W, N, S, W, S>\}$$

Step 3:

$$x_2 = \{<S\ W\ H, S\ W\ S>\}, +$$

$$h_2 = <S\ W\ ?\ S\ W\ S>$$

Step 3:-

$$x_3 = <R\ C\ H\ S\ W\ C>, -$$

$$h_3 = <S\ W\ ?\ S\ W\ S>$$

Step 4:-

$$x_4 = <S\ W\ H\ S\ C\ C>, +$$

$$\boxed{h_4 = <S\ W\ ?\ S\ ?\ ?>}$$

Candidate Elimination Algo

Consistent hypothesis:- A hypothesis 'h' is said to be consistent with a set of traing examples 'D' iff $h(x) = c(x)$ for every example $(x, c(x))$ in D.

$$\text{Consistent}(h, D) \equiv \forall (x, c(x)) \in D,\ h(x) = c(x)$$

Version space:- Version space represented as $VS_{H,D}$ with respect to hypothesis space 'H' and training example 'D', is the subset of hypothesis from H consistent with the training examples in 'D':

$$VS_{H,D} \equiv \{h \in H \mid Consistent(h, D)\}$$

Candidate Elimination Algo

1. a) Initialise G to set of maximally general hypothesis in H

b) Initialise S to set of maximally specific hypothesis in H

2. For every training example 'd', do

(a) If d is +ve,

- Remove from G any hypothesis that is inconsistent with d.
- For every hypothesis s in S that is inconsistent with 'd'
  - Remove s from S
  - Add to S all minimal generalisations h of s such that h is consistent with d & some member of G is more general than h
  - Remove from S any hypothesis that is more general than another hypothesis in S.

(b) If d is -ve,

ulta of (a)

CE Algo for enjoy sport data set:-

Step 1:- Initialisations

$S_0 = \{\langle \phi, \phi, \phi, \phi, \phi, \phi \rangle\}$

$G_0 = \{\langle ?, ?, ?, ?, ?, ? \rangle\}$

Step 2:-

$S_1 = \{\langle S, W, N, S, W, S \rangle\}$

G_ϕ = same.

Step 3:- $S_2 = \{\langle S, W, ?, S, W, S \rangle\}$

G_2 = same

Step 4:- S_3 = same

$G_3 = \{\langle S, ?, ?, ?, ?, ? \rangle, \langle ?, W, ?, ?, ?, ? \rangle, \langle ?, ?, ?, ?, ?, S \rangle\}$

Step 5:- $S_4 = \{\langle S, W, ?, S, ?, ? \rangle\}$

$G_4 = \{\langle S, ?, ?, ?, ?, ? \rangle, \langle ?, W, ?, ?, ?, ? \rangle\}$

$\boxed{S_4 : \{\langle S, W, ?, S, ?, ? \rangle\}}$

$\langle S, W, ?, ?, ?, ? \rangle$, $\langle S, ?, ?, S, ?, ? \rangle$, $\langle ?, W, ?, S, ?, ? \rangle$

$\boxed{G_4 : \{\langle S, ?, ?, ?, ?, ? \rangle, \langle ?, W, ?, ?, ?, ? \rangle\}}$

Version space of Enjoy Sport.

The boundary sets S_4 & G_4 will delimit the version space of all the hypothesis that is consistent with the set of incrementally observed training examples. This learned version space is independent of the sequence in which the training examples are presented. As further the training data is encountered, the S & G boundaries will move monotonically closer to each other, delimiting a smaller & ~~s~~ smaller version space of candidate hypothesis.

ID3

ID3 algo is a greedy algo in which tree grows from top to bottom. At each node, it selects the attribute that best classifies the training examples. This process repeats until all the attributes have been used or until the tree perfectly classifies all the training examples.

ALGORITHM.

ID3 (Examples, Target_attribute, Attributes)

Examples → Training examples.

Target_attribute → Attribute whose value is to be predicted by the tree.

Attributes → List of other attributes that maybe tested by the learned decision tree.

Returns ~~Attributes~~ a decision tree that classifies the given examples correctly.

1. Create the root node for the tree.
2. If all Examples are +ve, return the single-node tree Root, with label = +.
3. If all Examples are –ve, return the single-node tree Root, with label = –
4. If Attributes is empty, return the single-node tree Root, with label = most common value of Target_attribute in Examples.
5. Otherwise Begin
    - A ← the attribute in Attributes that best classifies Examples.
    - The decision attribute for Root ← A
    - For each possible value v_i of A,
        - Add a new branch tree below Root corresponding to the test $A = v_i$.
        - Let $Examples_{v_i}$ be the subset of Examples that have the value v_i for A.
        - If $Examples_{v_i}$ is empty,
            - Then add a leaf node below this new branch, with label = most common value of Target_attribute in Examples.
            - Else below this new branch, add a subtree, ID3 ($Examples_{v_i}$, target_attribute, Attributes – {A})
6. End
7. Return Root

	A1	A2	
1	T	T	+
2	T	T	+
3	T	F	−
4	F	F	+
5	F	T	−
6	F	T	−

$\boxed{P = 3, n = 3}$

$+ = 6$

$$E(S) = \frac{-P}{p+n}\log_2\left(\frac{P}{p+n}\right) - \frac{n}{p+n}\log_2\left(\frac{n}{p+n}\right)$$

$$= \frac{-3}{6}\log_2\left(\frac{3}{6}\right) - \frac{3}{6}\log_2\left(\frac{3}{6}\right)$$

$$= -\frac{3}{6}\left(\log_2\frac{1}{2} + \log_2\frac{1}{2}\right)$$

$$(-1 - 1)$$

$$= \frac{-\cancel{3}^{1}}{\cancel{6}_{2}}(-2) \quad \frac{-1}{\cancel{2}} \times -\cancel{2} = 1$$

$$\boxed{E(S) = 1}$$

E(A1) = E

A1	
T	+
T	+
T	−

A1 → True
A1 → False

A1	
F	+
F	−
F	−

A1	p	n	Entropy
True	2	1	0.91823
False	1	2	0.91823

$$E(A1 = \text{True}) = -\frac{2}{3}\log_2\left(\frac{2}{3}\right) - \frac{1}{3}\log_2\left(\frac{1}{3}\right)$$

$$= -\frac{1}{3}\left[(2 \times -0.5849) + (-1.5849)\right]$$

$$= -\frac{1}{3}\left[-2.7547\right]$$

$$= \boxed{0.91823}$$

$$I(A1) = \frac{p_T + n_T}{p+n} E(A1 = T) + \frac{p_F + n_F}{p+n} E(A1 = F)$$

$$= \frac{2+1}{6}(0.91823) + \frac{1+2}{6}(0.91823)$$

$$= \left(\frac{1}{\cancel{2}}(0.91823)\right) \times \cancel{2} = \boxed{0.91823}$$

$$G_{A1} = S - I(A1)$$

$$= 1 - 0.91823 = \boxed{0.0818}$$

A2 < True / False

A2		A2	
T	+	F	+
T	+	F	−
T	−		
T	−		

A2	n	p	Entropy
T	2	2	1
F	1	1	1

$$I(A2) = \frac{p_T + n_T}{p+n} E(A1 = T) + \frac{p_F + n_F}{p+n} E(A1 = F)$$

$$= \frac{2+2}{6}(1) + \frac{1+1}{6}(1)$$

$$= \frac{4}{6} + \frac{2}{6} = \frac{6}{6} = 1$$

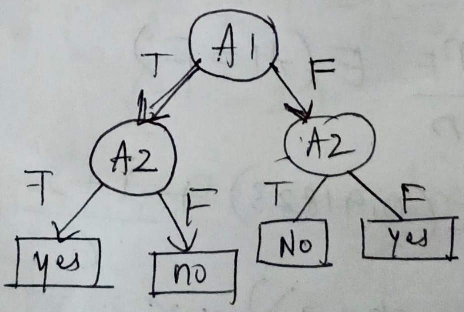
$$G(A2) = S - I(A2) = 1 - 1.00$$

$$= 0$$

$G_{A1} = 0.08$, $G_{A2} = 0.00$

$\Rightarrow$ $\boxed{\text{Root node} = A1}$

Decision Tree



Candidate Elim algo. $\rightarrow$ MRI scans data set.

Step 1:- $S_0 = \{<\phi, \phi, \phi, \phi, \phi>\}$, $G_0 = \{<?\ ?\ ?\ ?\ ?>\}$.

Step 2:- $x_1 = \{C, L, Li, S, Th\}, +$

$S_1 = \{<C, L, Li, S, Th>\}$

$G_1 = G_0$.

Step 3:- $x_2 = \{C, L, Li, I, Th\}, +$

$S_2 = \{<C, L, Li, ?, Th>\}$

$G_2 = G_1$.

Step 4:- $x_3 = \{O, L, D, S, T\}, -$

$S_3 = S_2$

$G_3 = \{<C, ?, ?, ?, ?>, <?\ ?\ L\ ?\ ?>, <?\ ?\ ?\ ?\ Th>\}$

contradiction

Step 5:- $x_4 = \{O, L, Li, I, Th\}, +$

$S_4 = \{<?, L, Li, ?, Th>\}$

$G_4 = \{<?\ ?\ Li\ ?\ ?>, <?\ ?\ ?\ ?\ Th>\}$

Step 6:- $x_5 = \{C, S, Li, S, Th\}, -$

$S_5 = \{<?, L, Li, ?, Th>\}$

$G_5 =$ if we consider $(x_5, -)$, we get G_5 a null set i.e., with one attribute, So, we consider 2 attributes for classifying +ve & -ve.

$G_5 = \{ \langle ?, \text{Large}, \text{Light}, ?, ? \rangle, \langle ?, \text{Large}, ?, ?, \text{thick} \rangle \}$

Version space :-

$\boxed{S_5 : \{ \langle ?, \text{Large}, \text{Light}, ?, \text{thick} \rangle \}}$

$\boxed{G_5 = \{ \langle ?, L, Li, ?, ? \rangle, \langle ?, L, ?, ?, Th \rangle \}}$