Advantages of Inductive bias :

⇒ Inductive bias provides a nonprocedural means of characterizing their policy for generalizing beyond the observed data.

⇒ It allows comparison of different learners according to the strength of the inductive bias they employ.

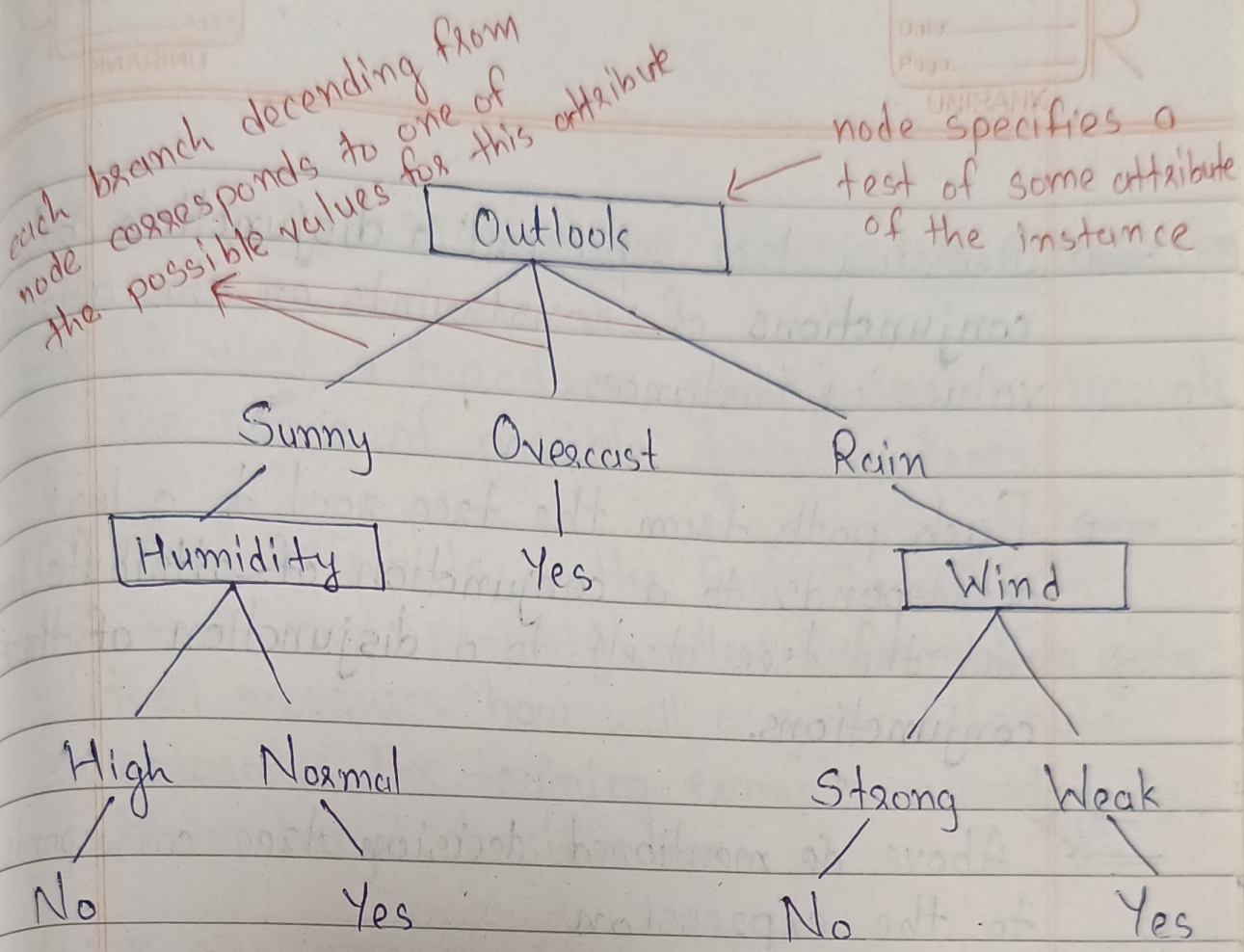
Module 2: Supervised Learning

✯ Decision Tree Learning:

⇒ Decision Tree learning is one of the most widely used and practical methods for inductive inference.

⇒ It is a method for approximating discrete valued functions that is robust to noisy data and capable of learning disjuctive expressions.

⇒ Decision Tree Learning is a method for approximating discrete-valued target functions, in which the learned ~~as~~ function is represented by a decision tree.



⇒ Above tree classifies Saturday mornings according to whether or not they are suitable for playing tennis.

Ex// Instance = < Outlook = Sunny, Temperature = Hot, Humidity = High, Wind = Strong >

⇒ above instance would be sorted down the leftmost branch of this decision tree and would therefore be classified as a negative instance.

⟹ Decision tree represent a disjunction of conjunctions of constraints on the attribute values of instances.

⟹ Each path from the tree root to a leaf corresponds to a conjunction of attribute tests and the tree itself to a disjunction of these conjunctions.

⟹ Above ~~de~~ mentioned decision tree corresponds to the expression

$$(Outlook = Sunny \wedge Humidity = Normal)$$
$$\vee$$
$$(Outlook = Overcast)$$
$$\vee$$
$$(Outlook = Rain \wedge Wind = Weak)$$

✱ When to use Decision Tree Learning?

1. Instances are represented by attribute-value pairs.
2. The target function has discrete output values. ex (Yes or No)
3. Disjunctive descriptions may be required.
4. The training data may contain errors.
5. The training data may contain missing attribute values

BASIC DECISION TREE LEARNING ALGORITHM

⇒ It used a top-down, greedy search through the space of possible decision trees.

(a) Which attribute is the Best Classifier?

⇒ Statistical property called **Information gain**, that measures how well a given attribute separates the training examples according to their target classification.

(i) Entropy measures homogeneity of Examples:-

boolean classification → $Entropy(S) = -p_{\oplus} \log_2 p_{\oplus} - p_{\ominus} \log_2 p_{\ominus}$

multi class classification → $Entropy(S) = \sum_{i=1}^{c} -p_i \log_2 p_i$

where;

S = collection of examples containing positive and negative examples of some target concept.

Ex: S = collection of 14 examples of some boolean concept, including 9 positive and 5 negative examples.

Note:- Entropy is considered as a measure of the impurity in a collection of training examples.

Note:- ⇒ Entropy is 0 if all members of S belong to the same class.

⇒ Entropy is 1 if collection contains an equal number of positive and negative examples.

$$Entropy(S) = -(9/14)\log_2(9/14) - (5/14)\log_2(5/14)$$

$$= 0.940$$

(ii) Information gain measures the expected reduction in Entropy :-

⟹ Information gain for this is simply the expected reduction in entropy caused by partitioning the examples according to this attribute.

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

where;

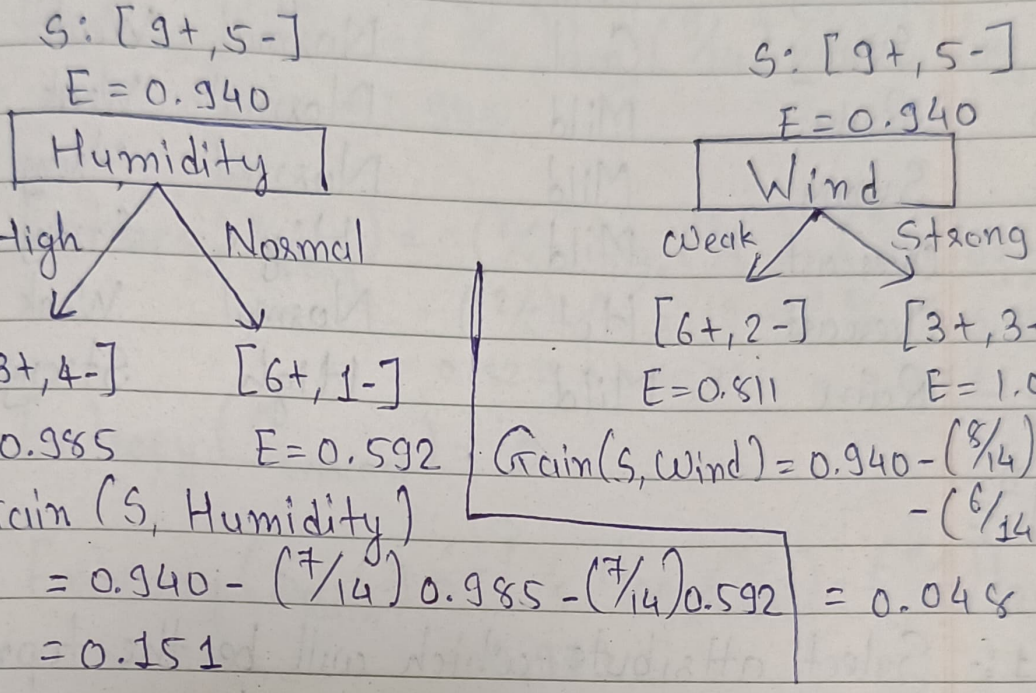
S = collection of examples

A = attribute

Values(A) = set of all possible values for attribute A

S_v = subset of S for which attribute A has value v (i.e. $S_v = \{s \in S | A(s) = v\}$

Which attribute is the best classifier



Gain (S, Humidity)
$= 0.940 - (7/14)\,0.985 - (7/14)\,0.592$
$= 0.151$

Gain(S, Wind) $= 0.940 - (8/14)\,0.811 - (6/14)\,1.0 = 0.048$

⇒ Humidity provides greater information gain then Wind, relative to the target classification.

☆ Example to illustrate the operation of ID3:-

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Sol^{n} :-

step 1 :- Select attribute which will be the topmost node of the decision tree.

⟹ ID3 determines the information gain for each candidate attribute, then selects the one with highest information gain.

$$Gain(S, Outlook) = 0.246$$
$$Gain(S, Humidity) = 0.151$$
$$Gain(S, Wind) = 0.048$$
$$Gain(S, Temperature) = 0.029$$

$$Gain(S, Outlook) = Entropy(S) - \sum_{v \in \{Sunny, Overcast, Rain\}} \frac{|S_v|}{|S|} Entropy(S_v)$$

0.6374
-1.4854
0.5305 × 0.4097
-0.4418
-1.030
0.2840
0.4097
-0.1919
-0.4472
0.1234 + 0.1597

Gain(S, Outlook) =

$$\text{Entropy}(S) = -(9/14)\log_2(9/14) - (5/14)\log_2(5/14)$$
$$= 0.940$$

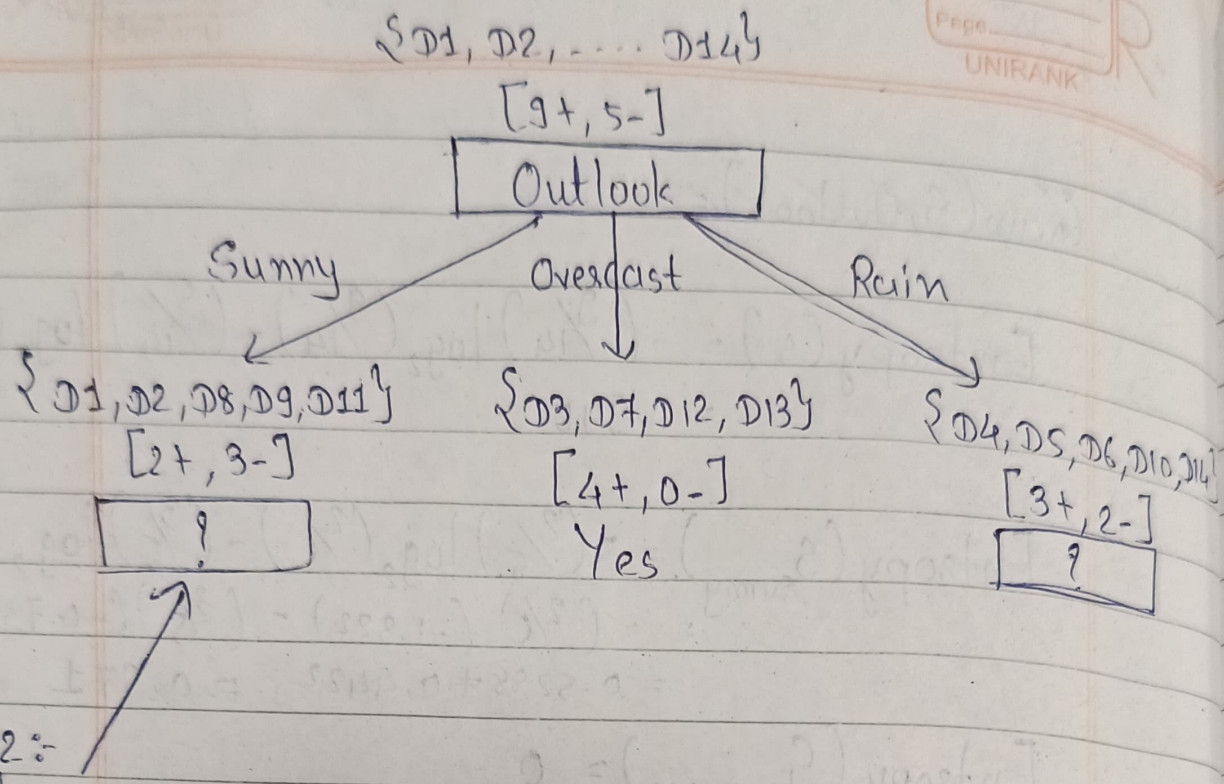
$$\text{Entropy}(S_{sunny}) = -(2/5)\log_2(2/5) - (3/5)\log_2(3/5)$$
$$= -(2/5)(-1.322) - (3/5)(-0.737)$$
$$= 0.5288 + 0.4422 = 0.971$$

$$\text{Entropy}(S_{overcast}) = 0$$

$$\text{Entropy}(S_{Rain}) = -(3/5)\log_2(3/5) - (2/5)\log_2(2/5)$$
$$= 0.971$$

$$\text{Gain}(S, \text{Outlook}) = 0.940 - \left[\frac{5}{14} \cdot 0.971 + 0 + \frac{5}{14} \cdot 0.971\right]$$
$$= 0.940 - [0.3468 + 0.3468]$$
$$= 0.2464$$

⇒ According to the information gain measure, the Outlook attribute provides the best prediction of the target attribute, PlayTennis, over the training examples. Therefore, **Outlook** selected as the root node for decision tree and branches are created below the root for each of its possible value



Step 2:-
Which attribute should be tested here?

$S_{sunny} = \{D1, D2, D8, D9, D11\}$

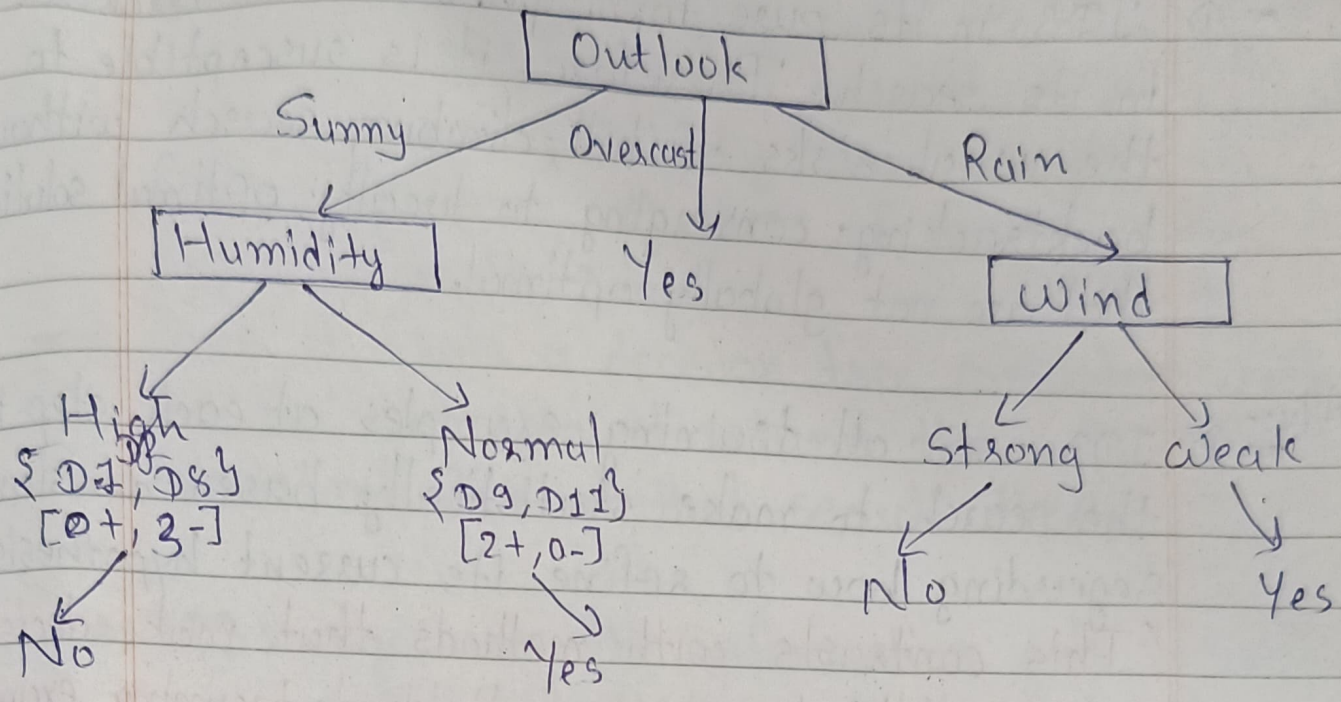
$Gain(S_{sunny}, Humidity) = 0.970 - (3/5)0 - (2/5)0 = 0.970$

$Gain(S_{sunny}, Temperature) = 0.970 - (2/5)0 - (2/5)1 - (1/5)0 = 0.570$

$Gain(S_{sunny}, Wind) = 0.970 - (2/5)1 - (3/5)0.918 = 0.19$

$\Rightarrow$ Humidity attribute is selected at this step.

✱ Final constructed decision tree:-



✱ Capabilities and Limitations of ID3:-

* Capabilities:-

⟹ ID3's hypothesis space of all decision trees is a complete space of finite discrete-valued functions, relative to the available attributes.

Limitation:-

⟹ ID3 maintains only a single current hypothesis as it searches through the space of decision trees. So ID3 loses the capabilities that follow from explicitly representing all consistent hypotheses.

limitation:
⟹ ID3 in its pure form performs no backtracking in its search. Therefore, it is susceptible to the usual risks of hill-climbing search without backtracking: converging to locally optimal solutions that are not globally optimal.

Adv:-
⟹ ID3 uses all training examples at each step in the search to make statistically based decisions regarding how to refine its current hypothesis. (This contrasts with methods that make decisions incrementally, based on individual training example (Ex FIND-S or CANDIDATE-ELIMINATION). One advantage of using statistical properties of all the examples (ex information gain) is that the resulting search is much less sensitive to errors in individual training examples.

⟹ ID3 can be easily extended to handle noisy training data by modifying its termination criterion to accept hypotheses that imperfectly fit the training data.

☆ Inductive bias in Decision Tree learning (ID3):
Shorter trees are preferred over larger trees. CA
-Th
an
thi

- BFS-ID3: algo. begins with the empty tree and searches breadth first through progressively more complex trees, first considering all trees of depth 1, then all trees of depth 2 etc.

⟶ Once it finds a decision tree consistent with the training data, it returns the smallest consistent tree at that search depth.

- ID3: algo. uses the information gain heuristic and a hill climbing strategy, it exhibits a more complex bias than BFS-ID3.

☆ A closer approximation to the inductive bias of ID3
- Shorter trees are preferred over longer trees. Trees that place high information gain attributes close to the root are preferred over those that do not..

Difference between the hypothesis space search in these following two approaches

CANDIDATE-ELIMINTION

- The version-space of this algo searches an incomplete hypothesis space. It searches this space completely finding every hypothesis consistent with training data.

ID3

- ID3 searches a complete hypothesis space but it searches incomplete through this space. (from simple to complex)