Ankit Goyal
Roll No. - 17103011
Group - G1
Subject - ML Assignment (I+II)

Page No.
Date : / /20

1. **Info gain:**

$$I(p,n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$s = (p+n)$$

$$E(A) = \sum_{i=1}^{v} \frac{p_i+n_i}{p+n} I(p_i,n_i)$$

$$\text{Gain}(A) = I(p,n) - E(A)$$

$$s = ST + A + C = 10$$

$$\text{Info gain} = -\left[\frac{3}{10}\log_2\frac{3}{10} + \frac{3}{10}\log_2\frac{3}{10} + \frac{4}{10}\log_2\frac{4}{10}\right]$$

$$= -\left[0.6 \times \left(\frac{-0.522}{0.301}\right) + \frac{(0.4)(-0.397)}{0.0301}\right]$$

Info gain = 1.562

- finding spliting attr. (attr with highest gain)

1. Sore throat

	S.T	A	C	
Yes	2	1	2	(info gain) I_1
No	1	2	2	(info gain) I_2

$\}+$ = Entropy (S.T)

E (Sore throat)

$$I(\text{Yes}) = -\left[\frac{2}{5}\log_2\frac{2}{5} + \frac{1}{5}\log_2\frac{1}{5} + \frac{2}{5}\log_2\frac{2}{5}\right]$$

$I(\text{Yes}) = 1.52$
$I(\text{No}) = 1.52$

E (Sore throat) = 0.5 × 1.52 + 0.5 × 1.52 = 1.52

Gain = 1.562 − 1.52

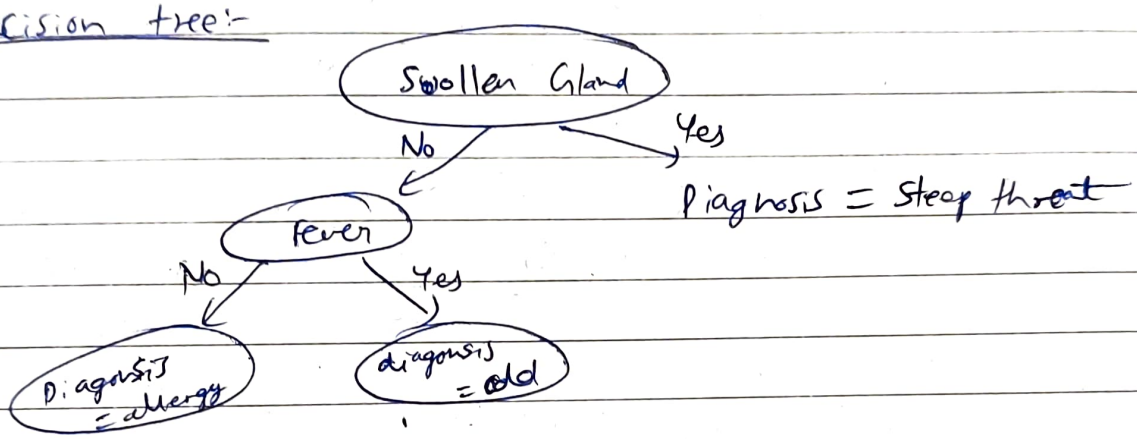
Gain = 0.05

Page No.
Date : / /20

Gain of each attr.

attr.	gain
Sore throat	0.05
Fever	0.72
Swollen Gland	0.88
Congession	0.45
Headache	0.05

Decision tree:-



2. Posilini example (Japan, Honda, blue, 1980, economy)
initialize G to a singelton that includes everything

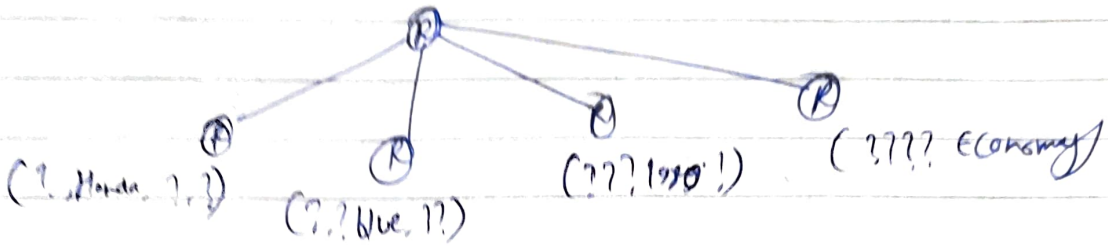
(?, ?, ?, ?) (A)

(Japan, Honda, blue, 1980, economy) (B)

These models represent the most general and the most specific heuristics one might learn.

The actual heuristics to be learned "Japanese Economy or" probably lie between them somewhere within the version space.

Negative Example:- (Japan, toyota, Green, 1970, sports)

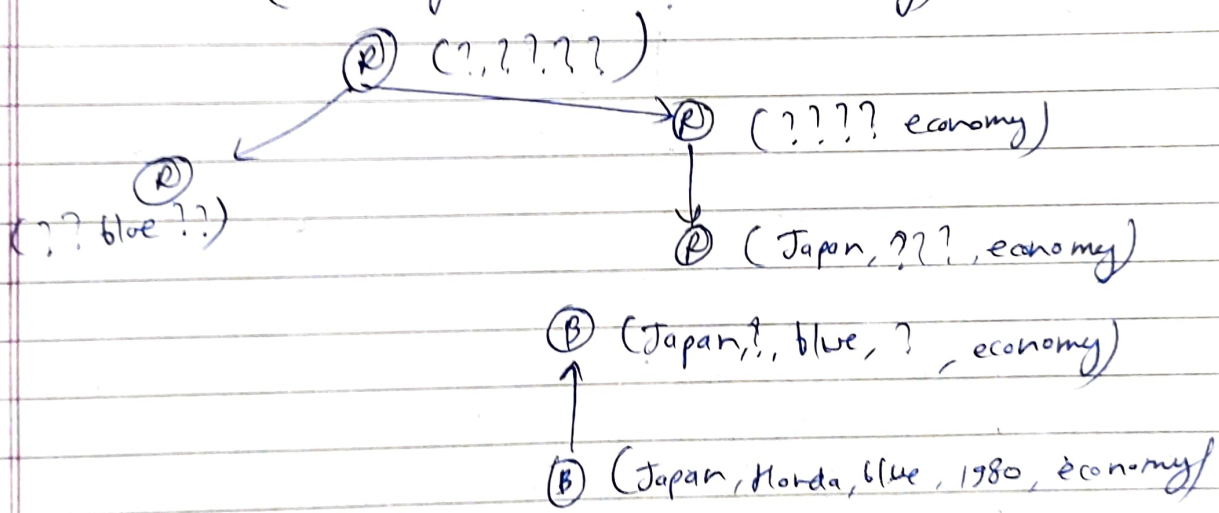


(Japan, Honda, blue, 1980, economy)

Refinement occurs by generalising S or specialising G, until the heuristic hopefully converges to one that works well.)

Negative example:-

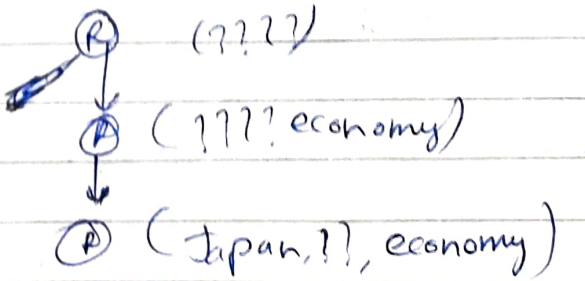
(USA, chrysler, Red, 1980, economy)



Positive example:- (Japan, Honda, white, 1980, economy)

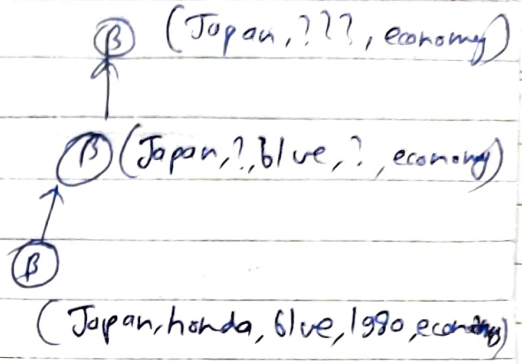
Prune G to exclude description inconsistent with positive example.

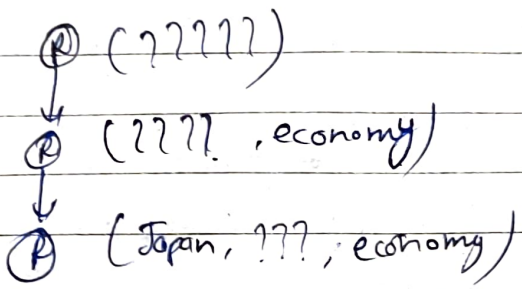
Generate S to include positive example.

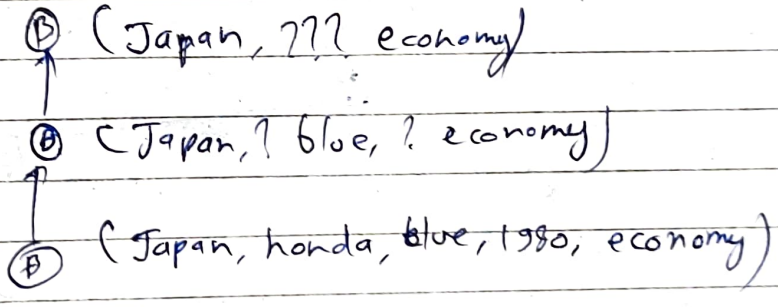


G & S are singleton sets
and S = G
converged

No more data, so algorithm stops.







3. Genetic Algorithm:-

A genetic algo. is a search heuristic that is inspired by Darwin's theory of natural evolution. The algorithm reflects the process of natural selection where the fittest individuales are selected for reproduction in order to produce offspring of the next generation.

equality "a+2b+3c+4d" = 30

Step-1 :- initialisation

We define no. of chromosomes in the population are 6.

Page No.
Date : / /20

c[1] = [a, b, c, d] = [12, 05, 23, 08]
c[2] = [a, b, c, d] = [02, 21, 18, 03]
c[3] = [a, b, c, d] = [10, 04, 13, 14]
c[4] = [a, b, c, d] = [20, 01, 16, 06]
c[5] = [a, b, c, d] = [01, 04, 13, 19]
c[6] = [a, b, c, d] = [20, 05, 17, 01]

Step-2 ⇒ Evaluation:

Eobj[1] = Abs(12 + 2×0.5 + 3×23 + 4×08 − 30) = 93
Eobj[2] = Abs(02 + 2×21 + 3×18 + 4×03 − 30) = 80
Eobj[3] = Abs(10 + 2×04 + 3×13 + 4×14 − 30) = 83
Eobj[4] = Abs(20 + 2×1 + 3×10 + 4×6 − 30) = 46
Eobj[5] = Abs(1 + 2×4 + 3×13 + 4×19 − 30) = 94
Eobj[6] = Abs(20 + 2×05 + 3×17 + 4×01 − 30) = 55

Step-3 ⇒ Selection

Fitness[1] = 0.0106
Fitness[2] = 0.0123
Fitness[3] = 0.0119
Fitness[4] = 0.0213
Fitness[5] = 0.0105
Fitness[6] = 0.0179

total = 0.0106 + 0.0123 + 0.0119 + 0.0213 + 0.0105 + 0.0179

= 0.0845

P[i] = Fitness[i] / total

P[1] = 0.1254
P[2] = 0.1456
P[3] = 0.1408
P[4] = 0.2521
P[5] = 0.1243
P[6] = 0.2118

Page No.
Date : / /20

c[1] = 0.1254 , c[2] = 0.2710 , c[3] = 0.4119
c[4] = 0.6639 , c[5] = 0.7882 , c[6] = 1.0

R[1] = 0.201 , R[2] = 0.284 , R[3] = 0.099
R[4] = 0.822 , R[5] = 0.398 , R[6] = 0.581

Thus chromosomes in population become :
c[1] = [02, 21, 18, 03] , c[2] = [10, 04, 13, 14]
c[3] = [12, 05, 23, 08] , c[4] = [20, 05, 17, 01]
c[5] = [10, 04, 13, 14] , c[6] = [20, 01, 10, 06]

```
begin
    K ← 0
    while (K < population) do
        R[K] = random (0-1);
        if (R[K] < Pc) then
            select chromosome [K] as parent;
        end
        K = K+1;
        end;
end;
```

Now we generate a random number R as the number of population

R[1] = 0.191 , R[2] = 0.259 , R[3] = 0.760
R[4] = 0.006 , R[5] = 0.159 , R[6] = 0.340

parents c[1], c[4] & c[5] will be selected for crossover

c[1] x c[4]
c[1] x c[5]
c[4] x c[5]

Page No.
Date : / /20

c[1] = [02, 05, 17, 01] c[2] = [10, 04, 13, 14]
c[3] = [12, 05, 23, 08] c[4] = [20, 04, 13, 14]
c[5] = [10, 04, 18, 03] c[6] = [20, 01, 10, 06]

Steps Mutation :-

Total gen = 4 * 6 = 24

Best chromosone obtained = [07, 05, 03, 01]

a = 7, b = 5, c = 3, d = 1

=> a + 2b + 3c + 4d = 0

=> 7 + 2 * 5 + 3 * 3 + 4 * 1 = 30

4. Bayes theorme is a way to figure out conditional probability. Conditional probability is the probability of an event happening, given that it has some relationship to one or more other events.

$$P(A|B) = P(B|A) \cdot \frac{P(A)}{P(B)}$$

Problems in Applying :-

1. Choice of Prior :- This is the usual carping for a reason coming up with a prior well that well reasons and actually represents your best attempt at summarizing a prior is a great deal of work.

2. Computationally intensive :-
For large dataset with many variables, being estimated. It may very intensive in computation.

Page No.
Date : / /20

5. K-Nearest Neighbours (KNN) algorithm is a type of supervised ML algorithm which can be used for both classification as well as regression predictive problems.

- **Lazy Learning Algorithm :-**

KNN is lazy learning algorithms because it does not have a specialised training phase.

- **Non parametric learning Algo. :-**

KNN is also a non-parametric learning algorithm because it does not assume anything about the underlying data.

- **Numerical :-**

1. Determine K = nearest neighbour
   use K=3
2. calculate distance
   $(7,7) \rightarrow (7-3)^2 + (7-7)^2 = 16$
   $(7,4) \rightarrow (7-3)^2 + (7-4) = 25$
   $(3,4) \rightarrow (3-3)^2 + (4-7)^2 = 9$
   $(1,4) \rightarrow (1-3)^2 + (4-7)^2 = 13$
3. Sort the distance & determine minimum distance.
4. Gather the category of the nearest neighbours.
5. Use simple majority of the category of nearest neighbour as the prediction value of quary instance.
6. $x_1 = 3, x_2 = 7$ lies in good category.

Date
Page

6. Types of Neural Networks

1. Feed-forward Neural Network:- Artificial Neuron

This neural network is one of the simplest forms of ANN, where the data or the input travels in one direction.

2. Radical basis function Neural Network:-

It consider the distance of a point with respect to the center. RBF function have two layers. First where the feature are combined with the radial basis functions in the inner large & output of these features

3. Kohonen self organising Neural Network:-

The objective of Kohonen map is to input vectors of arbitory dimension to discrete map comprised of neurons.

4. Recurrent Neural Network:-

The RNN work on the principle of savings the output of a large and feeding this back to the input to help in predicting the outcome of the layer.

5. Convolutional neural Network:-

It is similar to feed forward neural network, where the neurons have learnable weights & biases.

Date
Page

7. The required perception has 3 inputs A, B, C. the constants 1. The values of A & B are 1 (true) or -1 (false)

A	B	O = A ∧ ¬B
-1	-1	-1
-1	1	-1
1	-1	1
1	1	-1

the line crosses the A axis at 1 & B axis at -1.
the equation of the line is

$$\frac{A-0}{1-0} = \frac{B-(-1)}{0-(-1)} \Rightarrow A = B+1$$

$$\Rightarrow \boxed{1 - A + B = 0}$$

B, 1, -1, 1 are possible values for the weights w_1, w_2, w_0 using these values the output of the perception for A=1, B=-1, is -ve. Therefore we need to negate the weights and therefore we can conclude that
$w_0 = -1$, $w_1 = 1$, $w_2 = 1$

9. The main Requirements of clustering algos:-
- Scalability
- dealing with diff. types of attributes
- discovering clusters of arbitary shapes
- minimal requirements for domain Knowledge to determine input parameters.
- ability to deal with noise & others
- insensitivity to order of input records
- high dimensionality
- interpretability & usability.

Rush
PAGE NO.:
DATE: / /

9. The goal of a support vector machine is to find the optimal hyperplane which maximised the margin of the training data.

Calculating Margin:
Given a particular hyperplane, we can compute the distance b/w the hyperplane and the closest data point. Once we have their values if we double it we will get what is called the margin.
That is why the objective of the sum is to find the operating hyperplane which maximises the margin of the training data.

10. Steps of machine learning:-

1. data collection → Quantity and Quality of the data indicate how accurate the model is.

2. Data PReparation → Arrange the data and propose it for training.

3. choose Model:- Different Algo are for different tasks choose the right one.

4. Train the Model:- The goal of training is to answers a question or make a prediction correctly.

5. Evaluate the Model:- Uses some metric or combination of metrics to measure objective performance of the Model.

6 Parameter learning:- Tune model parameter for improve performance.

7. Make ~~Pre~~ Prediction:-
Using test set data a better approximation of how the model will perform in real world.

The checker problem:-
T: playing checkers.
P: percent of games won against opponents
E: playing practice games against itself

target function V. Board → R
$\hat{V} = w_0 + w_1 x_1(b) + w_2 x_2(b) + w_3 x_3(b) + w_4 x_4(b) + w_5 x_5(b) + w_6 x_6(b)$

Adjusting the weights.
$E = \sum_{(b, V_{train}(b)) \in \text{learning example}} (V_{train}(b) - \hat{V}(b))^2$

LMS Algo:-
1. compute error(b) = $V_{train}(b) - \hat{V}(b)$
2. for each board feature x_i, update weight
$w_i \leftarrow w_i + \eta \cdot error(b) \cdot x_i$

Find design for checkers system:
① The performance system → takes a new board as I/o a trace of the game.
② The critic :- Takes the trace of game as Input & output a set of training example of the target fxn.
③ The generalises:- Takes training example as input output hypothesis that estimates the target.
④ Experiment Generator:- It takes the current I/o a new problem for the performance to expose.