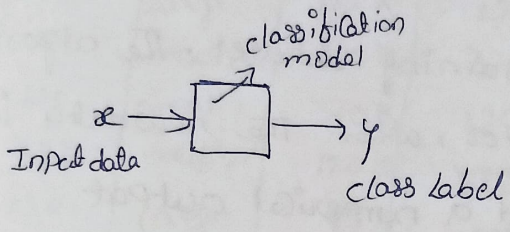
UNIT - IV

Classification



It is the task of assigning objects to one of several predefined categories.

It is the task of learning a target function that maps each attribute set to one of predefined classification function.

Purpose of classification model:-

Descriptive modeling:-

A classification technique (or classifier) is a systematic approach to build classification models from an input data set. It is to identify category (or) class label.

In classification data is grouped into categories based on a training data set. using the training data set, the algorithm derives a model or the classifier, the derived model can be a decision tree.

In test data set usually unlabeled data is given to the model. Then it finds the class to which it belongs

eg: Computer sales (Y/N)

Prediction:

Prediction is the process of identifying the missing or unavailable numerical data for a new observation

According to the training dataset, the algorithm derives the model or a predictor, when the new data is given the model should bend a numerical output

unlike in classification, this method does not have to class label the model predicts a continuous-valued functions or ordered values

eg:- computer sales(How much he/she spend)

Classification eg:-

computer sales - profession, salary, requirment - Y/N

Customer loan - salary designation - ?

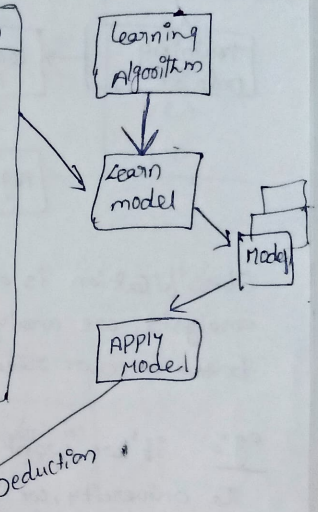
Diagnosis - Fever, Cold, Cough, tests - ?

online shoping - Previous search, item interested

Illustrating Classification Task

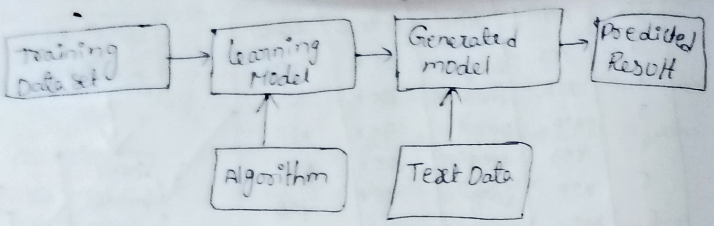
Building a classification model (General approach)

Tid	Attri1	Attri2	Attri3	class
1	Yes	Large	125K	NO
2	NO	medium	100K	NO
3	NO	small	70K	NO
4	Yes	medium	120K	~~NO~~
5	NO	Large	95K	~~Yes~~
6	NO	Medium	60K	NO
7	~~Yes~~	Large	220K	~~NO~~
8	NO	small	85K	Yes
9	NO	medium	75K	NO
10	NO	small	90K	Yes



Tid	Attrib1	Attrib2	Attri3	class
11	NO	small	55K	?
12	Yes	medium	80K	?
13	Yes	large	110K	?
14	NO	small	95K	?
15	NO	Large	67K	?

General approaches to solving a classification problem:



Classification is one of the methods used for data analysis, we analyze the data and classify it based on our requirment

eg:- If we want to know the performance of the university, we classify the student database based on their performance as above average, average and below average students.

If the classification shows that the no. of students under "below average" category are more, then the university needs to improve

Data classification process:

Let us consider the data classification where a decision is to be made for increasing the pay scale of employees in organization based on their performance level and current pay scale

Training Data.

Name	age	Performance	Current-Pay	Like_decission
Ram	30	Good	20,000	yes
Raj	40	Good	25,000	yes
Ramya	25	Good	27,000	yes
Ravi	30	Bad	20,000	no

↓

Classification algorithm

↓

Rules of classification

If Performance = Bad then Like_decisim =No

If Current_Pay_Scale < 26,000 and Performance = Good then Like_decision = yes

If age ≥ 30 and Current_Pay_Scale < 30000 then Like_decission = yes.

↓

Test Data.

Name	Agg	Performance	Current-Pay. Scale	Like-division
Vikki	25	Good	20,000	yes
Yash	35	Good	26,000	yes
Arjun	35	Good	22,000	yes
Varun	27	Bad	24,000	no

↓

Rules of classification

↓

New Data

Rahul, 30, Good, 20,000 Like_decision?

↓

General approach to classification:

- → Training set consists of records with known Class Labels
- → Training set is used to build a classification model
- → A labeled test set of previously unseen data records is used to evaluate the quality of the model
- → The classification model is applied to new records with unknown Class Labels.

Evaluation of classifiers & classification techniques.

A classification technique (or) classifier is a systematic approach to building classification models from an input dataset.

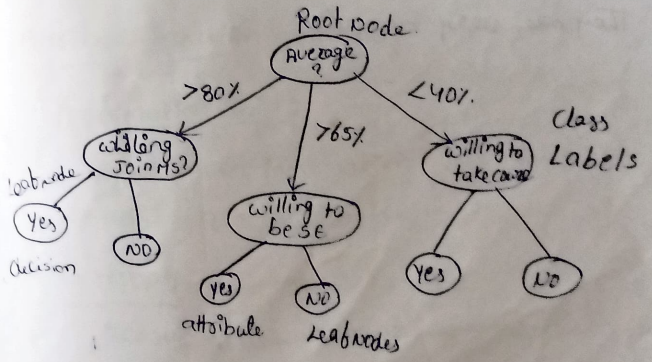
Examples
- → Decision tree classifier
- → Rule based c classifier
- → Neural Networks
- → Support Vector machines &
- → Na"ive Bayes Classifiers

Each technique employs a learning algorithm to identify a model that best fits the relationship between the attribute set and class label of the input data.

The model generated by a learning algorithm should both fit the input data well and correctly predict the class labels of records it has never seen before.

Decision Trees - Decision tree construction:-

- → A decision tree is a tree - structure
- → where each non-leaf node represents the test on an attribute
- → branches represents the outcome of the test and the test and the leaf nodes represents the class labels.



The decision tree enables the organization to identify the no of students who are going to join a software company. Some decision trees are binary and some trees are no binary.

Decision trees are mostly used for classification rules for tuples which dont have class labels identifier for them

The class predictions can be made by traversing from node to the leaf node.

Advantages of Decision tree :-

(i) the don't require domain knowledge
(ii) they are easy to understand
(iii) Handles high dimensional data
(iv) classification and learning becomes simpler when decision trees are used.
(v) They are very accurate.

Decision tree uses Information Gain

$$I(P,n) = -\frac{P}{S}\log_2\frac{P}{S} - \frac{n}{S}\log_2\frac{n}{S} \quad S = (P \& n)$$

$$\text{Entropy} - E(A) = \sum_{i=1}^{v} \frac{P_i + n_i}{P+n} I(P_i, n_i)$$

$$\left\{\log_2 x = \frac{\log_{10} x}{\log_{10} 2}\right\}$$

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

Decision tree induction :-

→ Flow chart like tree structure
→ supports in taking decisions
→ it defines the rules visually in form of tree

Types of node :-

1. Root node : Main question
2. Branch node : Intermediate processes
3. Leaf node : Answer

Attribute selection measures :-

1. Information Gain :

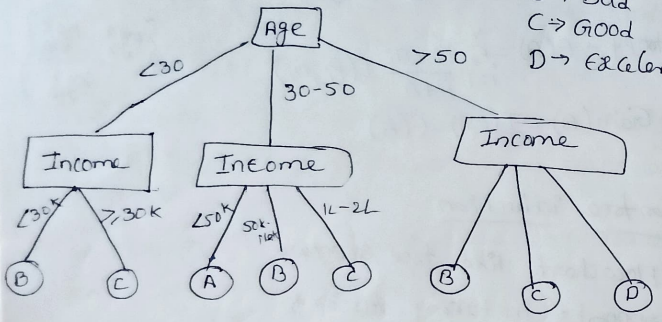
How much information does the answer to the specific question provide.

2. Entropy :- (IG ↑ E ↓)

Measures the amount of uncertainity (or randomness) in the information

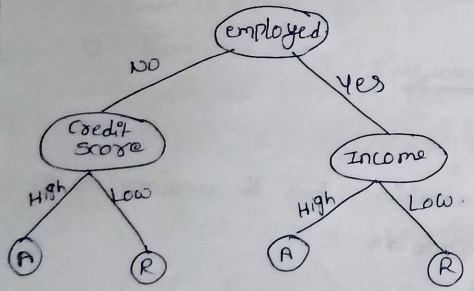
Example: credit score Rating

A → Avg
B → Bad
C → Good
D → Excelent



Dataset → Algorithm → classifies the data
(decision tree algorithm)

example: Loan System
(decides the loan should be approved or not)



→ Algorithm (ID3) : (repeatedly) (divides) Iterative Dichotomiser 3

1. In the given dataset, choose a target attributes
2. Calculate information gain of target attributes

$$IG = \frac{-P}{P+N}\log_2\left(\frac{P}{P+N}\right) - \frac{N}{P+N}\log_2\left(\frac{N}{P+N}\right)$$

3. For remaining attributes, find entropy

Entropy = IG × Probability.

$$E(A) = \sum \frac{Pi+Ni}{P+N} I(P_i, N_i) \qquad \left(\log_2^x = \frac{\log x_{10}}{\log 2_{10}}\right)$$

4. Calculate Gain = IG − E(A)

Age	Competetion	Type	Profit
old	yes.	s/w	Down
old	No	s/w	Down
old	No	h/w	Down
mid	yes	s/w	Down
mid	yes.	h/w	Down
mid	No	h/w	up
mid	No	s/w	up
new	yes.	s/w	up
new	No	h/w	up
new	No	s/w	up

1453
1589
1290
21
4685

Step1: Target Attribute = Profit.

Step2:- Information Gain

$$IG = \frac{-P}{P+N}\log_2\left(\frac{P}{P+N}\right) - \frac{N}{P+N}\log_2\left(\frac{N}{P+N}\right)$$

P = count(Down) = 5

N = count(UP) = 5

$$= \frac{-5}{10}\log_2\left(\frac{5}{10}\right) - \frac{5}{10}\log_2\left(\frac{5}{10}\right)$$

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

$$= -\left(\frac{1}{2}\times -1\log_2^2 + \frac{1}{2}-1\log\frac{2}{2}\right)$$

$$= -\left(\frac{1}{2}\times -1 + \frac{1}{2}\times -1\right)$$

$$= -\left(-\frac{1}{2} - \frac{1}{2}\right)$$

$$= -(-1) = 1$$

IG = 1

Step3: Calculate entropy for remaining attributes

$$E(A) = \sum \frac{P_i+N_i}{P+N} I(A_i N_i)$$ [IG x probability]

Age:-

1. Prepare a table for each attribute

rows: Values of undertaken attributes (old, mid, new)

Columns: Values of target attributes (down, UP)

	down	UP
old	3	0
mid	2	2
new	0	3

Entropy = IG x Probability

P → down count

N → UP count

$$IG(old) = -\left(\left(\frac{3}{3}\right)\log_2\left(\frac{3}{3}\right) + \frac{0}{3}\log\left(\frac{0}{3}\right)\right)$$

$$= -\left(1\log_2 1 + 0\log 0\right) = 0$$

Probability = $\frac{3}{10}$

Entropy = $0 \times \frac{3}{10} = 0$

$$IG(mid) = -\left(\left(\frac{2}{4}\right)\log_2\left(\frac{2}{4}\right) + \frac{2}{4}\log_2\left(\frac{2}{4}\right)\right]$$

$$= -\left[\frac{2}{4}\log_2(1/2) + \frac{2}{4}\log_2(1/2)\right)$$

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

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

$$= -\left(-\frac{1}{2} - \frac{1}{2}\right)$$

$$= -(-1) = 1$$

Probability = $4/10$ = Entropy(mid) = $1 \times \frac{4}{10} = 0.4$

$$IG(new) = -\left(\left(\frac{0}{3}\right)\log_2\left(\frac{0}{3}\right) + \left(\frac{3}{3}\right)\log\left(\frac{3}{3}\right)\right)$$

$$= 0$$

Probability = $\frac{3}{10}$ Entropy = $0 \times \frac{3}{10} = 0$

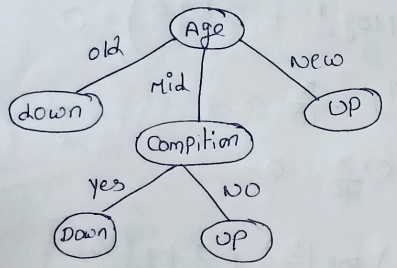
Entropy(Age) = E(O) + E(M) + E(N) = 0 + 0.4 + 0 = 0.4

(4) Gain = IG − E(A) = 1 − 0.4 = 0.6

Gain (Type) = 0

Gain (competition) = 0.124

Highest gain → root node (Age)



old → all down

mid → some down some up

new → all UP

Competition :-

Prepare table

	UP	down
old	1	3
mid	4	2
new		

$$IG(yes) = -\left(\left(\frac{1}{4}\right)\log_2\left(\frac{1}{4}\right) + \left(\frac{3}{4}\right)\log_2\left(\frac{3}{4}\right)\right)$$

$$= -\left(\frac{1}{4}\log_2\left(\frac{1}{2^2}\right) + \left(\frac{3}{4}\right)\log_2\left(\frac{3}{4}\right)\right)$$

Navive-Bayes classifier :-

→ Classification technique based on Bayes Theorem with an assumption of independence among features.

$$P(A/B) = \frac{P(B/A)\,P(A)}{P(B)}$$

Fruit = {yellow, sweet, long}

Fruit	yellow	sweet	Long	total
orange	350	450	0	650
Banana	400	300	350	400
Others	50	100	50	150
Total	800	850	400	1200

$$P(\text{yellow}/\text{orange}) = \frac{P(O/Y)\,P(Y)}{P(O)}$$

$$= \frac{\left(\frac{350}{800}\right)\left(\frac{800}{1200}\right)}{\frac{650}{1200}}$$

$$= 0.53$$

$$P(\text{sweet}/\text{orange}) = \frac{P(O/S)\,P(S)}{P(O)} = 0.69$$

$$P(\text{long}/\text{orange}) = 0$$

$$P(\text{Fruit}/\text{orange}) = P(Y/O) * P(S/O) * P(L/O)$$

$$= 0.53 \times 0.69 * 0$$

$$= 0$$

$$P(\text{yellow Fruit}/\text{Banana}) = \frac{P(B/Y)\,P(Y)}{P(B)}$$

$$= \frac{\left(\frac{400}{800}\right)\left(\frac{800}{1200}\right)}{P\left(\frac{400}{1200}\right)}$$

$$= \frac{\frac{4}{8} * \frac{8}{12_3}}{\frac{1}{3}} = \frac{1}{3} * \frac{3}{1} = 1$$

$$P(\text{sweet}/\text{Banana}) = \frac{P(B/S)\,P(S)}{P(B)}$$

$$= \frac{\frac{300}{850} * \frac{850}{1200}}{\frac{400}{1200}} = \frac{1}{4} \times \frac{3}{1}$$

$$= 0.75$$

$$P(\text{long}/\text{Banana}) = \frac{P(B/L)\,P(L)}{P(B)} = \frac{\frac{350}{400} * \frac{400}{1200}}{\frac{400}{1200}}$$

$$= \frac{7}{\frac{35}{120}} \times \frac{3}{1} \quad \frac{400}{1200}$$

$$= 0.89$$

P(Fruit/Banana) = 1 * 0.75 * 0.89 = 0.65

P(Fruit/Others) = P(Y/O) * P(S/O) * P(T/O).

= 0.33 * 0.66 * 033 = 0.072

Banana having most Probability

Confusion Matrix:-

The Confusion matrix is a matrix used to determine the Performance of the classification models for a given set of test data. It can only be determined if the True value for test data are known.
The matrix itself can be easily understood but the related terminologies may be Confusion.
it shows the errors in the model performance in the form of a matrix, hence also known as an error matrix

Features of Confusion Matrix:-

→ For the 2 Prediction classes of classifiers, the matrix is of 2*2 table for three classes it is 3*3 table.

→ The matrix is divided into two dimensions, that are Predictive values and actual values along with the total no. of Predictions

→ Predicted values are those values, which are predicted by the model, and actual values are the true values for the given observations

It Looks Like.

n = total predictions	Predicted: NO	Predicted: Yes
Actual: NO	True Negative	False Positive
Actual: Yes	False Negative	True Positive

True Negative: Model has given Prediction NO, and the real or actual value was also NO.

True Positive: The model has Predicted Yes and the actual value was also true.

False Negitive :- The model has Predicted no, but the actual value was yes, it is also called as Type-II error.

False Positive :- The model has predicted yes, but the actual value was no, it is also called a type-I error.

Need for Confusion matrix :-

→ It evaluates the Performance of the classification models when they make predictions on test Data and tell how good our classification model is

→ It not only tells the error made by the classifier but also the type of errors such as it is either type-1 or type-2 errors.

→ with the help of the Confusion matrix we can calculate the different Parameters for the model, such as accuracy, Precision etc.

we can perform various calculations for the model, such as the model's accuracy, using the matrix.

→ Classification accuracy :-

It defines how often the model Predicts the correct output. It can be calculated as the ratio of the no. of correct Predictions made by the Classifier to all no. of Predictions made by the classifiers.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN}$$

→ Misclassification rate :- It is also termed as Error rate, and it defines how often the model gives the wrong predictions. The value of error rate can be calculated as the no. of the predictions made by the classifier.

$$\text{Error Rate} = \frac{FP+FN}{\text{Total}(TP+TN+FP+FN)}$$

→ Precision :- It can be defined as the no. of correct outputs provided by the model or out of all Positive classes that have predicted correctly by the model. how many of them were actually true.

$$\text{Precision} = \frac{TP}{TP+FP}.$$

Recall :-

It is defined as the out of total positive cases, how our model predicted correctly. The recall must be high as possible

$$\text{Recall} = \frac{TP}{TP+FN}$$

F-Measure :- If two models have low Precision and high recall. It is difficult to compare these models. For this we use F-score. This score helps us to evaluate the recall and precision at the same time. The F-score is maximum if the recall is equal to the Precision.

$$\text{F-Measure} = \frac{2 * \text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}}$$

example :-

Actual		Predicted: NO	Predicted: yes	
NO	165	50 TN	100 FP	60
yes		5 FN	100 TP	105
		55	110	

Accuracy :-

$$\frac{TP+TN}{\text{Total}} = \frac{100+50}{165} = \frac{150}{165} = 0.91$$

Error Rate :- 1 - accuracy = 1 - 0.91 = 0.09

(or)

$$\frac{FP+FN}{\text{Total}} = \frac{5+10}{165} = \frac{15}{165} = 0.09$$

Precision :-

$$\frac{TP}{TP+FP} = \frac{100}{100+10} = \frac{100}{110} = 0.64$$

(predicted yes)

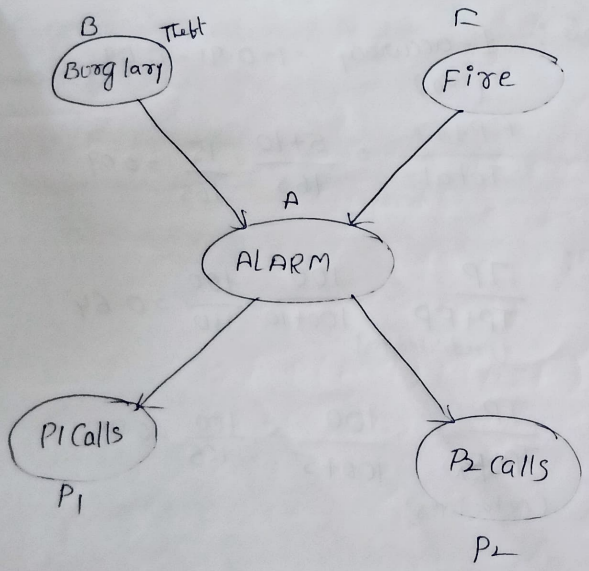
Recall :-

$$\frac{TP}{TP+FN} = \frac{100}{100+5} = \frac{100}{105} = 0.95$$

(Actual yes)

Bayesian-Belief Networks:-

Bayesian-Belief network is a graphical representation of different probabilistic relationships among random variables in a particular set. It is a classifier with no dependency on attributes i.e. it is condition independent. Due to its feature of joint probability, the probability in Bayesian Belief network is derived based on a condition - P(attribute/parent) i.e probability of an attribute, true over Parent attribute.



we have an alarm 'A' - a node, say installed in a house of a person 'gfg' which rings upon two probabilities i.e burglary 'B' and fire 'F', which are - Parent nodes of the alarm node. The alarm is the Parent node of two Probabilities P_1 Calls 'P_1' & P_2 Calls P_2 Person nodes.

→ Upon the instance of burglary and fire. 'P_1' and 'P_2' Call Person 'gfg' respectively. But there are few drawbacks in this Case, as sometimes 'P_1' may forget to call the person 'gfg' even after hearing the alarm, as he has a tendency to forget things quik. similarly 'P_2' sometimes fails to call the Person 'gfg', as he is only able to hear the alarm from a certain distance.

(Q) Find the Probability that 'P1' is true (P1 has called 'gfg') 'P2' is true (P2 has called 'gfg') when the alarm 'A' rang, but no burglary 'B' and Fire 'F' has occured.

⇒ P(P1, P2, A, ~B, ~F) [where - P1, P2 & A are true events and ~B & ~F are false events]

Burglary 'B':

→ P(B=T) = 0.001 ('B' is true i.e burglary has occured)

P(B=F) = 0.999 ('B' is false i.e burglary has not occured)

Fire 'F':-

P(F=T) = 0.002 ('F' is true i.e fire has occured)

P(F=F) = 0.998 ('F' is false i.e Fire has not occured)

Alarm 'A' :-

B	F	P(A=T)	P(A=F)
T	T	0.95	0.05
T	F	0.94	0.06
F	T	0.29	0.71
F	F	0.001	0.999

The alarm 'A' node can be 'true' or False (i.e may have rung or may not have rung). It has two Parent nodes burglary 'B' and Fire 'F' which can be 'true' or 'false' (i.e may have occured or may not have occured) depending upon different conditions

Person 'P1' -

A	P(P1=T)	P(P1=F)
T	0.95	0.05
F	0.05	0.95

The Person 'P1' node can be 'true' or 'false' (i.e may have called The Person 'gfg' or not) It has a Parent node, The alarm 'A', which can be 'true' or 'false' (i.e may have rung or may not have rung, upon burglary 'B' or Fire 'F').

Person 'P2' :-

A	P(P2=T)	P(P2=F)
T	0.80	0.20
F	0.01	0.99

The Person 'P2' node can be 'true' or 'false' (i.e may have called the person 'gfy' or not) It has a parent node, the Alarm 'A' which can be 'true' (or) 'false' (i.e may have rung or may not have rung, upon burglary B or Fire 'F').

solution: Consider the observed probabilistic scan with respect to the question - $P(P_1, P_2, A, \sim B, \sim F)$ we need to get the probability of 'P1' we find it with regard to its parent node - alarm 'A' To get the probability of 'P2' we find it with regard to its parent node - alarm 'A'.

we find the probability of alarm 'A' node with regard to '~B' & '~F' since burglary 'B' and fire 'F' are parent nodes of alarm 'A'

From the observed probabilistic scan, we can

$$P(P_1, P_2, A, \sim B, \sim F)$$

$$= P(P_1/A) * P(P_2/A) * P(A/\sim B \sim F) * P(\sim B) * P(\sim F)$$

$$= 0.95 * 0.80 * 0.001 * 0.999 * 0.998$$

$$= 0.00075.$$