UNIT 4

PART 1

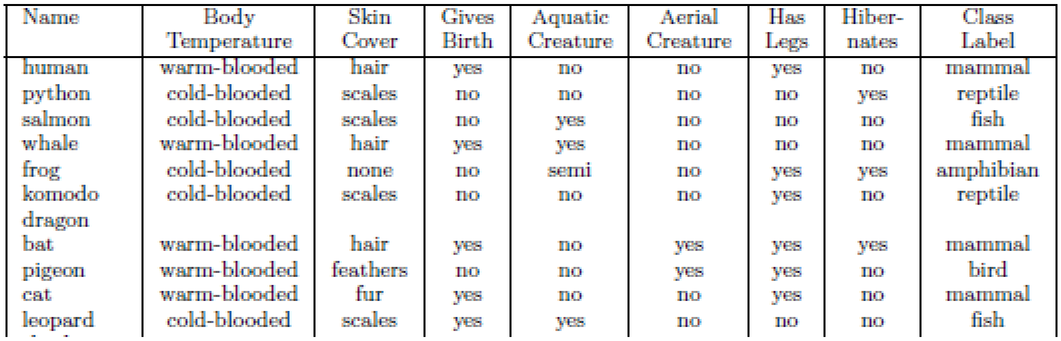
BASIC CONCEPTS

Que 1: What is classification? What is purpose of classification?

Classification is the process of finding a model (or function) using training data that describes and distinguishes data classes or concepts, for the purpose of being able to use the model to predict the class of objects whose class label is unknown.

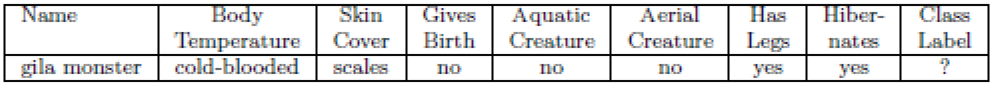
Purpose of classification: **1) Descriptive modeling 2) Predictive modeling**

Descriptive modeling is a classification model used for summarizing the data in terms of classes and attributes as shown below



Name	Body Temperature	Skin Cover	Gives Birth	Aquatic Creature	Aerial Creature	Has Legs	Hiber-nates	Class Label
human	warm-blooded	hair	yes	no	no	yes	no	mammal
python	cold-blooded	scales	no	no	no	no	yes	reptile
salmon	cold-blooded	scales	no	yes	no	no	no	fish
whale	warm-blooded	hair	yes	yes	no	no	no	mammal
frog	cold-blooded	none	no	semi	no	yes	yes	amphibian
komodo dragon	cold-blooded	scales	no	no	no	yes	no	reptile
bat	warm-blooded	hair	yes	no	yes	yes	yes	mammal
pigeon	warm-blooded	feathers	no	no	yes	yes	no	bird
cat	warm-blooded	fur	yes	no	no	yes	no	mammal
leopard	cold-blooded	scales	yes	yes	no	no	no	fish

Predictive modeling is a classification model used to predict the class label of unknown records.



Name	Body Temperature	Skin Cover	Gives Birth	Aquatic Creature	Aerial Creature	Has Legs	Hiber-nates	Class Label
gila monster	cold-blooded	scales	no	no	no	yes	yes	?

Que2: What are the applications of Classification?

- **Customer Target Marketing**: Predict buying interests on the basis of previous training examples. The target variable may encode the buying interest of the customer.
- **Medical Disease Diagnosis**: The features may be extracted from the medical records, and the class labels correspond to whether or not a patient may pick up a disease in the future. In these cases, it is desirable to make disease predictions with the use of such information.
- **Biological Data Analysis:** Biological data can be used to build a model which can be used to predict a new virsus, bacteria etc…
- **Document Categorization and Filtering**: Many applications, such as newswire services, require the classification of large numbers of documents in real time. This application is referred to as document categorization.
- **Social Network Analysis:** For predicting useful properties of actors/person in a social network.
- **Detecting spam email messages** based upon the message header and content.
- **Classifying galaxies** based upon their shapes.

GENERAL APPROACH TO SOLVING A CLASSIFICATION PROBLEM

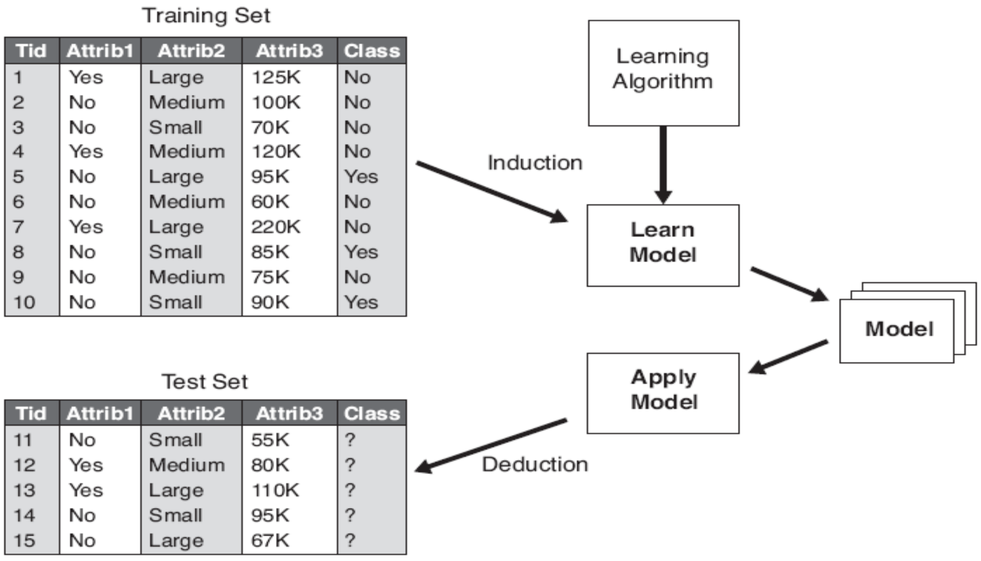
Que 3: What is the General approach to solve a classification problem?

Training data: Set of sample whose class labels are known and are used for building a model

Test set: Set of samples whose class labels are hided. The model is applied on test set to predict class labels. This predicted class labels are compared with actual class labels to know the accuracy of model.

Learning algorithm: Algorithm that learns from training data and create set of rules. This model can be used to predict class label of samples whose class label is unknown.

Model: It is set of rules to predict class labels.



Training Set

Tid	Attrib1	Attrib2	Attrib3	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

Test Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Figure 4.3. General approach for building a classification model.

1) First partition the historical data into training set and test set.
2) Use a classification algorithm like **decision tree classifiers**, to build a model from training data (Model is a set of rules learned from training set)
3) Apply this model on test set to check accuracy of model.

   $$\text{Accuracy of model} = \frac{\text{number of test samples correctly predicted}}{\text{total number of samples in test set}}$$

4) If accuracy is good, then apply the model on new /unknown sample whose class label is unknown and predict the class label.

Que 4: Write a short note on:

a. Confusion matrix

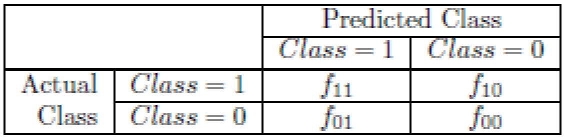
b. Performance metric:

i. Accuracy

ii. Error rate

Confusion matrix: It is a matrix having count of number of class labels correctly predicted and number of class labels incorrectly predicted by model.

Confusion matrix representation:



		Predicted Class	
		$Class = 1$	$Class = 0$
Actual Class	$Class = 1$	f_{11}	f_{10}
	$Class = 0$	f_{01}	f_{00}

In the above table $\boldsymbol{f_{11}}$ and $\boldsymbol{f_{00}}$ are correct predictions by model and $\boldsymbol{f_{01}}$ and $\boldsymbol{f_{10}}$ are incorrect predictions of model.

Accuracy: Defined as the ratio of samples correctly classified

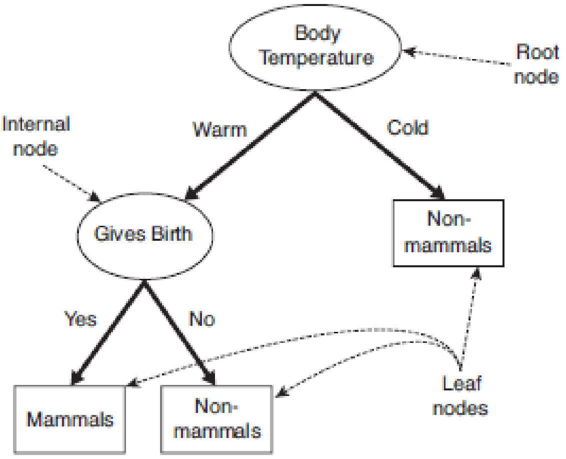
$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} = \frac{f_{11} + f_{00}}{f_{11} + f_{10} + f_{01} + f_{00}}.$$

Error Rate: Defined as the ratio of samples correctly classified

$$\text{Error rate} = \frac{\text{Number of wrong predictions}}{\text{Total number of predictions}} = \frac{f_{10} + f_{01}}{f_{11} + f_{10} + f_{01} + f_{00}}.$$

DECISION TREE INDUCTION

Que 5: Explain the working of a decision tree.



The tree has three types of nodes.

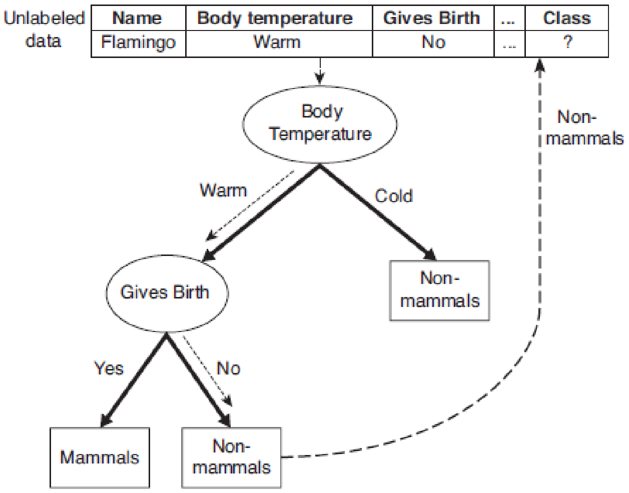
i) A **root node** has no incoming edges and zero or more outgoing edges.

ii) **Internal nodes**, each of which has exactly one incoming edge and two or more outgoing edges.

iii) **Leaf** or **terminal nodes**, each of which has exactly one incoming edge and no outgoing edges.

In terms of decision tree induction, root and internal nodes are called attribute test conditions. And, leaf (or) terminal nodes are class labels.

Once a decision tree is build, we apply *attribute test condition* on records and follow appropriate branches based on outcome of test condition. This will either lead to an internal *branch node* where another *attribute test condition* is applied or a *leaf node* where class label is assigned.


Unlabeled data

Name	Body temperature	Gives Birth	...	Class
Flamingo	Warm	No	...	?

BUILDING A DECISION TREE

Que 6 : Explain Hunt's algorithm for building decision trees

There are various methods for building a decision tree

1) **Hunt'algorithm**
2) ID3
3) C4.5
4) CART

Hunt's algorithm

In Hunt's algorithm, a decision tree is grown in a recursive fashion by partitioning the training records into subsets.

Let D_t be a set of training records that are associated with node t and $y=\{y_1,y_2,\ldots,y_c\}$ be the class labels.

The recursive procedure for hunt's algorithm is as follows:

STEP 1

If all the records in D_t belong to same class y_t, then t is a leaf node labeled as y_t.

STEP 2

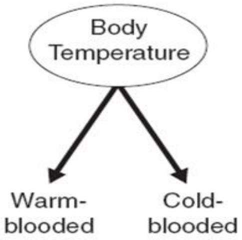
If D_t contains records that belong to more than one class, an attribute test condition is selected to partition the records into smaller subsets. A child node is created for each outcome and the records in D_t are distributed based on the outcomes. The algorithm is then recursively applied for each node.

METHODS FOR EXPRESSING AN ATTRIBUTE TEST CONDITIONS

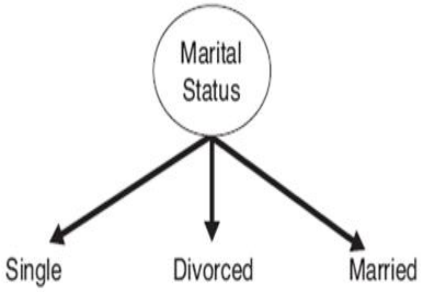
Que 7: What are the Methods for expressing attribute test conditions

The following are the methods for expressing attribute test conditions. They are:

i) **Binary attribute:** The test condition for binary attribute generate two outcomes as shown below:

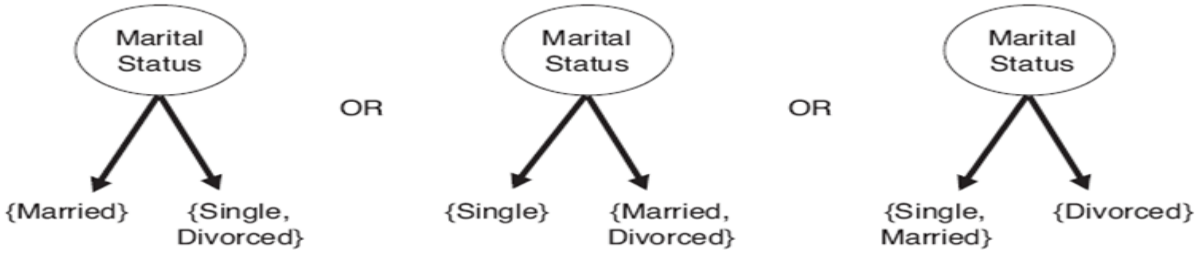


ii) **Nominal attributes:** since a nominal attribute can have many values, its test condition can be expressed in two ways as shown below:



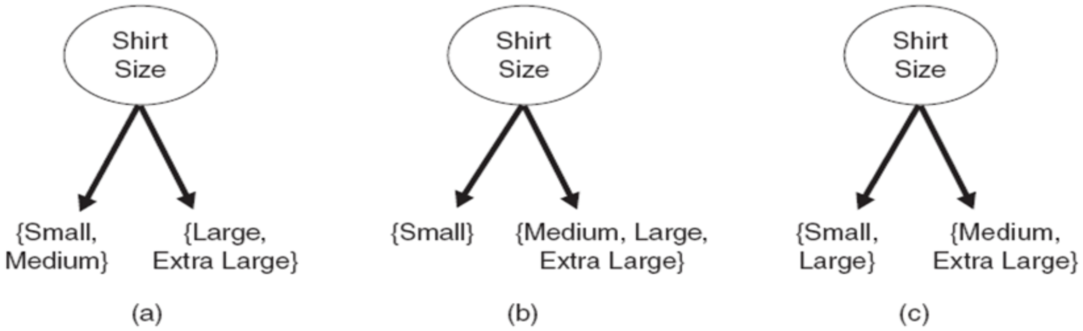
(a) Multiway split

For a multi way split, the number of outcomes depends on the number of distinct values for the corresponding attribute.



(b) Binary split {by grouping attribute values}

iii) Ordinal attribute: It can also produce binary or multi way splits. It is same as nominal attributes but the attribute values have order among them.

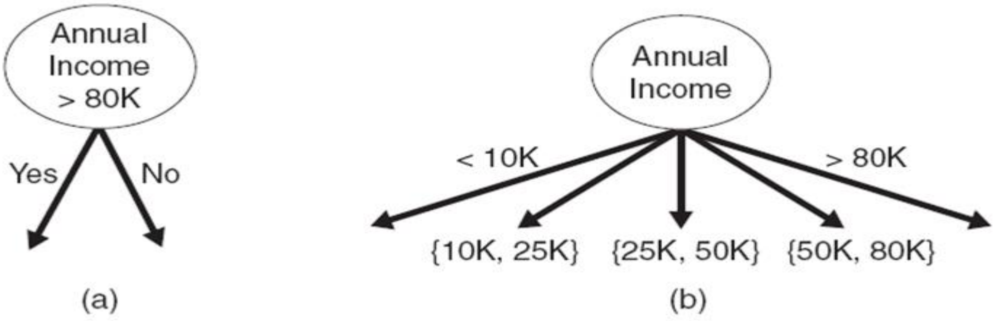


(a) (b) (c)

Figure 4.10. Different ways of grouping ordinal attribute values.

In the above example, condition 'a' and condition 'b' satisfies order but condition 'c' violates the order property.

iv) Continuous attributes: The test condition can be expressed as a comparison test with binary outcomes, or a range with many outcomes.



(a) (b)

Figure 4.11. Test condition for continuous attributes.

MEASURES FOR SELECTING THE BEST SPLIT

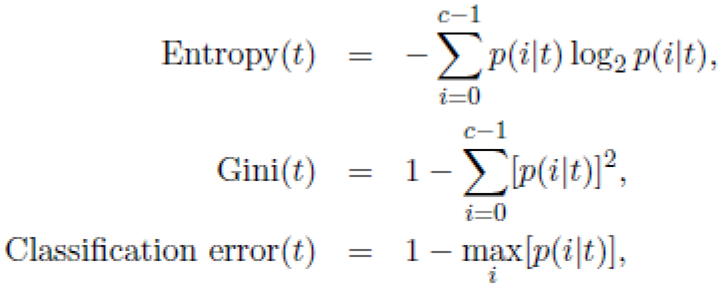
Que 8: what are the measures for selecting the best split

Three main methods for selecting the best split

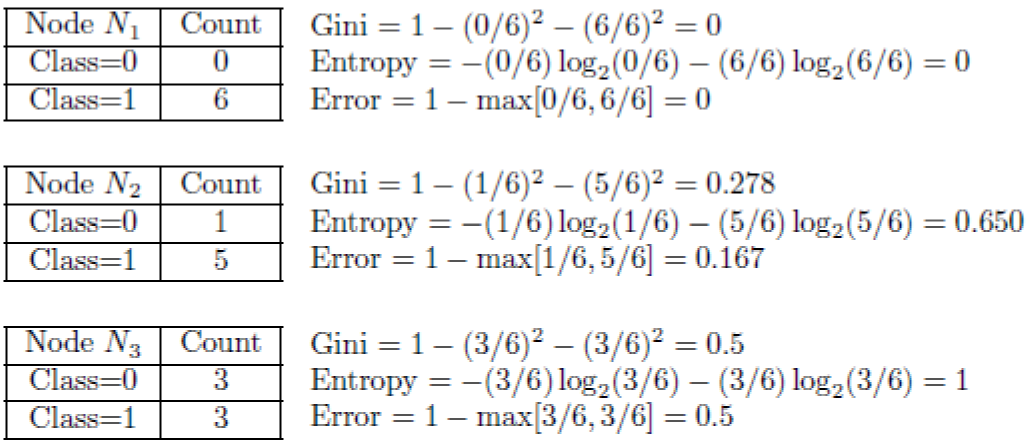
1) Entropy
2) Gini Index
3) Classification error

Let P(i|t) denote the fraction of records belonging to class i at a node t. the measures for selecting the best split are often based on the degree of impurity of the child nodes. The smaller the degree of impurity, the more skewed the class distribution. For example, a node with class distribution (0,1) has zero impurity, whereas a node with uniform class distribution (0.5,0.5) has the highest impurity.

Examples of impurity measures include: (Note: C is number of classes)



$$\begin{aligned}
\text{Entropy}(t) &= -\sum_{i=0}^{c-1} p(i|t)\log_2 p(i|t),\\
\text{Gini}(t) &= 1-\sum_{i=0}^{c-1}[p(i|t)]^2,\\
\text{Classification error}(t) &= 1-\max_i[p(i|t)],
\end{aligned}$$



Node N_1	Count
Class=0	0
Class=1	6

$$\begin{aligned}
\text{Gini} &= 1-(0/6)^2-(6/6)^2 = 0\\
\text{Entropy} &= -(0/6)\log_2(0/6)-(6/6)\log_2(6/6) = 0\\
\text{Error} &= 1-\max[0/6, 6/6] = 0
\end{aligned}$$

Node N_2	Count
Class=0	1
Class=1	5

$$\begin{aligned}
\text{Gini} &= 1-(1/6)^2-(5/6)^2 = 0.278\\
\text{Entropy} &= -(1/6)\log_2(1/6)-(5/6)\log_2(5/6) = 0.650\\
\text{Error} &= 1-\max[1/6, 5/6] = 0.167
\end{aligned}$$

Node N_3	Count
Class=0	3
Class=1	3

$$\begin{aligned}
\text{Gini} &= 1-(3/6)^2-(3/6)^2 = 0.5\\
\text{Entropy} &= -(3/6)\log_2(3/6)-(3/6)\log_2(3/6) = 1\\
\text{Error} &= 1-\max[3/6, 3/6] = 0.5
\end{aligned}$$

The 3 measures attain maximum values when the class distribution is uniform and minimum when all the records belong to same class.

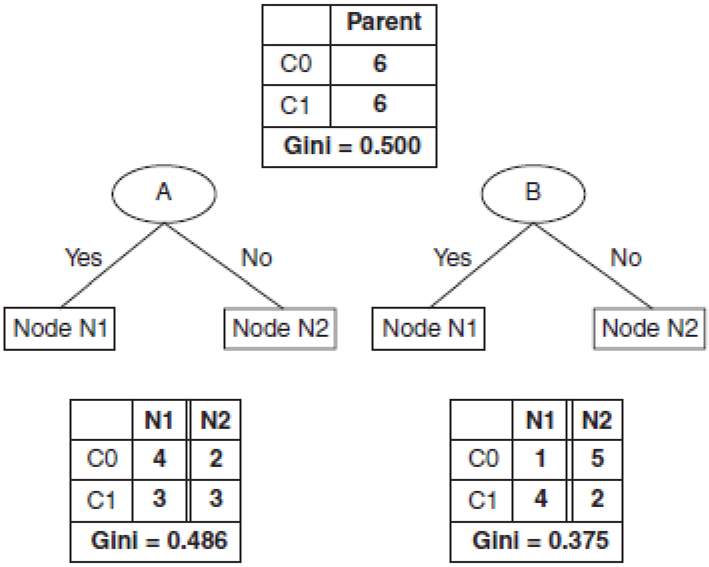
//Optional Compare the degree of impurity of the parent node with the degree of impurity of the child node. The larger their difference, the better the test condition. The gain, Δ, is a criterion that can be used to determine the goodness of a split.

$$\Delta = I(\text{parent}) - \sum_{j=1}^{k} \frac{N(v_j)}{N} I(v_j),$$

Where I(.) is the impurity measure of a given node, N is the total number of records at the parent node, k is the attribute values and N(v_j) is the number of records associated with node $v_{j.}$ when entropy is used as impurity measure the difference in entropy is known as information gain, $\Delta_{\text{info.}}$ **//**

Que 9: what are the methods for Splitting of binary attributes?

Suppose there are two ways to split the data into smaller subsets, say, A and B. before splitting the GINI index is 0.5 since there are equal number of records from both the classes.



	Parent
C0	6
C1	6
Gini = 0.500	

	N1	N2
C0	4	2
C1	3	3
Gini = 0.486		

	N1	N2
C0	1	5
C1	4	2
Gini = 0.375		

For attribute A,

For node N1, the GINI index is 1-[(4/7)^2+(3/7)^2]=0.4898

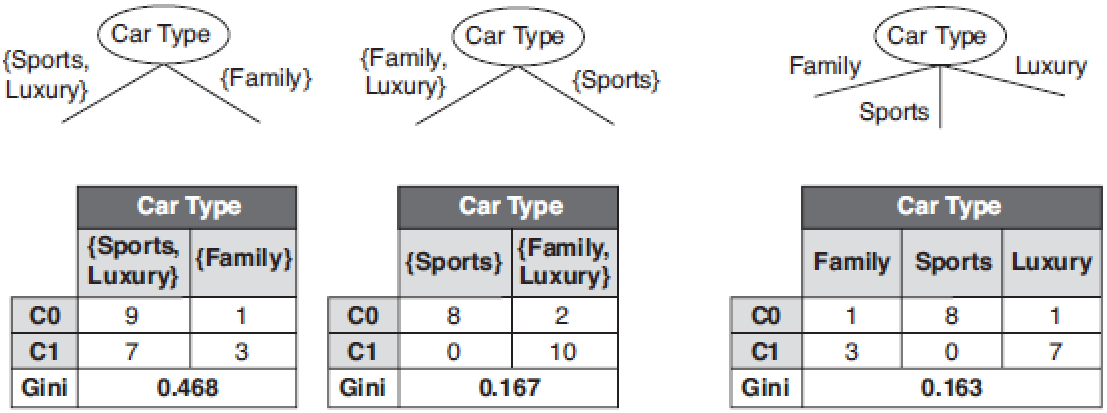
For node N2, the GINI index is 1-[(2/5)^2+(3/5)^2]=0.48

The average weighted GINI index is (7/12)(0.4898)+(5/12)(0.48)=0.486

For attribute B, the average weighted GINI index is 0.375, since the subsets for attribute B have smaller GINI index than A, attribute B is preferable.

Que 10: What are the methods for Splitting of nominal attributes?

A nominal attribute can produce either binary or multi way split.



	Car Type	
	{Sports, Luxury}	{Family}
C0	9	1
C1	7	3
Gini	0.468	

	Car Type	
	{Sports}	{Family, Luxury}
C0	8	2
C1	0	10
Gini	0.167	

	Car Type		
	Family	Sports	Luxury
C0	1	8	1
C1	3	0	7
Gini	0.163		

(a) Binary split

(b) Multiway split

The computation of GINI index is same as for binary attributes. The smaller the average GINI index is the best split. In our example, multi way split has the lowest GINI index, so it is the best split.

Consider 3rd table,

The Gini index for Family cars: 1-(1/4)^2-(3/4)^2 = 0.375

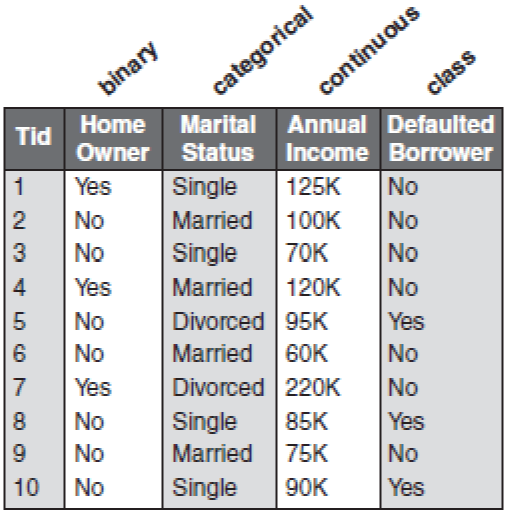
The Gini index for Sports cars: 1-(8/8)^2- (0/8)^2= 0

The Gini index for Luxury cars: 1-(7/8)^2-(1/8)^2=0.21938

(4/20)*0.375 + (8/20)*0+ (8/20)*0.21938=0.075+0.0877=0.163

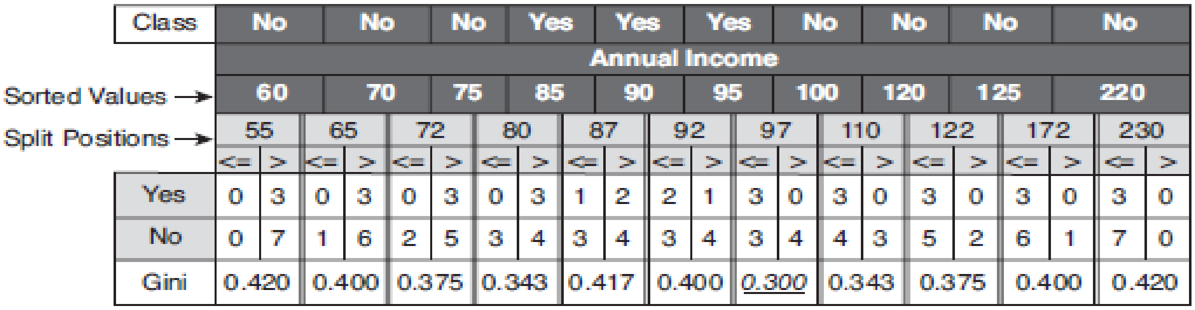
Que 11: What are the methods for Splitting of continuous attributes?

Consider continuous attribute Annual income from below table.



	binary	categorical	continuous	class
Tid	**Home Owner**	**Marital Status**	**Annual Income**	**Defaulted Borrower**
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

- Attribute values of annual income and corresponding class label are arranged in horizontal way as shown below. These are called **sorted values**
- Calculate the mean between every two adjacent values and plot below the sorted values. Theses mean values are called **split positions.**



Class	No	No	No	Yes	Yes	Yes	No	No	No	No
	Annual Income									
Sorted Values →	60	70	75	85	90	95	100	120	125	220

Split Positions →	55		65		72		80		87		92		97		110		122		172		230	
	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>
Yes	0	3	0	3	0	3	0	3	1	2	2	1	3	0	3	0	3	0	3	0	3	0
No	0	7	1	6	2	5	3	4	3	4	3	4	3	4	4	3	5	2	6	1	7	0
Gini	0.420		0.400		0.375		0.343		0.417		0.400		*0.300*		0.343		0.375		0.400		0.420	

Example:

Consider mean value 97,

Number of 'Yes' class labels <= 97 is 3

Number of 'No' class labels <=97 is 3

Number of 'No' class labels >97 is 4

Number of 'yes' class labels >97 is 0

Calculate Gini Index for less than 97:

$1-(3/6)^2-(3/6)^2= 0.5$

$1-(0/4)^2-(4/4)^2=0$

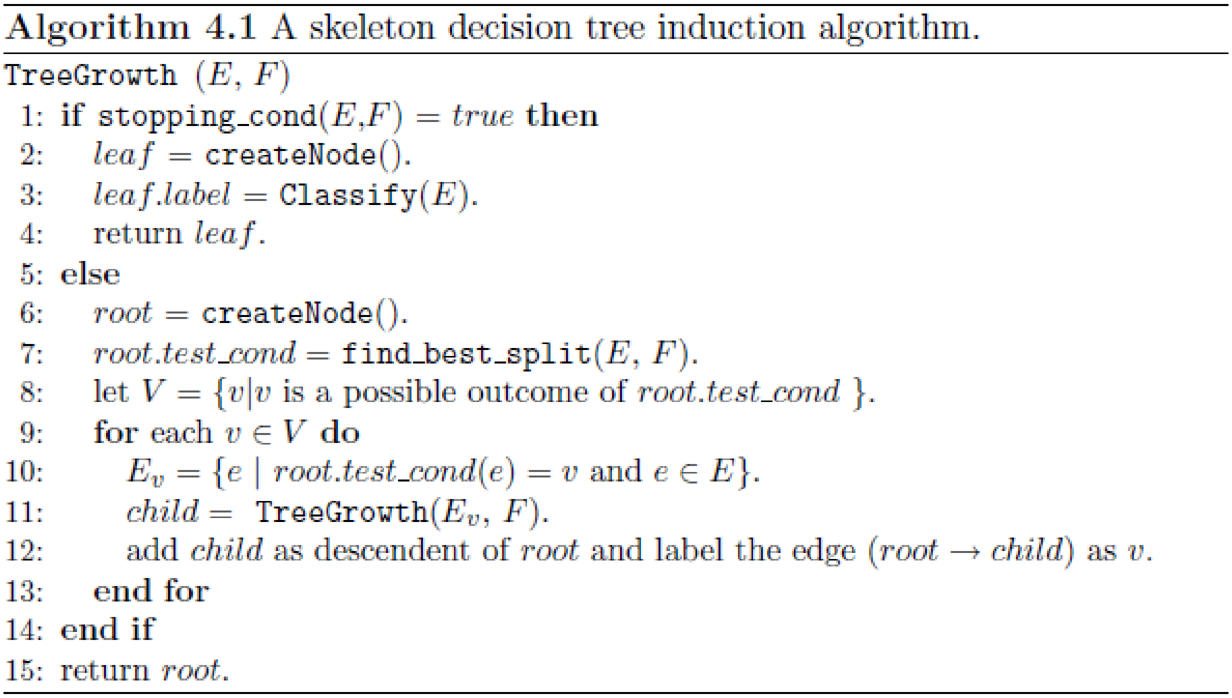
Average Gini index for 97 is

(6/10)* 0.5 + (4/10)*0=0.3

In the same way, Calculate the GINI index for every split position and the smaller GINI index split position can be chosen as the best split for the attribute.

ALGORITHM FOR DECISION TREE INDUCTION.

Que:12 Write the Algorithm for decision tree induction



Algorithm 4.1 A skeleton decision tree induction algorithm.

```
TreeGrowth ($E$, $F$)
if stopping_cond($E$,$F$) = $true$ then
    $leaf$ = createNode().
    $leaf.label$ = Classify($E$).
    return $leaf$.
else
    $root$ = createNode().
    $root.test\_cond$ = find_best_split($E$, $F$).
    let $V = \{v | v$ is a possible outcome of $root.test\_cond$ $\}$.
    for each $v \in V$ do
        $E_v = \{e \mid root.test\_cond(e) = v$ and $e \in E\}$.
        $child$ = TreeGrowth($E_v$, $F$).
        add $child$ as descendent of $root$ and label the edge $(root \rightarrow child)$ as $v$.
    end for
end if
return $root$.
```

i) The **createNode()** function extends the decision tree by creating a new node. A node in the decision tree has either a test condition, denoted as node.test_cond, or a class label, denoted as node.label.

ii) The **find.best_split ()** function determines which attribute should be selected as the test condition for splitting the training records.

iii) The **classify()** function determines the class label to be assigned to a leaf node.

iv) The **stopping_cond()** function is used to terminate the tree-growing process by testing whether all the records are classified or not.

Que 13: Explain a decision tree with an example

Consider the below table:

A	B	Class Label
T	F	+
T	T	+
T	T	+
T	F	−
T	T	+
F	F	−
F	F	−
F	F	−
T	T	−
T	F	−

The contingency tables after splitting on attributes A and B are:

	$A = T$	$A = F$
+	4	0
−	3	3

	$B = T$	$B = F$
+	3	1
−	1	5

The overall entropy before splitting is:

$$E_{orig} = -0.4 \log 0.4 - 0.6 \log 0.6 = 0.9710$$

The information gain after splitting on A is:

$$\begin{aligned}
E_{A=T} &= -\frac{4}{7} \log \frac{4}{7} - \frac{3}{7} \log \frac{3}{7} = 0.9852 \\
E_{A=F} &= -\frac{3}{3} \log \frac{3}{3} - \frac{0}{3} \log \frac{0}{3} = 0 \\
\Delta &= E_{orig} - 7/10 E_{A=T} - 3/10 E_{A=F} = 0.2813
\end{aligned}$$

The information gain after splitting on B is:

$$\begin{aligned}
E_{B=T} &= -\frac{3}{4} \log \frac{3}{4} - \frac{1}{4} \log \frac{1}{4} = 0.8113 \\
E_{B=F} &= -\frac{1}{6} \log \frac{1}{6} - \frac{5}{6} \log \frac{5}{6} = 0.6500 \\
\Delta &= E_{orig} - 4/10 E_{B=T} - 6/10 E_{B=F} = 0.2565
\end{aligned}$$

Therefore, attribute A will be chosen to split the node.

Que 14: What are the characteristics of Decision tree induction?

1) No previous assumptions and thresholds are needed prior to construction of decision tree induction.
2) Finding an optimal tree is heuristic
3) Constructing a decision tree is computationally less expensive. And, classifying an unknown sample is extremely fast.
4) Robust to presence of noise.
5) Redundant attributes do not adversely affect the accuracy of decision tree
6) Irrelevant attribute affect decision tree at large. Say, choosing an irrelevant attribute as test condition may result in irrelevant tree.
7) Data fragmentation problem: In some situations, number of training records become small as me move down the decision tree. At a stage, these samples may be insufficient to label the leaf node.
8) Tree replication problem: A same set of sub tree is replicated may times in a tree. This problem occurs when there are few attributes.

Que 15 : What do you mean by tree pruning? Why pruning is useful for decision tree induction. What is drawback of a separate set of tuples to evaluate pruning?

Tree pruning: Removing of branches (or) Trimming a tree in order to improve generalization capability of a decision tree is called tree pruning.

(or)

Trimming a tree to reduce generalization errors is called tree pruning. Tree pruning is mainly used to reduce over fitting of data

Techniques of tree pruning

1) Pre-pruning
2) Post-pruning

Pre-pruning: Tree growing on a training data is restricted by keeping certain thresholds fixed by user. Means, a leaf node is restricted to expand when its error crosses (Training and generalization errors) certain thresholds. But, large thresholds may lead to under fitting and low thresholds may lead to over fitting. And deciding correct thresholds is very tricky.

Post-pruning: Initially, a tree is fully grown on a training data. Then, the unnecessary branches are removed from the tree. Post-pruning is better than pre-pruning. However unnecessary components (leafs and branches) should be grown which are computationally very expensive.

The decision tree built may overfit the training data. There could be too many branches, some of which may reflect anomalies in the training data due to noise or outliers. Tree pruning addresses this issue of overfitting the data by removing the least reliable branches (using statistical measures). This generally results in a more compact and reliable decision tree that is faster and more accurate in its classification of data.

The drawback of using a separate set of tuples to evaluate pruning is that it may not be representative of the training tuples used to create the original decision tree. If the separate set of tuples are biased or distorted, then using them to evaluate the pruned tree would not be a good indicator of the pruned tree's classification accuracy. Furthermore, using a separate set of tuples to evaluate pruning means there are less tuples to use for creation and testing of the tree. While this is considered a drawback in machine learning, it may not be so in data mining due to the availability of larger data sets.

Que 16. Write a short note on

1) precision
2) Recall

3) F-Measure

Precision: It tells how many of the returned samples are correct

$$\mathbf{Precision = TP/(TP+FP)}$$

Recall: It tells how many of the positive samples does the model return.

$$\mathbf{Recall = TP/(TP+FN)}$$

F Measure $= (2*TP)/\ (2*TP+FP+FN)$

$$= (2 * (\mathit{PRECISION} * \mathit{RECALL}))/\ (\mathit{PRECISION} + \mathit{RECALL})$$

F Measure represents harmonic mean between recall and precision.

TP: True positive: Corresponds to number of positive samples correctly predicted by the classification model

FN: false negative: Corresponds to number of positive samples wrongly predicted as negative by the classification model

FP: false positive: Corresponds to number of negative samples wrongly predicted as positive by the classification model.

TN: True negative: Corresponds to number of negative examples correctly predicted by the classification model.

Que 17: Explain issues regarding classification and prediction:

The major issue of Classification and Prediction are:

- **Data Cleaning** – Data cleaning involves removing the noise and treatment of missing values. The noise is removed by applying smoothing techniques and the problem of missing values is solved by replacing a missing value with most commonly occurring value for that attribute.

- **Relevance Analysis** – Database may also have the irrelevant attributes. Correlation analysis is used to know whether any two given attributes are related.

- **Data Transformation and reduction** − The data can be transformed by any of the following methods.
    - **Normalization** − The data is transformed using normalization. Normalization involves scaling all values for given attribute in order to make them fall within a small specified range. Normalization is used when in the learning step, the neural networks or the methods involving measurements are used.
    - **Generalization** − The data can also be transformed by generalizing it to the higher concept. For this purpose we can use the concept hierarchies.
- **Accuracy** − Accuracy of classifier refers to the ability of classifier to predicts the class label correctly. and the accuracy of the predictor refers to how well a given predictor can guess the value of predicted attribute for a new data.
- **Speed** − This refers to the computational cost in generating and using the classifier or predictor.
- **Robustness** − It refers to the ability of classifier or predictor to make correct predictions from given noisy data.
- **Scalability** − Scalability refers to the ability to construct the classifier or predictor efficiently; given large amount of data.

Que 18: What do you mean by eager and lazy learner? What are advantages and disadvantages?

Eager learner: Eager learners, when given a set of training tuples, will construct a generalization (i.e., classification) model before receiving new (e.g., test) tuples to classify. We can think of the learned model as being ready and eager to classify previously unseen tuples.

Lazy learner: Lazy learner waits until the last minute before doing any model construction in order to classify a given test tuple. That is, when given a training tuple, a lazy learner simply stores it (or does only a little minor processing) and waits until it is given a test tuple. It performs generalization after it encounters a new test tuples. Lazy learner as also called instance based learner.

Eager learner	lazy learner
1. Simply stores training data (or only minor processing) and waits until it is given a test tuple	**1.** Given a set of training tuples, constructs a classification model before receiving new (e.g., test) data to classify
2. More time for training but less time for predicting	**2.** Less time in training but more time in predicting
3. Decision tree and naïve bayes are examples of eager learner classifier	**3.** Case Base approach and K-nearest neighbor are examples of lazy learners
4. must commit to a single hypothesis (i.e model) that covers the entire instance space	**4.** Lazy method effectively uses a richer hypothesis space since it uses many local linear functions to predict the output of target function
5. Requires less storage as compared to lazy learners	**5.** Requires efficient storage techniques and more efficient indexing is required.
6. Computationally less expensive as compared to lazy learner	**6.** Computationally expensive
7. Parallel computation is not required.	**7.** Requires parallel computation
8. Doesn't support incremental learning	**8.** Supports incremental learning

PART 2

MODEL OVER FITTING

Que 12: Write a short note on the following:

1) Training error
2) Generalization error
3) Model under-fitting.
4) Model Over-fitting.

Training errors is the number of misclassification errors committed on training records. Means, A Model may not correctly predict the class label of unknown samples which is same as training data.

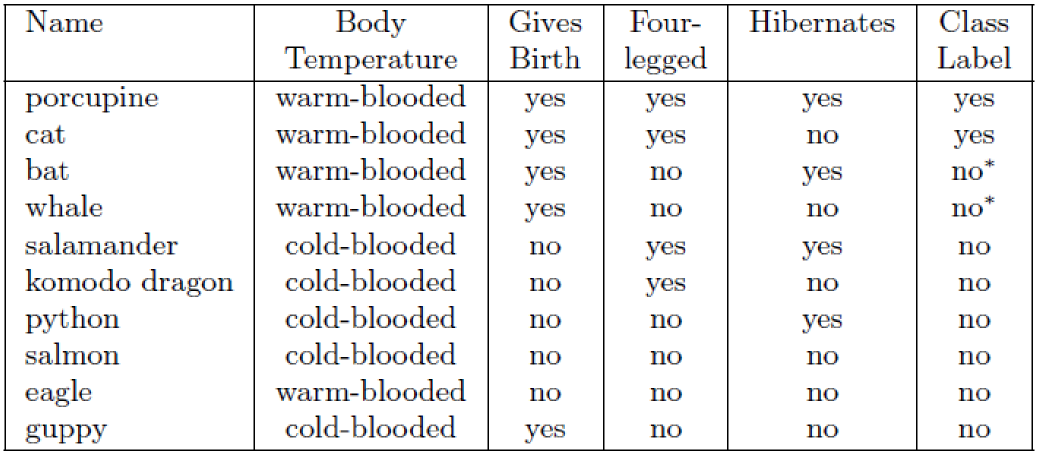
Generalization errors: It is the expected error of the model on previously unseen records. Means, A Model may not correctly predict the class label of unknown samples which is different from training data.

Model Under-fitting: The training and generalization error rates are large when the size of the tree is very small. This situation is known as **model underfitting**.

Model Over-fitting: When the tree becomes large, the test error rate increases and training error rate decreases. This situation is known as **model over-fitting**.

Que 13: What do you mean by over fitting due to present of noise?

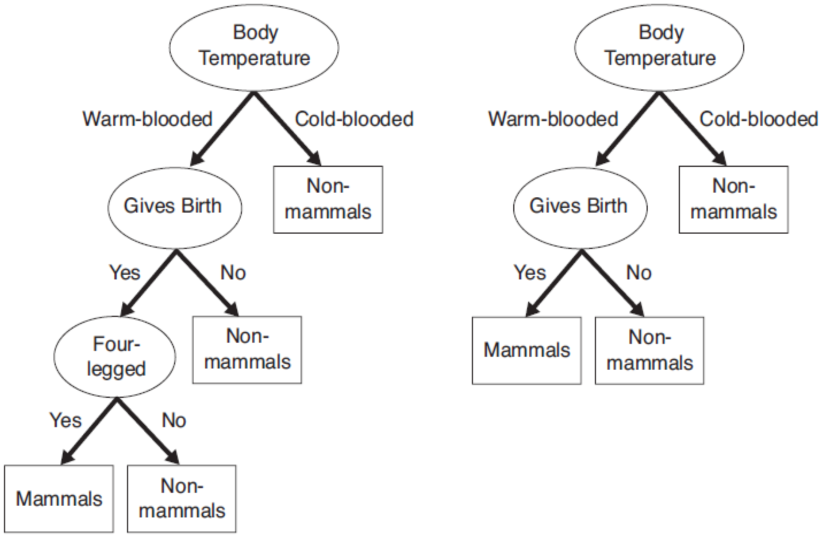
- Consider the training and test sets for the mammal classification problem. Two of the ten records are mislabeled. Bats and whales are classified as non mammals instead of mammals.



Name	Body Temperature	Gives Birth	Four-legged	Hibernates	Class Label
porcupine	warm-blooded	yes	yes	yes	yes
cat	warm-blooded	yes	yes	no	yes
bat	warm-blooded	yes	no	yes	no*
whale	warm-blooded	yes	no	no	no*
salamander	cold-blooded	no	yes	yes	no
komodo dragon	cold-blooded	no	yes	no	no
python	cold-blooded	no	no	yes	no
salmon	cold-blooded	no	no	no	no
eagle	warm-blooded	no	no	no	no
guppy	cold-blooded	yes	no	no	no

- The class label for {name='human', body-temperature='warm-blooded', gives berth='yes', four-legged='no', hibernates='no'} is non-mammals from above decision tree. But humans are mammals. The prediction is wrong due to presence of noise in data.

- Even the training error is zero the model has generalization errors.



(a) Model M1

(b) Model M2

Model M1 is Over trained (Over fitted) which wrongly classifies human as non-mammal .Model M2 (which is not over trained) correctly predicts human as mammal.

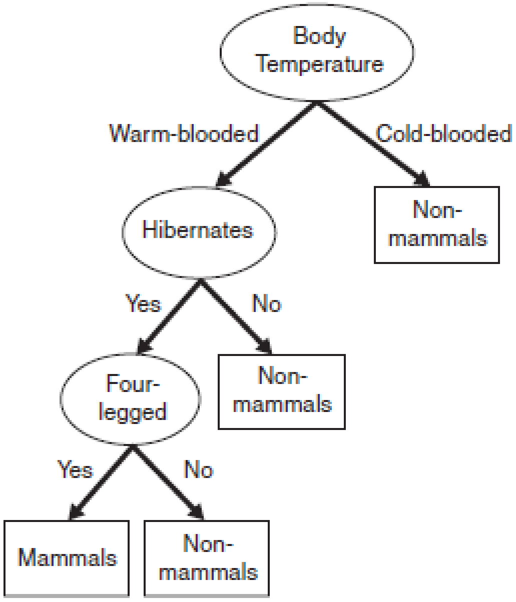
Que14: what do you mean by over fitting due to lack of representative samples?

When a model is built on very few training data, Over-fitting occurs. Means, lack of representative sample in training data may lead to a tree which wrongly predicts the class label of unknown sample.

For Example, Consider below table:

Name	Body Temperature	Gives Birth	Four-legged	Hibernates	Class Label
salamander	cold-blooded	no	yes	yes	no
guppy	cold-blooded	yes	no	no	no
eagle	warm-blooded	no	no	no	no
poorwill	warm-blooded	no	no	yes	no
platypus	warm-blooded	no	yes	yes	yes

The corresponding model is,



From above example, humans {*name='human', body-temperature='warm-blooded', gives berth='yes', four-legged='no', hibernates='no'*} are wrongly classified as non mammals.

EVALUATING THE PERFORMANCE OF CLASSIFIER

Que 15: What are the methods for evaluating performance of classifier?

1) **Holdout Method**
2) **Random Sampling**
3) **Cross-Validation**
   a. **Two fold cross validation**
   b. **K-fold cross validation**
   c. **Leave-one-out approach**
4) **Bootstrap**

- **Holdout Method**

  In this method the original data sets is divided into two parts, 50% or 2/3rd of original data is considered as training sets and another 50% or 1/3rd of original data as test sets respectively. Now, the classification model is trained on training tests and then applied on test sets. The performance of the classification algorithm is based on number of correct predictions made on the test set.

  Limitations

  1) Less number of samples for training(since the original samples are spitted)
  2) The model is highly dependent on the composition of the training and test sets

- **Random Sampling**

  Multiple repetition of holdout method is known as random sampling. Here the original data is divided randomly into training sets and test sets and the accuracy is calculated as in holdout method. This random sampling is then repeated k times and the accuracy is calculated for each time. The overall accuracy is:

$$acc_{\text{sub}} = \sum_{i=1}^{k} acc_i / k.$$

  Here acc$_i$ is the model accuracy during *i* th iteration

  Limitations

  1) Less number of samples for training(since the original samples are spitted)
  2) A record may be used more than once in training and test tests.

- **Cross-Validation**

  There are three variations of cross-validation approach

  a) Two fold cross validation

  In this approach data is partitioned into two parts. The first part is considered as training set and the second part as test set. Now they are swapped and the first part is considered as test set and second one as training set. The total error is the sum of both the errors.

b) K-fold cross validation

In this approach the data is partitioned into k subsets. One of the partitions is considered as test set and remaining sets are considered as training set. This process is repeated k times and the total error is the sum of all the k runs.

c) Leave-one-out approach

In this approach one record is considered as test set and rest of the samples are considered as training set. This process is repeated k times (k= number of records) and the total error is the sum of all the k runs. But this process is computationally very expensive.

- **Bootstrap**

  In this approach a record may be sampled more than once. Means a record when sampled is again kept back in the original data. So it is likely that the record may be sampled again and again. Consider original data of size N. The probability of a record to be chosen as bootstrap sample is $1-(1-1/N)^{N}$ .When the Size of N is very large then the probability is $1-e^{-1}$. The sampling is repeated B times to generate b bootstrap samples.

Annexure

These questions may not be in syllabus but some of these questions were asked in previous question papers.

Que 16: What do you mean by Bayesian classifier and bayes theorem? Explain how Bayesian classifier can be used for classification.

Bayesian classifiers

Bayesian classifier is the classification technique which uses **bayes theorem** for identifying unknown class label. Bayesian classifier can be implemented in two ways. They are:

i) Naïve Bayes Classifier

ii) Bayesian belief networks

Bayes Theorem

Bayes Theorem can be given as:

$$P\ (Y\,|\,X) = \frac{P(X|Y)*P(Y)}{P(X)}$$

Example: A conversation was held in a train with a person about long hair. There are 50% men and 50% women. In most of the cases, this conversation will be initiated by women. Suppose that 75% of women have long hair and 15% of men have long hair. We need to identify who initiated the conversation (men or women),

Bayes theorem can be written as:

$$P\ (W\,|\,L) = \frac{P(L|W)*P(W)}{P(L)}$$

where P (W|L) represents probability of women with long hair initiating the conversation.

P (L|W) represents probability of women with long hair. (75% of women have long hair. So, the probability is 0.75)

P (W) represents probability of women population. (there are 50% women. So, the probability is 0.50)

P (L) represents total probability of long hair.

Where P(L)= P(L|W)* P(W) + P(L|M)* P(M)

$$P\ (W|L) = \frac{0.75*0.50}{0.75*0.50+0.15*0.50} = \frac{5}{6}$$

=0.83 (83%)

The probability that a woman initiating the conversation was 83% and men initiating the conversation was 17%

Using the Bayes Theorem for classification

Let **X** denotes the attribute set and **Y** denote the class label or variable.

$$P\ (Y|X) = \frac{P(X|Y)*P(Y)}{P(X)}$$

Where **P (Y|X)** is the conditional probability also known as posterior probability for **Y**

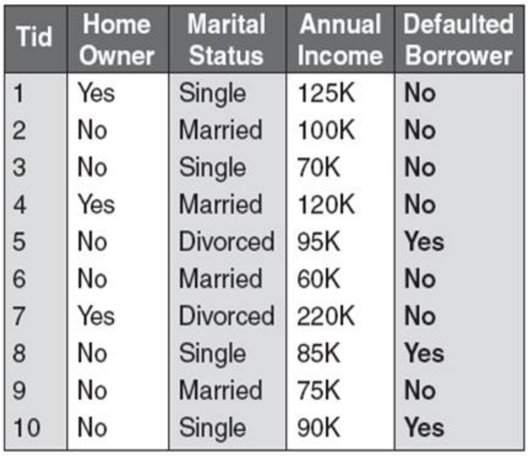
P(Y) is the priori probability

During the training phase, we need to learn the posterior probabilities **P (Y|X)** for every combination of **X** and **Y** based on information gathered in the training data.

Consider the task of predicting whether a loan borrower will default on their payments. The below table is a training set with following attributes: Home Owner, Marital Status and Annual Income. Loan borrowers who paid the loan are classified as **no** and who defaulted is classified as **yes.**

Let us consider a test record **X= (Home Owner=No, Marital Status=Married, Annual Income=120K)**

In order to classify the record, we need to compute P (Yes |X) and P (No |X). If P (Yes |X) > P (No |X), then the record is classified as Yes, else, No.



Tid	Home Owner	Marital Status	Annual Income	Defaulted Borrower
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Que 17: Explain Naïve Bayes Classifier with an example.

A Naïve Bayes Classifier estimates the class conditional probability by assuming that the attributes are conditionally independent, given the class label y. The conditional independence assumption can be formally stated as follows:

$$P\ (X|Y=y) = \prod_{i=1}^{d} P(\ Xi|Y = y)$$

To classify a test record, the naïve bayes classifier computes the posterior probability for class label Y as follows:

$$P\ (Y|X) = \frac{\prod_{i=1}^{d} P(\ Xi|Y=y)*P(Y)}{P(X)}$$

P(X) is fixed for every Y.

Estimating conditional probability for categorical attribute

P (X_i= x_i |Y=y) is estimated according to the fraction of training instances in class y

Example

The conditional probability for P (Home Owner=Yes | No) = $\frac{3}{7}$ (7 represents the total number of samples with class label=No and 3 represents the total number of samples with Home Owner=Yes and class label =No)

Estimating conditional probability for continuous attribute

$$P\ (X_i{=}x_i\ |\ Y{=}y) = \frac{1}{\sqrt{2\Pi\sigma ij}}\, exp^{\frac{-(xi-\mu ij)2}{2\sigma ij2}}$$

Where μ_{ij} or $\overline{x}$ is the mean of values in a continuous attribute

$$\overline{x} = \frac{\sum_{i=1}^{n} xi}{n}$$

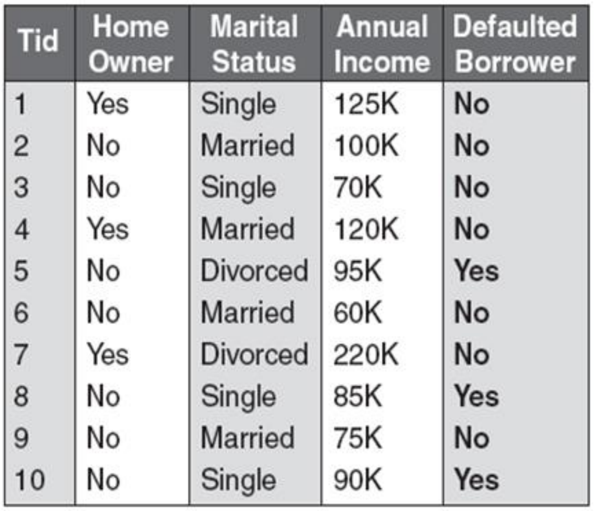
σ_{ij}^2 or s^2 is the variance or standard deviation

$$S^2 = \frac{\sum_{i=1}^{n}(xi-\overline{x})2}{n(n-1)}$$

Example

Consider the below loan data set and predict the class label of a test record

X= (Home owner=No, Marital Status=married, annual income=120K)



Tid	Home Owner	Marital Status	Annual Income	Defaulted Borrower
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

There are three records that belong to the class **Yes** and seven records that belong to class **No** P(Y) can be calculated by

Therefore, P (Yes) = $\frac{3}{10}$

P (No) = $\frac{7}{10}$

Now calculate $P(\,\mathrm{Xi}|Y = y)$,

For this we need to compute the posterior probabilities P (No| X) and P (Yes| X) Home owner and marital status are categorical attributes. The conditional probability can be calculated as follows:

$$P\ (\text{Home owner=No}|\ \text{No}) = \frac{4}{7}$$

$$P\ (\text{Home owner=No}|\ \text{Yes}) = \frac{3}{3} = 1$$

$$P\ (\text{Marital status= Married}|\ \text{No}) = \frac{4}{7}$$

$$P\ (\text{Marital status= Married}|\ \text{Yes}) = \frac{0}{3} = 0$$

But annual income is a continuous attribute, so, we need to use the below formula for calculating the conditional probability for (annual income= 120K)

$$P\ (X_i=x_i\ |Y=y) = \frac{1}{\sqrt{2\Pi\sigma ij}}\, exp^{\frac{-(xi-\mu ij)2}{2\sigma ij2}}$$

$$P\ (\text{annual income= 120K}|\ \text{No}) = \frac{1}{\sqrt{2\Pi\sigma ij}}\, exp^{\frac{-(xi-\mu ij)2}{2\sigma ij2}}$$

$$\mu ij = \frac{125+100+70+120+60+220+75}{7} = 110$$

$$\sigma_{ij}^2 = \frac{(110-125)2+(110-100)2+(110-70)2+(110-120)2+(110-60)2+(110-220)2+(110-75)2}{7(7-1)}$$

$$= 420.23$$

$$\sigma_{ij} = \sqrt{420.23} = 20.49$$

Substituting the values of μij, σ_{ij}^2 and σ_{ij} in the above equation, we get the value of P (annual income= 120K| No) = 0.017

$$P\ (\text{annual income= 120K}|\ \text{Yes}) = \frac{1}{\sqrt{2\Pi\sigma ij}}\, exp^{\frac{-(xi-\mu ij)2}{2\sigma ij2}}$$

$$\mu ij = \frac{95+85+90}{3} = 90$$

$$\sigma_{ij}^2 = \frac{(90-95)2+(90-85)2+(90-90)2}{3(3-1)} = 8.12$$

$\sigma_{ij} = \sqrt{8.12} = 2.84$

Substituting the values of μij, σ_{ij}^2 and σ_{ij} in the above equation, we get the value of P (annual income= 120K| Yes) = 0.014

Therefore,

P (X| No) = P (Home owner=No| No) * P (Marital status= Married| No) *
P (annual income= 120K| No)

$$= \frac{4}{7} * \frac{4}{7} * 0.017 = 0.0055$$

P (X| Yes) = P (Home owner=No| Yes) * P (Marital status= Married| Yes) *
P (annual income= 120K| Yes)

$$= 1 * 0 * 0.014 = 0$$

The posterior probability for class No is P (No| X) =α * $\frac{7}{10}$ * 0.0055 = 0.00385α

The posterior probability for class Yes is P (Yes| X) =α * $\frac{3}{10}$ * 0 = 0

Where α=$\frac{1}{P(X)}$ is a constant term

Since P (No| X) > P (Yes| X), the record is classified as **No**

Que 18: What are the characteristics of Naïve bayes classifier? Explain how correlated attributes affect the performance of a naïve bayes classifier.

1. They are robust to isolated noise points because such points are averages out when estimating conditional probabilities from data.
2. It can also handle missing values by ignoring the example during model building and classification.
3. They are robust to irrelevant attributes.
4. Correlated attributes can degrade the performance of naïve bayes classifiers because the conditional independence assumption no longer holds for such attributes.

Example

Consider the following probabilities:

P (A=0 | Y=0) =0.4, P (A=1 | Y=0) =0.6,

P (A=0 | Y=1) =0.6, P (A=1 | Y=1) =0.4

where A is a binary attribute and Y is a binary class variable. Suppose there is another attribute B which is perfectly correlated with A when Y=0, but is independent of A when Y=1. Let us assume that class conditional probabilities for B are same as for A.

Given a record (A=0, B=0), the posterior probabilities are calculated as follows:

$$P(Y=0|A=0, B=0)= \frac{P(A=0|Y=0)P(B=0|Y=0)P(Y=0)}{P(A=0)P(B=0)}$$

$$=\frac{0.16*P(Y=0)}{P(A=0)P(B=0)}$$

$$P(Y=1|A=0, B=0)= \frac{P(A=0|Y=1)P(B=0|Y=1)P(Y=1)}{P(A=0)P(B=0)}$$

$$=\frac{0.36*P(Y=1)}{P(A=0)P(B=0)}$$

If P(Y=0) =P(Y=1), then the record is classified to class 1. However the truth is,

P (A=0, B=0 | Y=0) = P (A=0|Y=0) =0.4

Because A and B are perfectly correlated when Y=0. The result for Y=0 is

$$P(Y=0|A=0, B=0) = \frac{P(A=0,B=0|Y=0)P(Y=0)}{P(A=0)P(B=0)}$$

$$=\frac{0.4*P(Y=0)}{P(A=0)P(B=0)}$$

Which is larger than that for y=1. The record should have been classified as class 0.

Que 19: Write a short note on Bayes Error Rate

Suppose we know the true probability distribution that governs P(X|Y). The Bayesian classification method allows us to determine the ideal decision boundary for the classification task.

Example

Consider the task of identifying alligators and crocodiles based on their lengths. The average length of an adult crocodile is about 15 feet and alligator is 12 feet. Assuming that their length x follows a Gaussian distribution with a standard deviation equal to 2 feet, we can express their class conditional probabilities as follows:

P (X| crocodile) = $\frac{1}{\sqrt{2\Pi\ 2}} exp\, \frac{-1}{2}^{(\frac{x-15}{2})2}$

P (X| alligator) = $\frac{1}{\sqrt{2\Pi\ 2}} exp\, \frac{-1}{2}^{(\frac{x-12}{2})2}$

Assuming that the prior probabilities are same

P(X=$\hat{x}$ |Crocodile)= P(X=$\hat{x}$ |alligator)

Using above equations w obtain:

$$(\frac{x-15}{2})^2 = (\frac{x-12}{2})^2$$

Which can be solved to yield x=13.5 i.e., the length< 13.5 are alligators and length> 13.5 are crocodiles.

Que 20: Explain briefly about Bayesian Belief Networks

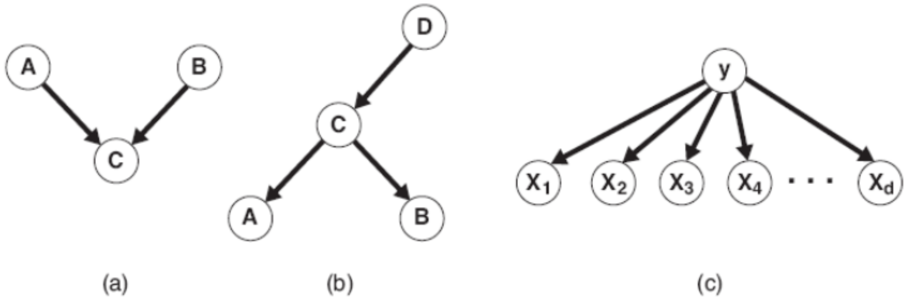
There are some problems with correlated attributes when calculating conditional probabilities using naïve bayes classifier. These problems can be rectified using Bayesian belief networks.

Model Representation

A Bayesian Belief Networks (BBN) provides a graphical representation of the probabilistic relationships among a set of random variables. There are two key elements of a Bayesian network. They are:

1. A directed acyclic graph (DAG) encoding the relationship between the set of variables.
2. A probability table associating each node to its immediate parent nodes.

Consider three random variables A, B and C, in which A and B are independent variables but have direct influence with C as shown below:



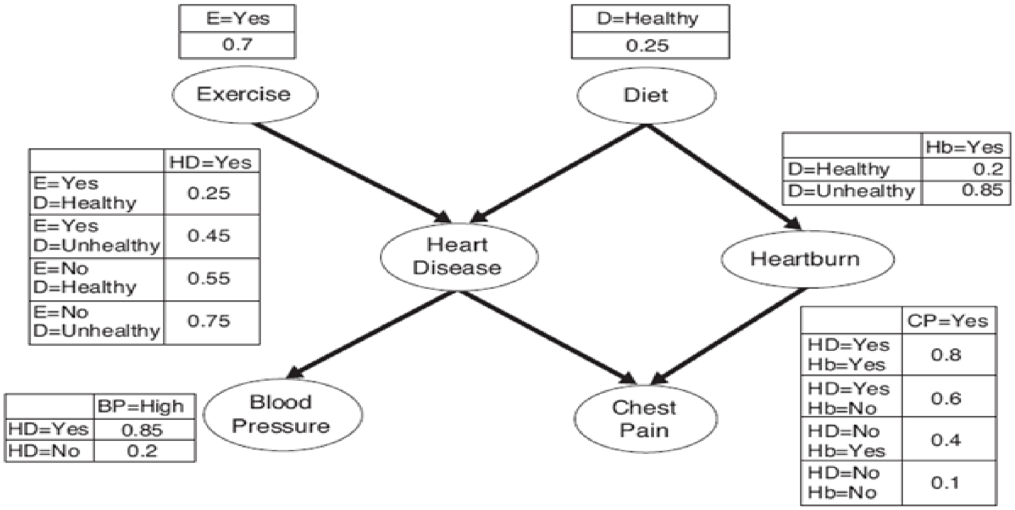
Representing probabilistic relationships using directed acyclic graphs.

If there is a directed arc from X to Y, then X is the **parent** of Y and Y is the **child** of X. If there is a directed path from X to Z, then X is the **ancestor** of Z and Z is a **descendant** of X.

Here **nodes** represent **variables or attributes** and **arcs** represent the **relationship** between those variables or attributes.

Each node is associated with a probability table basing on the following conditions:

1. If a node X does not have any parents, then the table contains only the prior information P(X)
2. If a node X has only one parent, then the table contains the conditional probability P(X|Y)
3. If a node X has multiple parents, then the table contains the conditional probability P (X|Y1,Y2,….., Y_k)



E=Yes
0.7

D=Healthy
0.25

	HD=Yes

E=Yes
D=Healthy
	0.25

E=Yes
D=Unhealthy
	0.45

E=No
D=Healthy
	0.55

E=No
D=Unhealthy
	0.75

	Hb=Yes
D=Healthy	0.2
D=Unhealthy	0.85

	BP=High
HD=Yes	0.85
HD=No	0.2

	CP=Yes

HD=Yes
Hb=Yes
	0.8

HD=Yes
Hb=No
	0.6

HD=No
Hb=Yes
	0.4

HD=No
Hb=No
	0.1

A Bayesian belief network for detecting heart disease and heartburn in patients.

The above figure is an example of a Bayesian network for modeling patients with heart disease and heart burn. Each variable in the diagram is assumed to be a binary valued.

Conditions

a) Exercise and diet are risk factors of heart disease.
b) Blood pressure and chest pain are the symptoms of heart disease.
c) Diet is the risk factor for heart burn.
d) Chest pain is the symptom for heart burn.

The nodes exercise and diet has only prior probability since it does not have any parent node. The nodes heart disease and heart burn and their symptoms contain conditional probabilities.

Example

P (Heart Disease=No | Exercise=No, Diet=healthy)

=1-P (Heart Disease=Yes | Exercise=No, Diet=healthy)

=1-0.55 =0.45

Model building

Model building in Bayesian networks involves two steps:

1) Creating the structure of the network.
2) Estimating the probability values in the tables associated with each node.

Algorithm for generating a Bayesian Network

```
1: let T=($X_1$,$X_2$,....,$X_n$) denote a total order of variables.
2: for j=1 to d do
3: let $X_{T(j)}$ denote the jth highest order variable in T
4: let □($X_{T(j)}$)= {$X_{T(1)}$,$X_{T(2)}$,...,$X_{T(j-1)}$} denote the set of variables preceding $X_{T(j)}$.
5: Remove the variables from □($X_{T(j)}$) that do not affect $X_j$ (using prior knowledge).
6: Create an arc between □($X_{T(j)}$) and the remaining variables in □($X_{T(j)}$).
7: end for.
```

Example

Consider the variables as shown in the above figure. After performing the first step, let us assume that the variables are ordered in the following way: (E, D, HD, HB, CP, BP)

- P(D|E) is simplified to P(D). //since D and E are independent variables
- P(HD|E, D) cannot be simplified. //both E and D are dependent on HD
- P(HB|E, D, HD) is simplified to P(HB|D) //since E and HD are not dependent on HB
- P(CP|E, D, HD, HB) is simplified to P(CP|HB, HD) //since E and D are not dependent on CP
- P(BP|E, D, HD, HB, CP) is simplified to P(BP|HD) //since D, E, HB, CP are not dependent on CP

Basing on these conditional probabilities, we can create arcs between the nodes (E, HD), (D, HD), (D, HB), (HD, CP), (HB, CP) and (HD, BP). By connecting these arcs a network is formed as above fig.

Characteristics of BBN

1. BBN provides an approach for capturing the prior knowledge of a particular domain using a graphical model. The network can also be used to encode casual dependencies among variables.
2. Constructing the network can be time consuming and requires a large amount of effort. However, once the structure of the network has been determined, adding a new variable is quite straightforward.
3. Bayesian networks are well suite to dealing with incomplete data. Instances with missing values can be handled by summing or integrating the probabilities over all possible values of the attribute.
4. Because the data is combined probabilistically with prior knowledge, the method is quite robust to model over fitting.