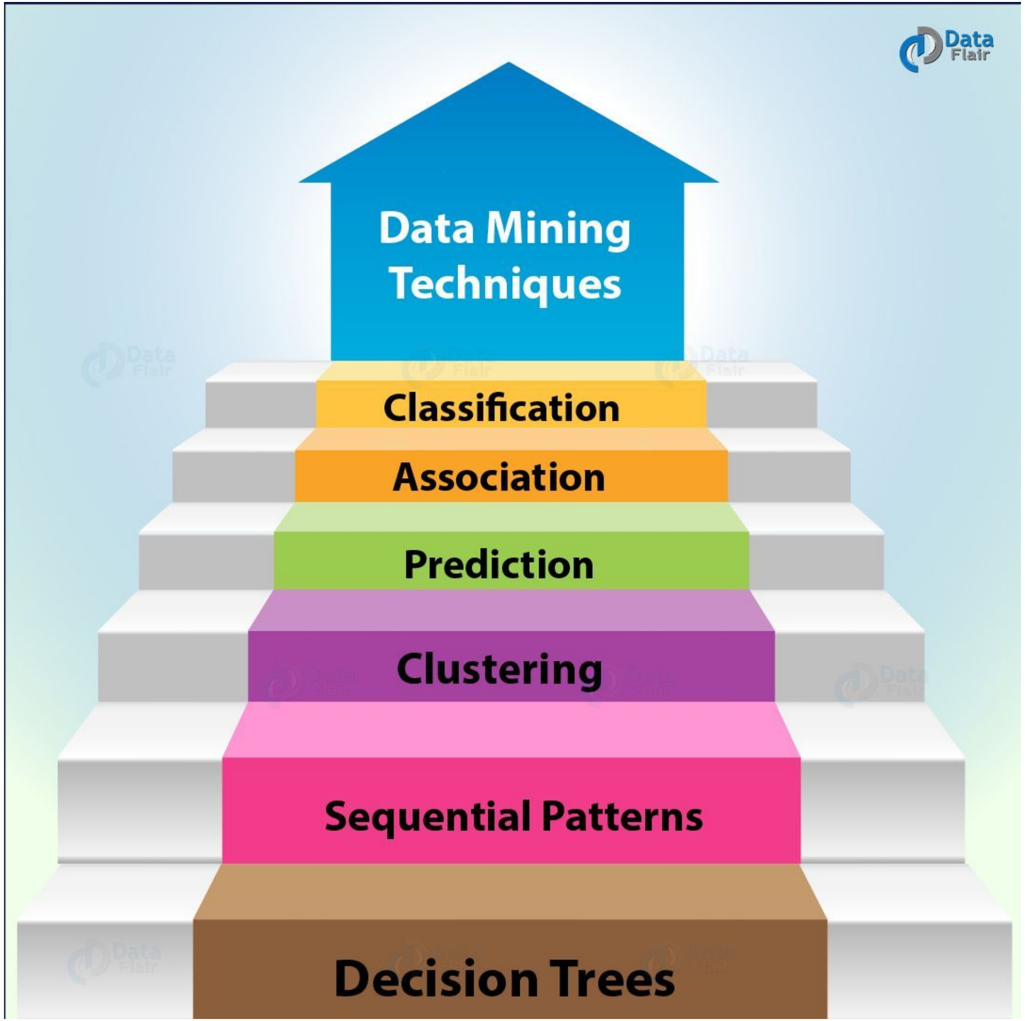
AMIRAJ
COLLEGE OF ENGINEERING & TECHNOLOGY

CHAPTER 6
CLASSIFICATION AND PREDICTION

Mining: Classification and Prediction

SUBJECT:DMBI CODE:2170715	PREPARED BY: ASST.PROF.NENSI KANSAGARA (CSE DEPARTMENT,ACET)	AMIRAJ COLLEGE OF ENGINEERING & TECHNOLOGY



Data Mining Techniques

Classification

Association

Prediction

Clustering

Sequential Patterns

Decision Trees

WHAT IS CLASSIFICATION & PREDICTION?

- There are two forms of data analysis that can be used for extracting models describing important classes or to predict future data trends. These two forms are as follows –
  - Classification
  - Prediction
- Classification models predict categorical class labels; and prediction models predict continuous valued functions. For example, we can build a classification model to categorize bank loan applications as either safe or risky, or a prediction model to predict the expenditures in dollars of potential customers on computer equipment given their income and occupation.

WHAT IS CLASSIFICATION?

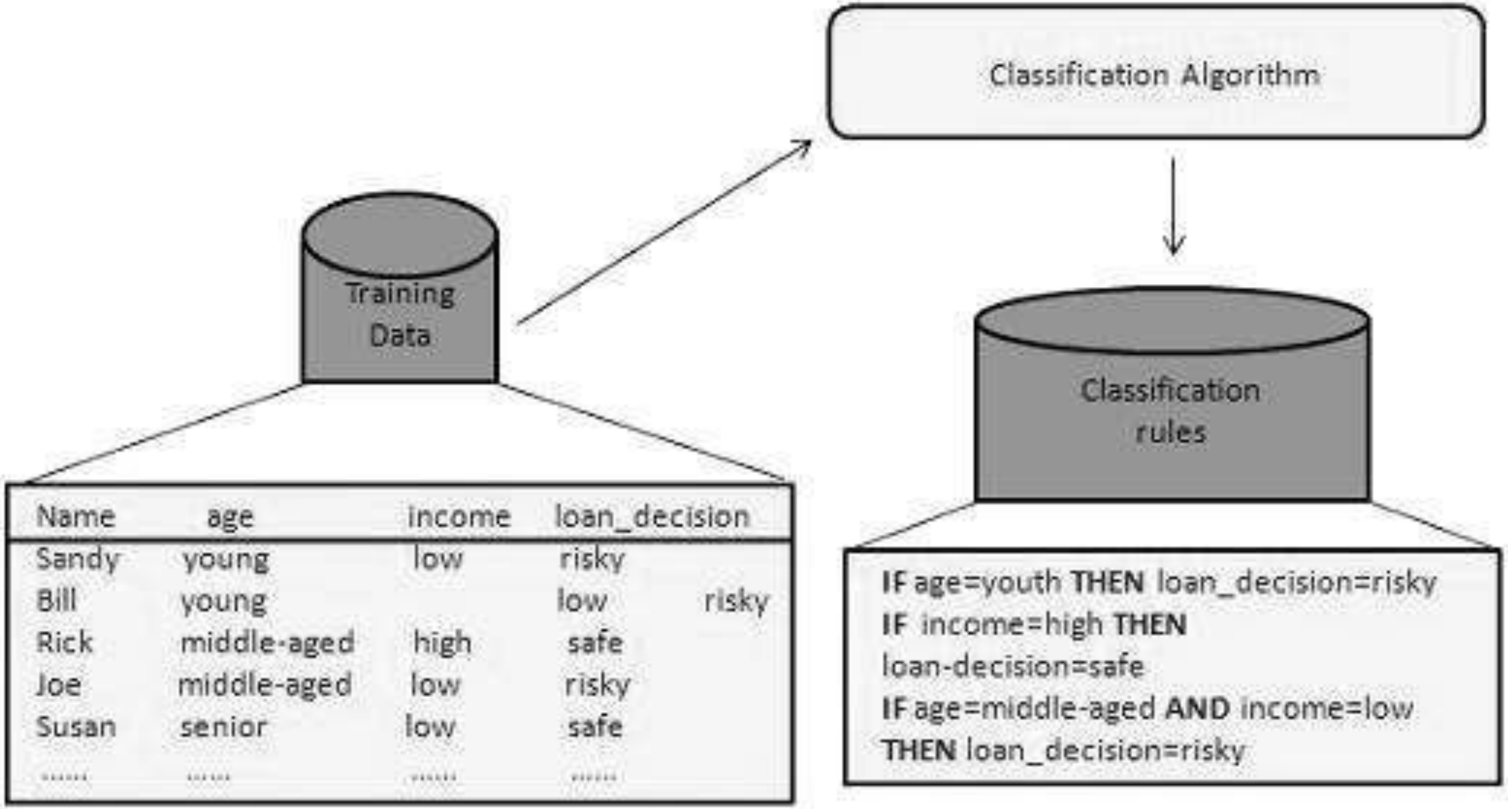
- Following are the examples of cases where the data analysis task is Classification –
  - A bank loan officer wants to analyze the data in order to know which customer (loan applicant) are risky or which are safe.
  - A marketing manager at a company needs to analyze a customer with a given profile, who will buy a new computer.
- In both of the above examples, a model or classifier is constructed to predict the categorical labels. These labels are risky or safe for loan application data and yes or no for marketing data.

WHAT IS PREDICTION?

- Following are the examples of cases where the data analysis task is Prediction −
- Suppose the marketing manager needs to predict how much a given customer will spend during a sale at his company. In this example we are bothered to predict a numeric value. Therefore the data analysis task is an example of numeric prediction. In this case, a model or a predictor will be constructed that predicts a continuous-valued-function or ordered value.
- Note − Regression analysis is a statistical methodology that is most often used for numeric prediction

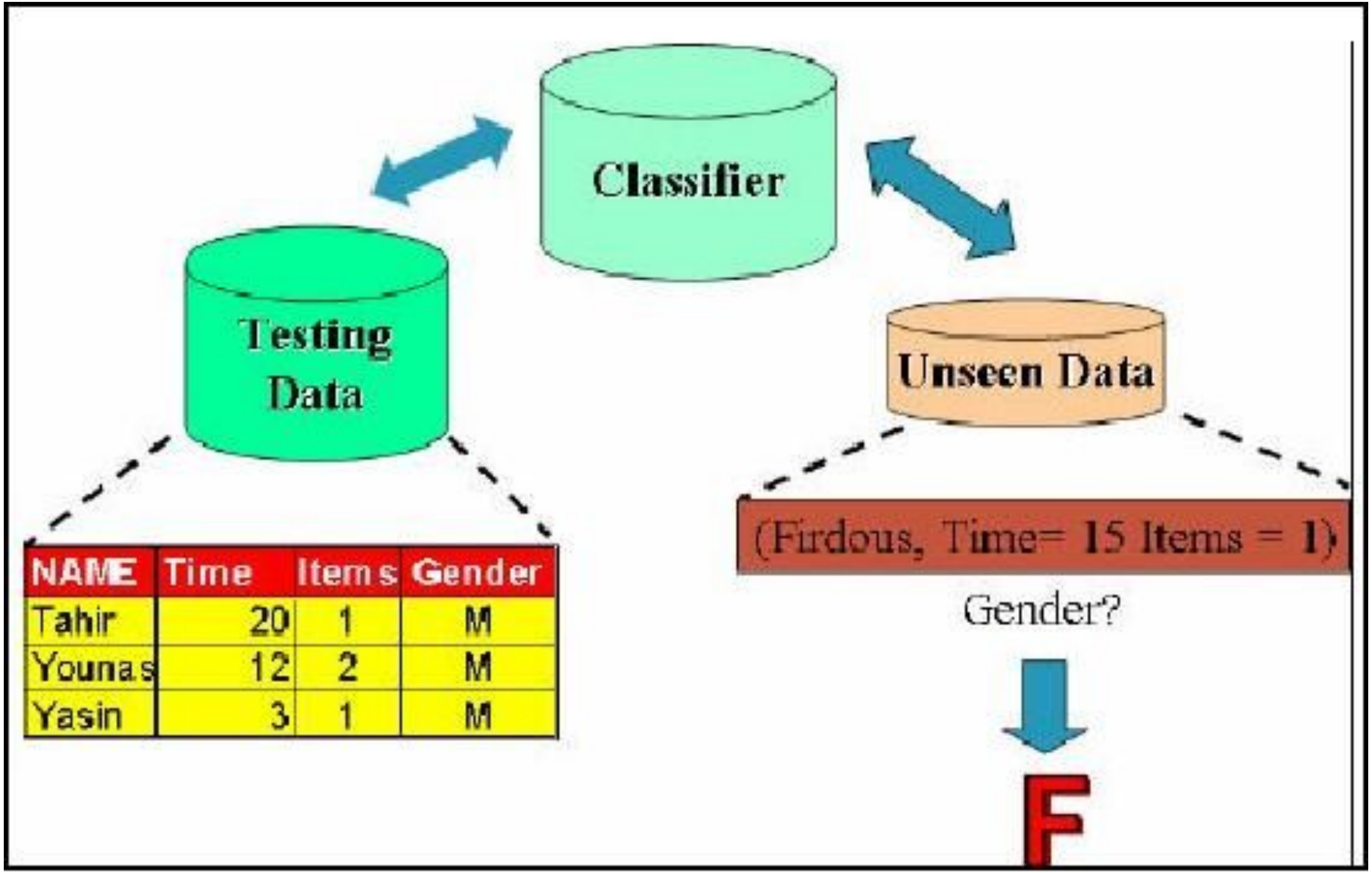
HOW DOES CLASSIFICATION WORKS?

- With the help of the bank loan application that we have discussed above, let us understand the working of classification. The Data Classification process includes two steps −
  - Building the Classifier or Model
  - Using Classifier for Classification
- Building the Classifier or Model
  - This step is the learning step or the learning phase.
  - In this step the classification algorithms build the classifier.
  - The classifier is built from the training set made up of database tuples and their associated class labels.
  - Each tuple that constitutes the training set is referred to as a category or class. These tuples can also be referred to as sample, object or data points.



Name	age	income	loan_decision	
Sandy	young	low	risky	
Bill	young		low	risky
Rick	middle-aged	high	safe	
Joe	middle-aged	low	risky	
Susan	senior	low	safe	
......	......	......	......	

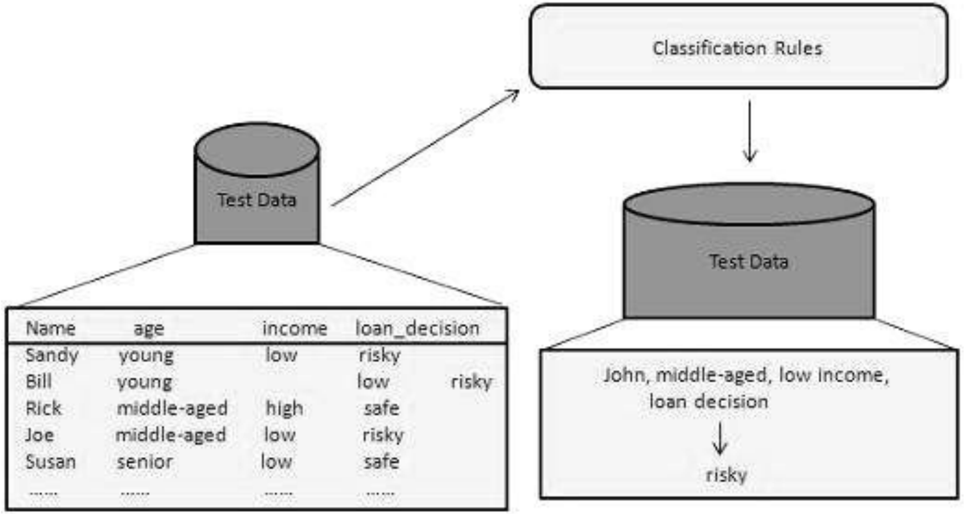
IF age=youth **THEN** loan_decision=risky
IF income=high **THEN**
loan-decision=safe
IF age=middle-aged **AND** income=low
THEN loan_decision=risky



NAME	Time	Items	Gender
Tahir	20	1	M
Younas	12	2	M
Yasin	3	1	M

USING CLASSIFIER FOR CLASSIFICATION

❖ In this step, the classifier is used for classification. Here the test data is used to estimate the accuracy of classification rules. The classification rules can be applied to the new data tuples if the accuracy is considered acceptable.



Name	age	income	loan_decision	
Sandy	young	low	risky	
Bill	young		low	risky
Rick	middle-aged	high	safe	
Joe	middle-aged	low	risky	
Susan	senior	low	safe	
......	......	......	......	

ISSUE REGARDING CLASSIFICATION & PREDICTION

- The major issue is preparing the data for Classification and Prediction. Preparing the data involves the following activities −
  - 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.

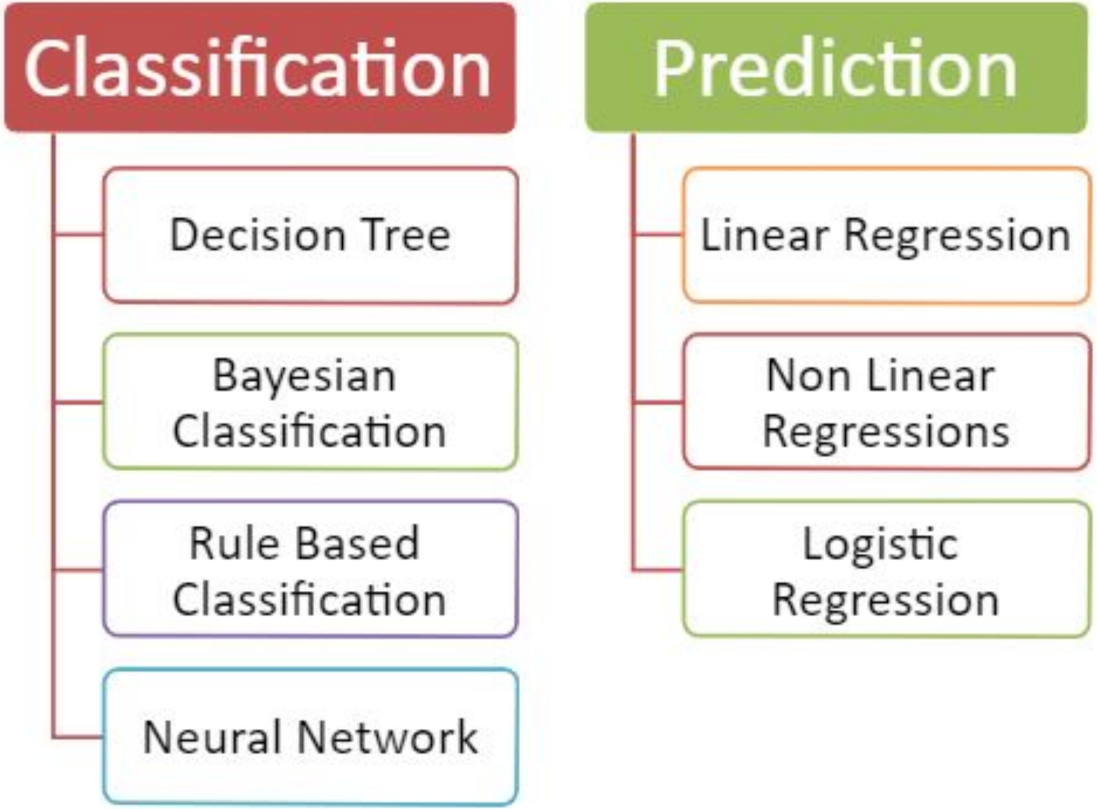
ISSUE REGARDING CLASSIFICATION & PREDICTION

- 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.

COMPARISON OF CLASSIFICATION AND PREDICTION METHOD

- Here is the criteria for comparing the methods of Classification and Prediction −
  - Accuracy − Accuracy of classifier refers to the ability of classifier. It predict 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.
  - Interpretability − It refers to what extent the classifier or predictor understands.

CLASSIFICATION & PREDICTION METHOD



Classification

- Decision Tree
- Bayesian Classification
- Rule Based Classification
- Neural Network

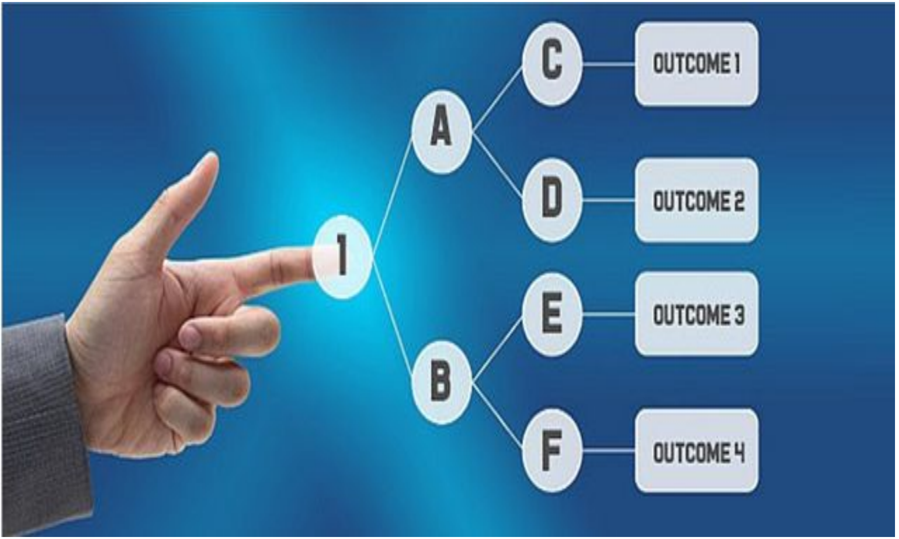
Prediction

- Linear Regression
- Non Linear Regressions
- Logistic Regression

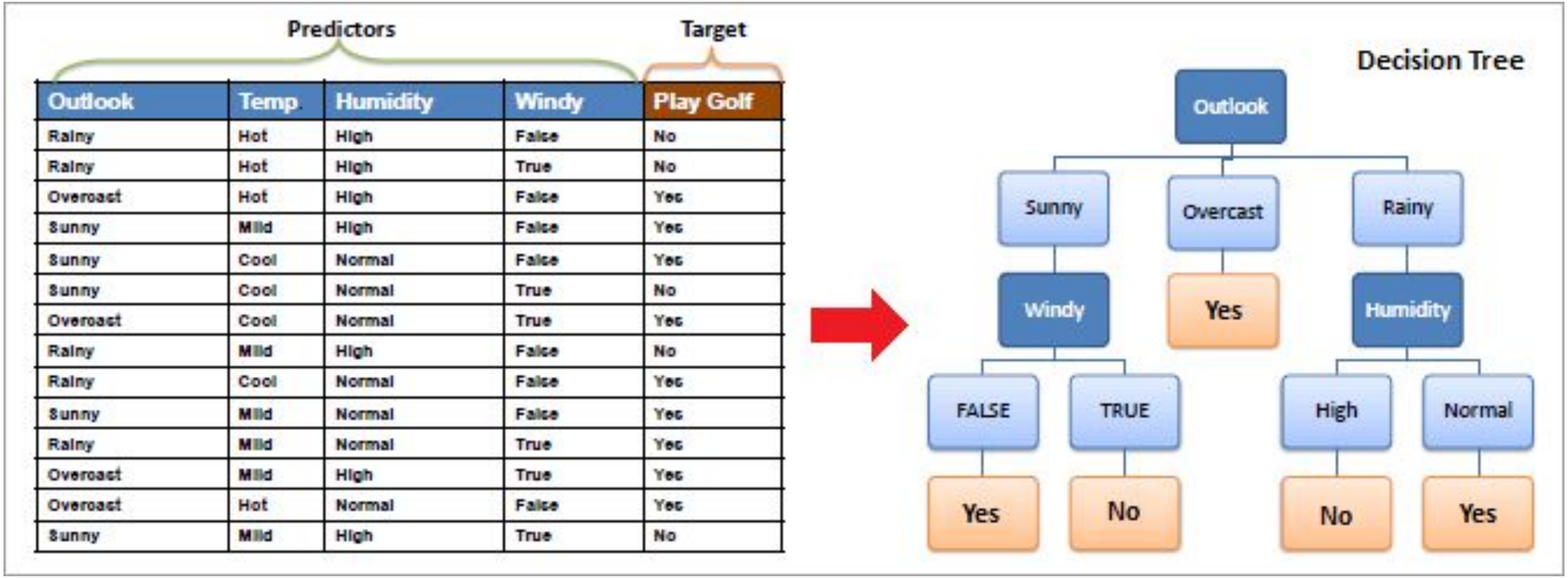
CLASSIFICATION METHODS

DECISION TREE

- Decision Tree Mining is a type of data mining technique that is used to build Classification Models. It builds classification models in the form of a tree-like structure, just like its name. This type of mining belongs to supervised class learning.
- In supervised learning, the target result is already known. Decision trees can be used for both categorical and numerical data. The categorical data represent gender, marital status, etc. while the numerical data represent age, temperature, etc.



EXAMPLE OF DECISION TREE



Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

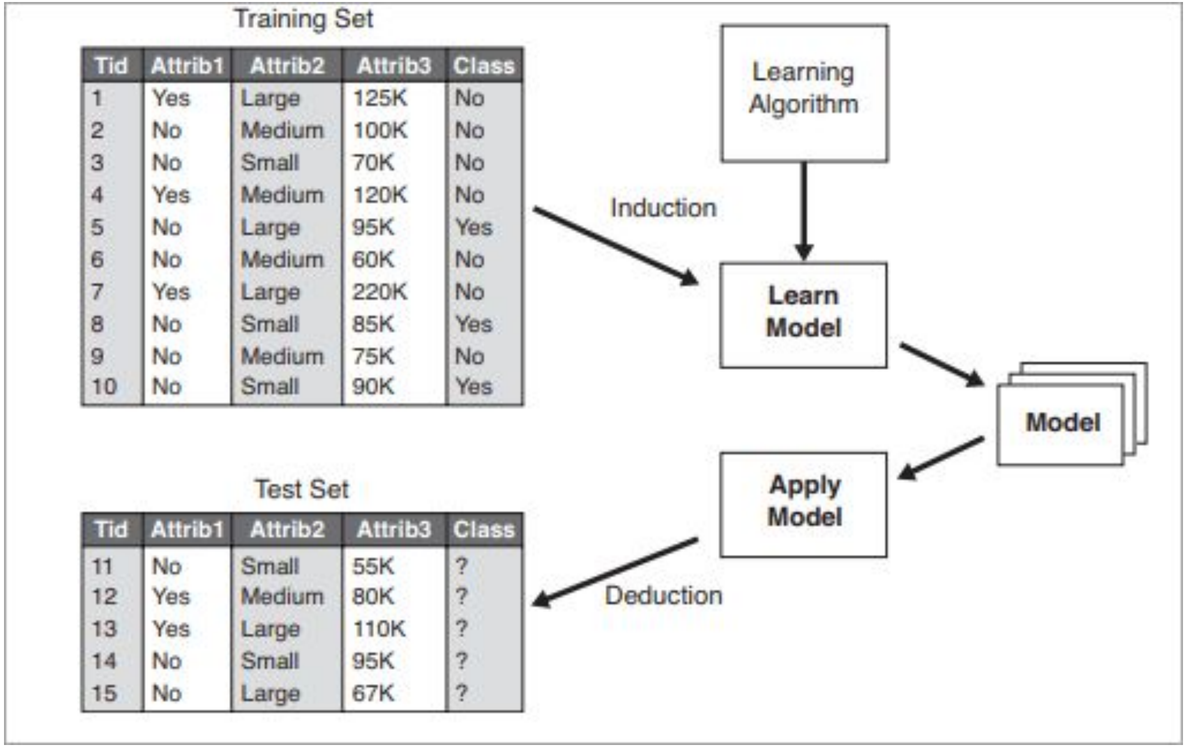
WHAT IS THE USE OF DECISION TREE?

- Decision Tree is used to build classification and regression models. It is used to create data models that will predict class labels or values for the decision-making process. The models are built from the training dataset fed to the system (supervised learning).
- Using a decision tree, we can visualize the decisions that make it easy to understand and thus it is a popular data mining technique.

CLASSIFICATION ANALYSIS

- Data Classification is a form of analysis which builds a model that describes important class variables. For example, a model built to categorize bank loan applications as safe or risky. Classification methods are used in machine learning, and pattern recognition.
- Application of classification includes fraud detection, medical diagnosis, target marketing, etc. The output of the classification problem is taken as “Mode” of all observed values of the terminal node.
- **A two-step process is followed, to build a classification model.**
  - In the first step i.e. learning: A classification model based on training data is built.
  - In the second step i.e. Classification, the accuracy of the model is checked and then the model is used to classify new data. The class labels presented here are in the form of discrete values such as “yes” or “no”, “safe” or “risky”.

GENERAL APPROACH TO BUILD CLASSIFICATION TREE



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	?

REGRESSION ANALYSIS

- Regression analysis is used for the prediction of numeric attributes.
- Numeric attributes are also called continuous values. A model built to predict the continuous values instead of class labels is called the regression model. The output of regression analysis is the “Mean” of all observed values of the node.

HOW DOES A DECISION TREE WORK?

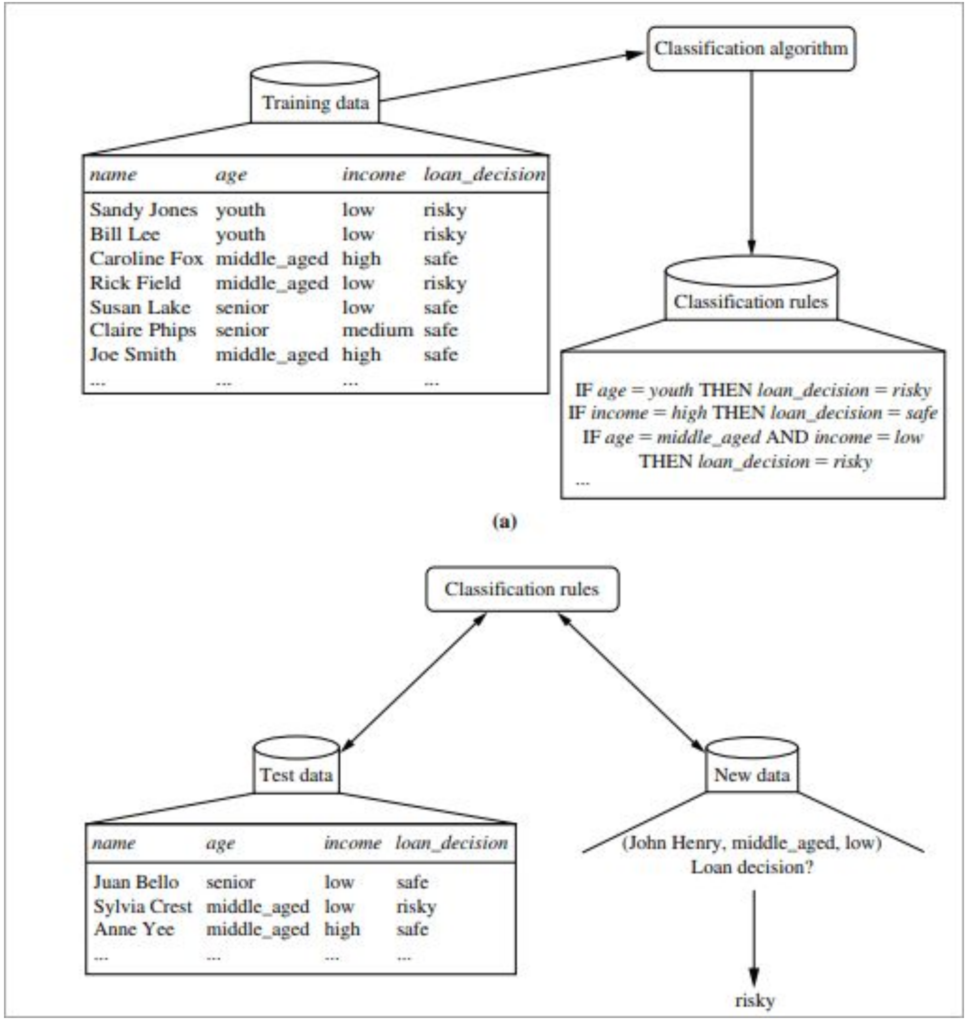
- A decision tree is a supervised learning algorithm that works for both discrete and continuous variables. It splits the dataset into subsets on the basis of the most significant attribute in the dataset. How the decision tree identifies this attribute and how this splitting is done is decided by the algorithms.
- The most significant predictor is designated as the root node, splitting is done to form sub-nodes called decision nodes, and the nodes which do not split further are terminal or leaf nodes.
- In the decision tree, the dataset is divided into homogeneous and non-overlapping regions. It follows a top-down approach as the top region presents all the observations at a single place which splits into two or more branches that further split. This approach is also called a *greedy approach* as it only considers the current node between the worked on without focusing on the future nodes.
- The decision tree algorithms will continue running until a stop criteria such as the minimum number of observations etc. is reached.

- Once a decision tree is built, many nodes may represent outliers or noisy data. Tree pruning method is applied to remove unwanted data. This, in turn, improves the accuracy of the classification model.
- To find the accuracy of the model, a test set consisting of test tuples and class labels is used. The percentages of the test set tuples are correctly classified by the model to identify the accuracy of the model. If the model is found to be accurate then it is used to classify the data tuples for which the class labels are not known.
- Some of the decision tree algorithms include Hunt's Algorithm, ID3, CD4.5, and CART.

EXAMPLE OF CREATING A DECISION TREE

#1) Learning Step: The training data is fed into the system to be analyzed by a classification algorithm. In this example, the class label is the attribute i.e. “loan decision”. The model built from this training data is represented in the form of decision rules.

#2) Classification: Test dataset are fed to the model to check the accuracy of the classification rule. If the model gives acceptable results then it is applied to a new dataset with unknown class variables.



name	*age*	*income*	*loan_decision*
Sandy Jones	youth	low	risky
Bill Lee	youth	low	risky
Caroline Fox	middle_aged	high	safe
Rick Field	middle_aged	low	risky
Susan Lake	senior	low	safe
Claire Phips	senior	medium	safe
Joe Smith	middle_aged	high	safe
...	...	...	...

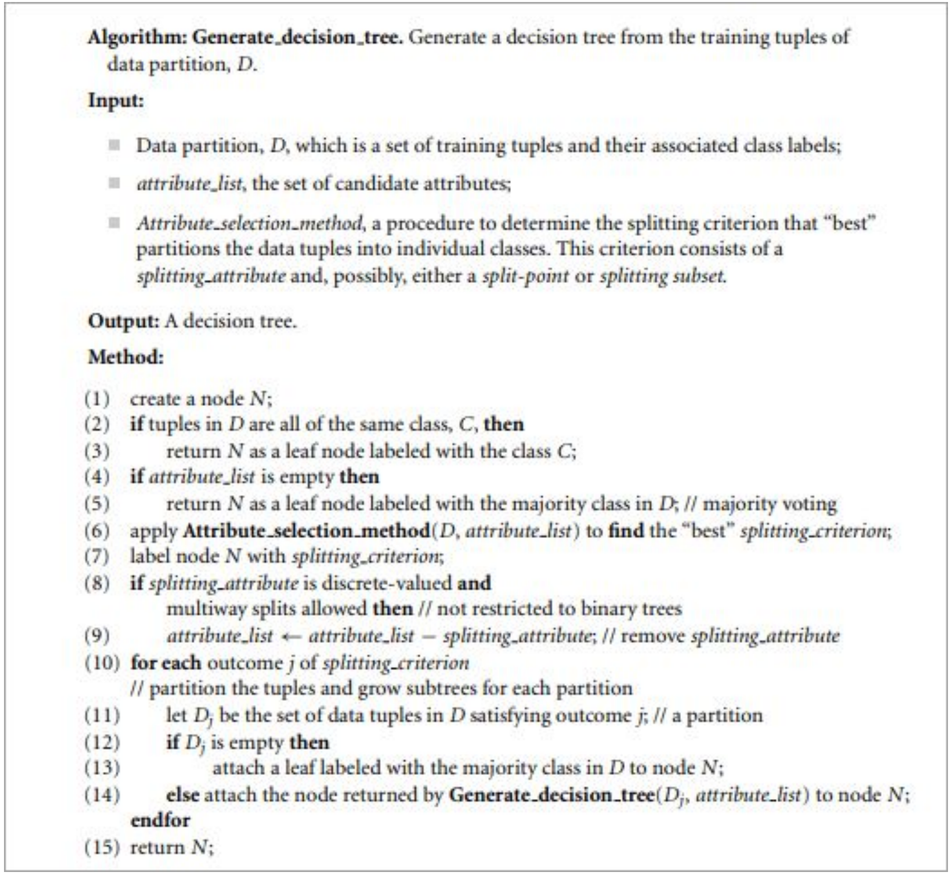
IF *age = youth* THEN *loan_decision = risky*
IF *income = high* THEN *loan_decision = safe*
IF *age = middle_aged* AND *income = low*
THEN *loan_decision = risky*
...

(a)

name	*age*	*income*	*loan_decision*
Juan Bello	senior	low	safe
Sylvia Crest	middle_aged	low	risky
Anne Yee	middle_aged	high	safe
...	...	...	...

(John Henry, middle_aged, low)
Loan decision?

risky



Algorithm: Generate_decision_tree. Generate a decision tree from the training tuples of data partition, *D*.

Input:

- Data partition, *D*, which is a set of training tuples and their associated class labels;
- *attribute_list*, the set of candidate attributes;
- *Attribute_selection_method*, a procedure to determine the splitting criterion that "best" partitions the data tuples into individual classes. This criterion consists of a *splitting_attribute* and, possibly, either a *split-point* or *splitting subset*.

Output: A decision tree.

Method:

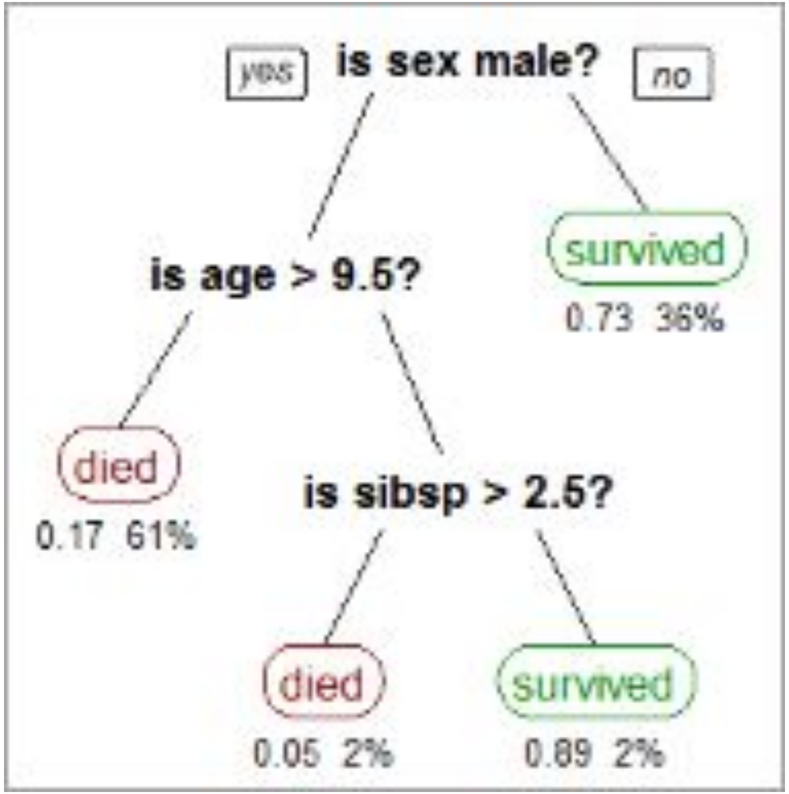
```
create a node $N$;
if tuples in $D$ are all of the same class, $C$, then
    return $N$ as a leaf node labeled with the class $C$;
if attribute_list is empty then
    return $N$ as a leaf node labeled with the majority class in $D$; // majority voting
apply Attribute_selection_method($D$, attribute_list) to find the "best" splitting_criterion;
label node $N$ with splitting_criterion;
if splitting_attribute is discrete-valued and
    multiway splits allowed then // not restricted to binary trees
    attribute_list ← attribute_list − splitting_attribute; // remove splitting_attribute
for each outcome $j$ of splitting_criterion
// partition the tuples and grow subtrees for each partition
    let $D_j$ be the set of data tuples in $D$ satisfying outcome $j$; // a partition
    if $D_j$ is empty then
        attach a leaf labeled with the majority class in $D$ to node $N$;
    else attach the node returned by Generate_decision_tree($D_j$, attribute_list) to node $N$;
endfor
return $N$;
```

DECISION TREE INDUCTION ALGORITHM

AMIRAJ
COLLEGE OF ENGINEERING & TECHNOLOGY

DECISION TREE INDUCTION

- Decision tree induction is the method of learning the decision trees from the training set. The training set consists of attributes and class labels. Applications of decision tree induction include astronomy, financial analysis, medical diagnosis, manufacturing, and production.
- A decision tree is a flowchart tree-like structure that is made from training set tuples. The dataset is broken down into smaller subsets and is present in the form of nodes of a tree. The tree structure has a root node, internal nodes or decision nodes, leaf node, and branches.
- The root node is the topmost node. It represents the best attribute selected for classification. Internal nodes of the decision nodes represent a test of an attribute of the dataset leaf node or terminal node which represents the classification or decision label. The branches show the outcome of the test performed.
- Some decision trees only have *binary nodes*, that means exactly two branches of a node, while some decision trees are non-binary.



The image above shows the decision tree for the Titanic dataset to predict whether the passenger will survive or not.

CART

- CART model i.e. Classification and Regression Models is a decision tree algorithm for building models. Decision Tree model where the target values have a discrete nature is called classification models.
- A discrete value is a finite or countably infinite set of values, **For Example,** age, size, etc. The models where the target values are represented by continuous values are usually numbers that are called Regression Models. Continuous variables are floating-point variables. These two models together are called CART.
- CART uses Gini Index as Classification matrix.

DECISION TREE INDUCTION FOR ID3

- In the late 1970s and early 1980s, J.Ross Quinlan was a researcher who built a decision tree algorithm for machine learning. This algorithm is known as **ID3, Iterative Dichotomiser**. This algorithm was an extension of the concept learning systems described by E.B Hunt, J, and Marin.
- ID3 later came to be known as C4.5. ID3 and C4.5 follow a greedy top-down approach for constructing decision trees. The algorithm starts with a training dataset with class labels that are portioned into smaller subsets as the tree is being constructed.

DECISION TREE INDUCTION FOR ID3

#1) Initially, there are three parameters i.e. **attribute list, attribute selection method and data partition**. The attribute list describes the attributes of the training set tuples.

#2) The attribute selection method describes the method for selecting the best attribute for discrimination among tuples. The methods used for attribute selection can either be Information Gain or Gini Index.

#3) The structure of the tree (binary or non-binary) is decided by the attribute selection method.

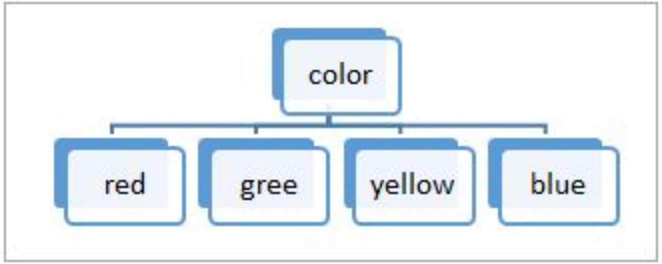
#4) When constructing a decision tree, it starts as a single node representing the tuples.

DECISION TREE INDUCTION FOR ID3

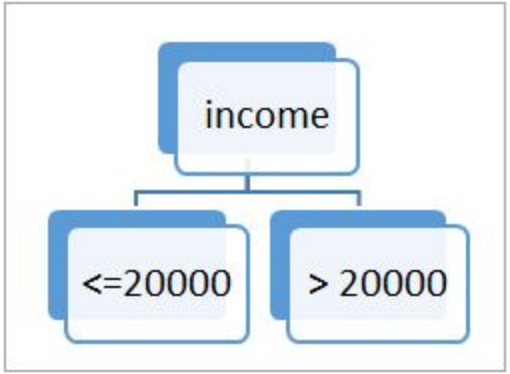
#5) If the root node tuples represent different class labels, then it calls an attribute selection method to split or partition the tuples. The step will lead to the formation of branches and decision nodes.

#6) The splitting method will determine which attribute should be selected to partition the data tuples. It also determines the branches to be grown from the node according to the test outcome. The main motive of the splitting criteria is that the partition at each branch of the decision tree should represent the same class label.

An example of splitting attribute is shown below:



a. The portioning above is discrete-valued.



b. The portioning above is for continuous-valued.

DECISION TREE INDUCTION FOR ID3

#7) The above partitioning steps are followed recursively to form a decision tree for the training dataset tuples.

#8) The portioning stops only when either all the partitions are made or when the remaining tuples cannot be partitioned further.

#9) The complexity of the algorithm is described by **n * |D| * log |D|** where n is the number of attributes in training dataset D and |D| is the number of tuples.

INFORMATION GAIN

- This method is the main method that is used to build decision trees. It reduces the information that is required to classify the tuples. It reduces the number of tests that are needed to classify the given tuple. The attribute with the highest information gain is selected.
- The original information needed for classification of a tuple in dataset D is given by:

$$E(S) = \sum_{i=1}^{c} -p_i \log_2 p_i$$

Where p is the probability that the tuple belongs to class C. The information is encoded in bits, therefore, log to the base 2 is used. E(s) represents the average amount of information required to find out the class label of dataset D. This information gain is also called **Entropy**.

The information required for exact classification after portioning is given by the formula:

$$E(T,X)=\sum_{c\in X}P(c)E(c)$$

Where P (c) is the weight of partition. This information represents the information needed to classify the dataset D on portioning by X.

Information gain is the difference between the original and expected information that is required to classify the tuples of dataset D.

$$Gain(T,X) = Entropy(T) - Entropy(T,X)$$

Gain is the reduction of information that is required by knowing the value of X. The attribute with the highest information gain is chosen as “best”.

GAIN RATIO

Information gain might sometimes result in portioning useless for classification. However, the Gain ratio splits the training data set into partitions and considers the number of tuples of the outcome with respect to the total tuples. The attribute with the max gain ratio is used as a splitting attribute.

$$\text{Gain Ratio}(A) = \frac{\text{Gain}(A)}{\text{SplitInfo}(D)}$$

GINI INDEX

Gini Index is calculated for binary variables only. It measures the impurity in training tuples of dataset D, as

$$\text{Gini} = 1 - \sum_{i} p(i|t)^2$$

P is the probability that tuple belongs to class C. The Gini index that is calculated for binary split dataset D by attribute A is given by:

$$\boxed{GINI_{split} = \sum_{i=1}^{k} \frac{n_i}{n} GINI(i)}$$

WHAT IS TREE PRUNING?

- Pruning is the method of removing the unused branches from the decision tree. Some branches of the decision tree might represent outliers or noisy data.
- Tree pruning is the method to reduce the unwanted branches of the tree. This will reduce the complexity of the tree and help in effective predictive analysis. It reduces the overfitting as it removes the unimportant branches from the trees.

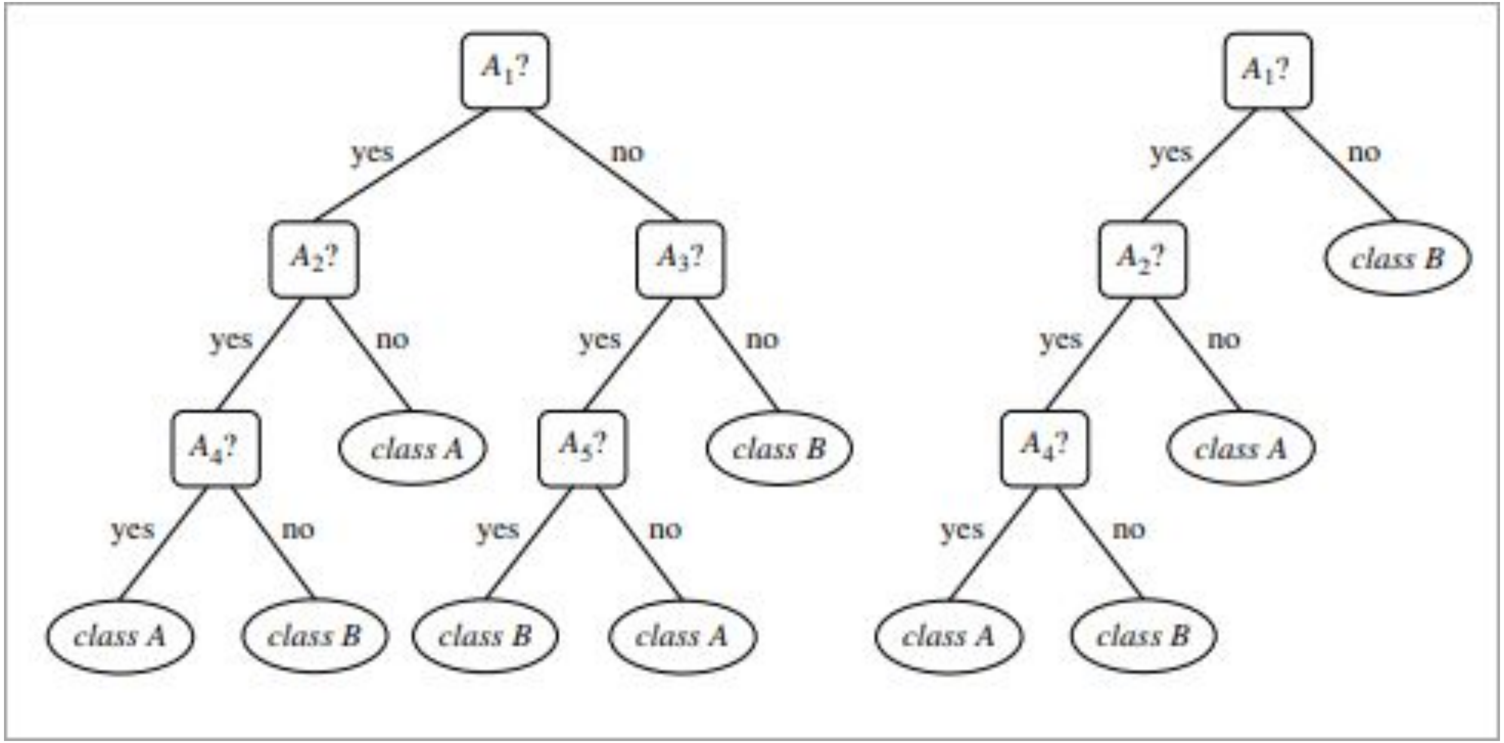
#1) Pre Pruning: In this approach, the construction of the decision tree is stopped early. It means it is decided not to further partition the branches. The last node constructed becomes the leaf node and this leaf node may hold the most frequent class among the tuples.

The attribute selection measures are used to find out the weightage of the split. Threshold values are prescribed to decide which splits are regarded as useful. If the portioning of the node results in splitting by falling below threshold then the process is halted.

#2) Post Pruning: This method removes the outlier branches from a fully grown tree. The unwanted branches are removed and replaced by a leaf node denoting the most frequent class label. This technique requires more computation than prepruning, however, it is more reliable.

The pruned trees are more precise and compact when compared to unpruned trees but they carry a disadvantage of replication and repetition.

Repetition occurs when the same attribute is tested again and again along a branch of a tree. *Replication* occurs when the duplicate subtrees are present within the tree. These issues can be solved by multivariate splits.

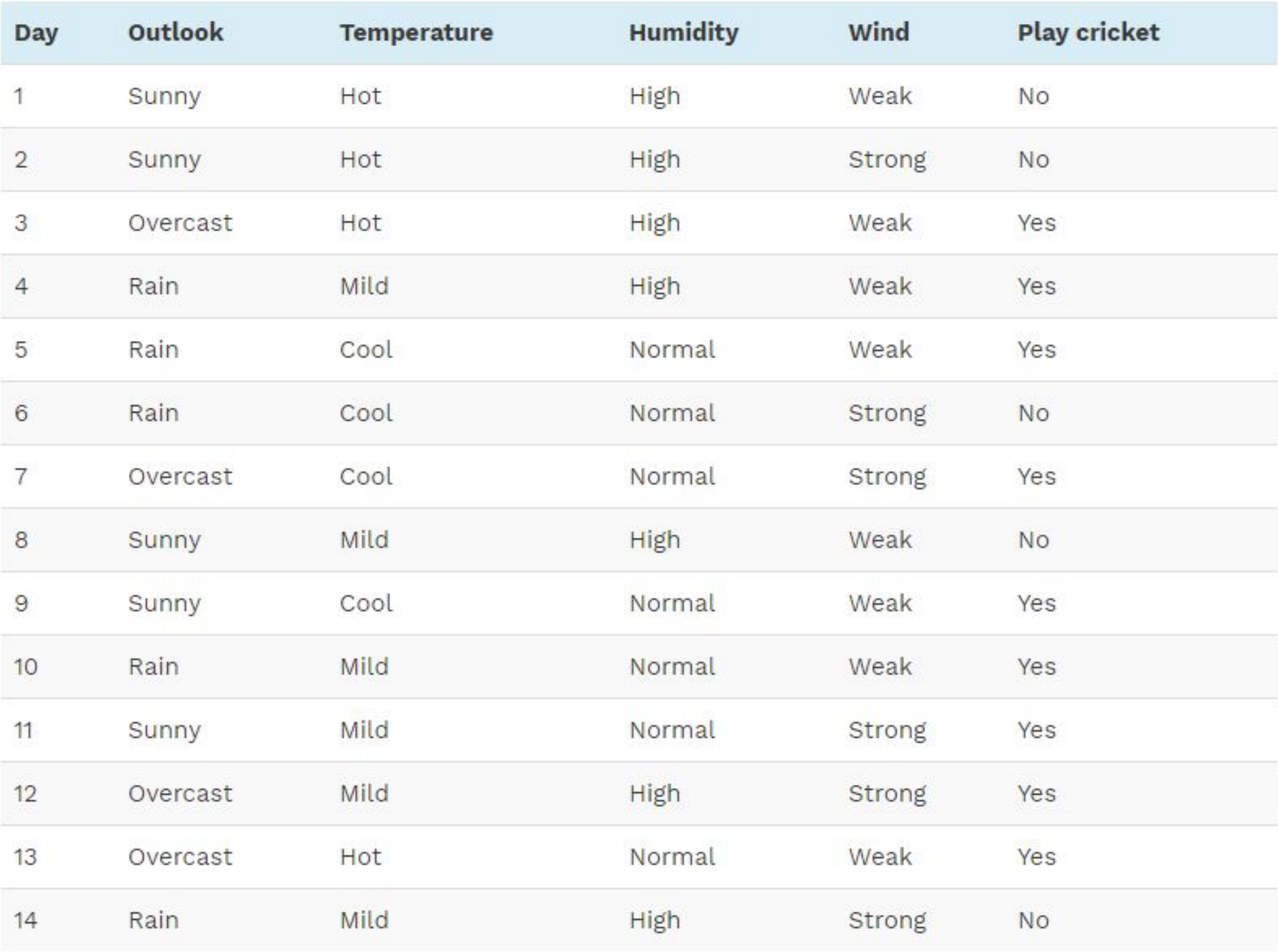


The above image shows an unpruned and pruned tree.

EXAMPLE OF DECISION TREE

Constructing a Decision Tree

Let us take an example of the last 10 days weather dataset with attributes outlook, temperature, wind, and humidity. The outcome variable will be playing cricket or not. We will use the ID3 algorithm to build the decision tree.



Day	Outlook	Temperature	Humidity	Wind	Play cricket
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Step1: The first step will be to create a root node.

Step2: If all results are yes, then the leaf node "yes" will be returned else the leaf node "no" will be returned.

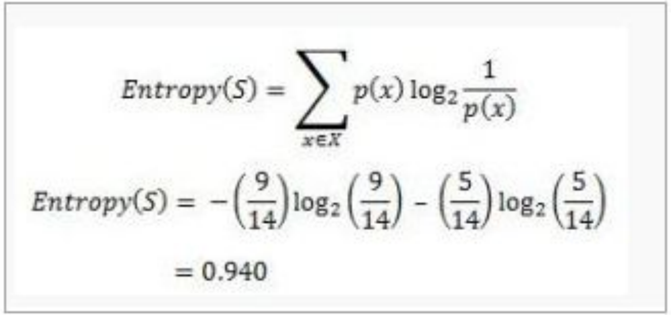
Step3: Find out the Entropy of all observations and entropy with attribute "x" that is E(S) and E(S, x).

Step4: Find out the information gain and select the attribute with high information gain.

Step5: Repeat the above steps until all attributes are covered.

Calculation of Entropy:

Yes	No
9	5



$$Entropy(S) = \sum_{x \in X} p(x) \log_2 \frac{1}{p(x)}$$

$$Entropy(S) = -\left(\frac{9}{14}\right)\log_2\left(\frac{9}{14}\right) - \left(\frac{5}{14}\right)\log_2\left(\frac{5}{14}\right)$$

$$= 0.940$$

If entropy is zero, it means that all members belong to the same class and if entropy is one then it means that half of the tuples belong to one class and one of them belong to other class. 0.94 means fair distribution.

Find the information gain attribute which gives maximum information gain.

For Example “Wind”, it takes two values: Strong and Weak, therefore, x = {Strong, Weak}.

$$IG(S, Wind) = H(S) - \sum_{i=0}^{n} P(x) * H(x)$$

Find out H(x), P(x) for x =weak and x= strong. H(S) is already calculated above.

Weak= 8

Strong= 8

$$P(S_{weak}) = \frac{Number\ of\ Weak}{Total}$$

$$= \frac{8}{14}$$

$$P(S_{strong}) = \frac{Number\ of\ Strong}{Total}$$

$$= \frac{6}{14}$$

For “weak” wind, 6 of them say “Yes” to play cricket and 2 of them say “No”. So entropy will be:

$$Entropy(S_{weak}) = -\left(\frac{6}{8}\right)\log_2\left(\frac{6}{8}\right) - \left(\frac{2}{8}\right)\log_2\left(\frac{2}{8}\right)$$
$$= 0.811$$

For “strong” wind, 3 said “No” to play cricket and 3 said “Yes”.

$$Entropy(S_{strong}) = -\left(\frac{3}{6}\right)\log_2\left(\frac{3}{6}\right) - \left(\frac{3}{6}\right)\log_2\left(\frac{3}{6}\right)$$
$$= 1.000$$

This shows perfect randomness as half items belong to one class and the remaining half belong to others.

Calculate the information gain,

$$IG(S,Wind) = H(S) - \sum_{i=0}^{n} P(x) * H(x)$$

$$IG(S,Wind) = H(S) - P(S_{weak}) * H(S_{weak}) - P(S_{strong}) * H(S_{strong})$$

$$= 0.940 - \left(\frac{8}{14}\right)(0.811) - \left(\frac{6}{14}\right)(1.00)$$

$$= 0.048$$

Similarly the information gain for other attributes is:

$$IG(S,Outlook) = 0.246$$

$$IG(S,Temperature) = 0.029$$

$$IG(S,Humidity) = 0.151$$

The attribute outlook has the **highest information gain** of 0.246, thus it is chosen as root.

Overcast has 3 values: Sunny, Overcast and Rain. Overcast with play cricket is always "Yes". So it ends up with a leaf node, "yes". For the other values "Sunny" and "Rain".

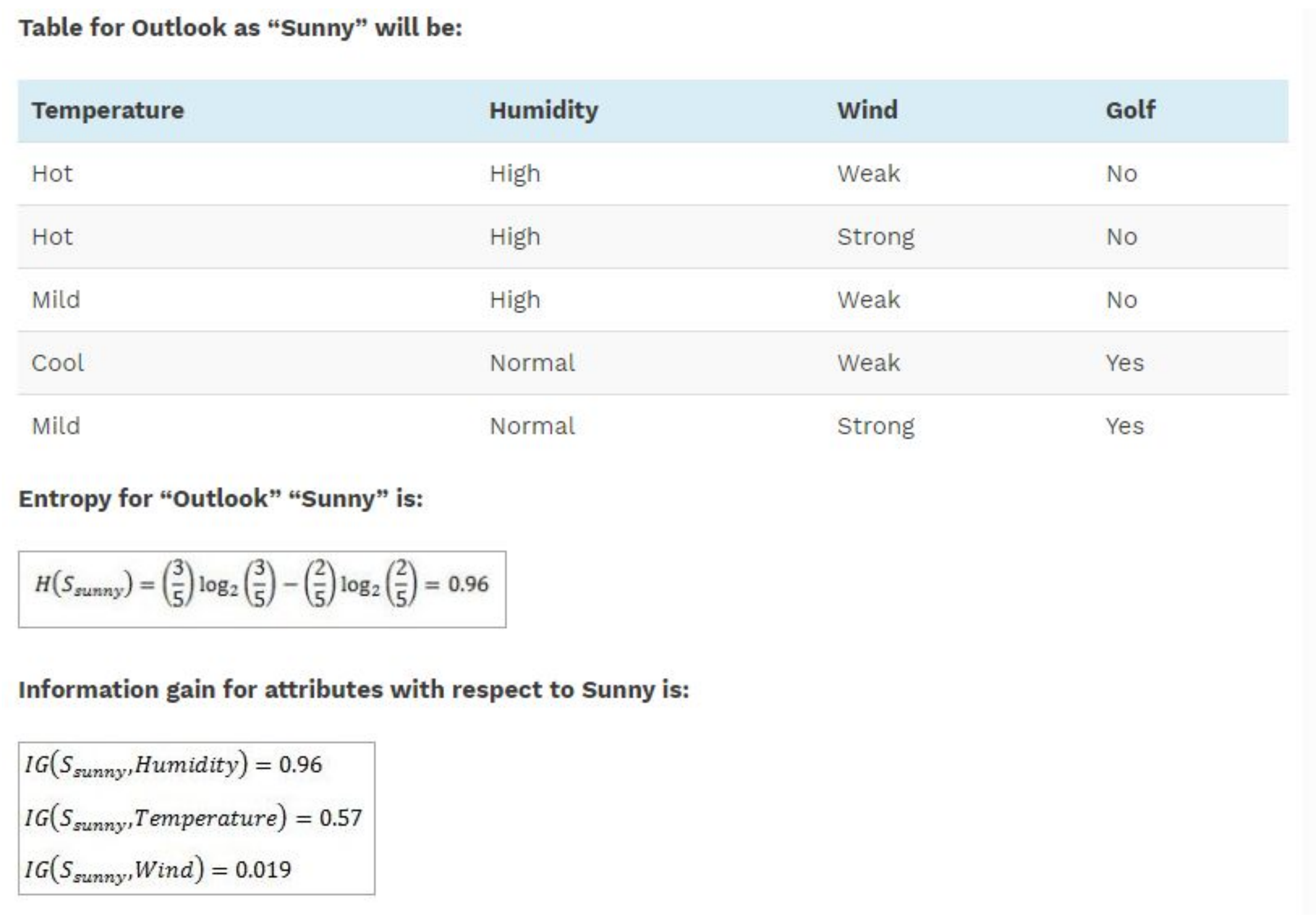


Table for Outlook as "Sunny" will be:

Temperature	Humidity	Wind	Golf
Hot	High	Weak	No
Hot	High	Strong	No
Mild	High	Weak	No
Cool	Normal	Weak	Yes
Mild	Normal	Strong	Yes

Entropy for "Outlook" "Sunny" is:

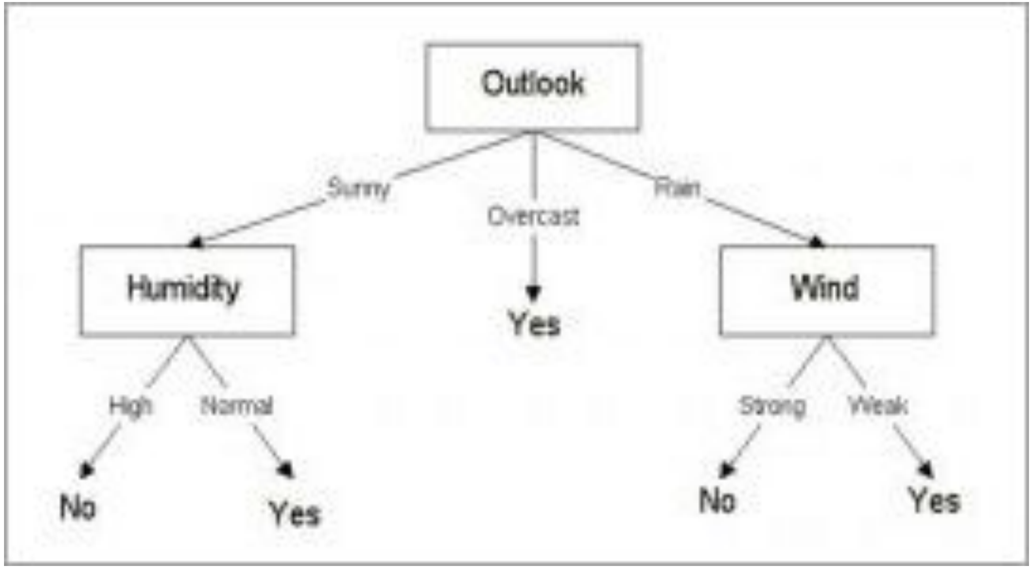
$$H(S_{sunny}) = \left(\frac{3}{5}\right)\log_2\left(\frac{3}{5}\right) - \left(\frac{2}{5}\right)\log_2\left(\frac{2}{5}\right) = 0.96$$

Information gain for attributes with respect to Sunny is:

$$IG(S_{sunny}, Humidity) = 0.96$$

$$IG(S_{sunny}, Temperature) = 0.57$$

$$IG(S_{sunny}, Wind) = 0.019$$


Outlook
Sunny
Overcast
Rain
Humidity
Yes
Wind
High
Normal
Strong
Weak
No
Yes
No
Yes

Advantages Of Decision Tree Classification

Enlisted below are the various merits of Decision Tree Classification:

1. Decision tree classification does not require any domain knowledge, hence, it is appropriate for the knowledge discovery process.
2. The representation of data in the form of the tree is easily understood by humans and it is intuitive.
3. It can handle multidimensional data.
4. It is a quick process with great accuracy.

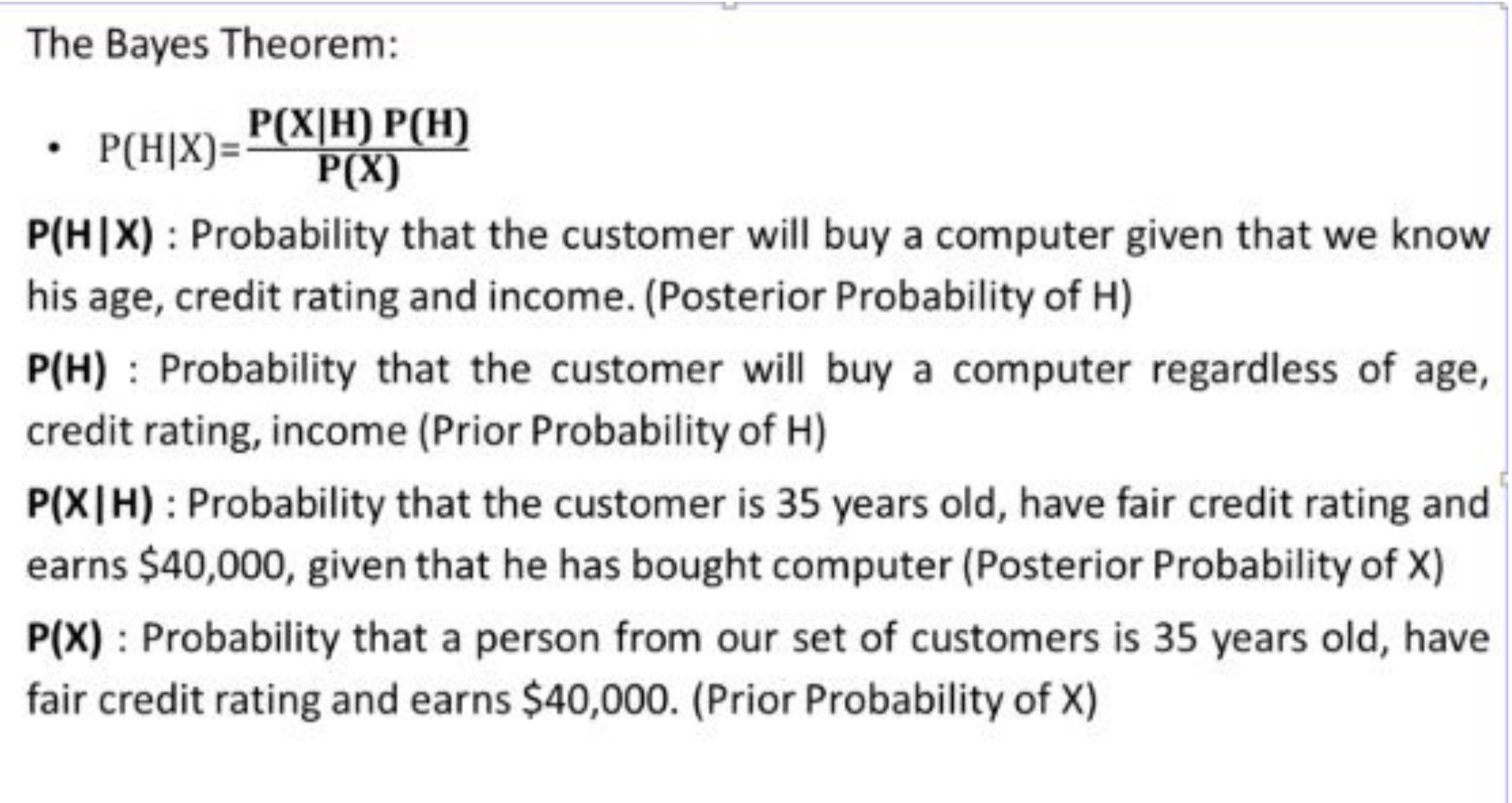
Disadvantages Of Decision Tree Classification

Given below are the various demerits of Decision Tree Classification:

1. Sometimes decision trees become very complex and these are called overfitted trees.
2. The decision tree algorithm may not be an optimal solution.
3. The decision trees may return a biased solution if some class label dominates it.

BAYESIAN CLASSIFICATION

- Thomas Bayes, who proposed the Bayes Theorem so, it named Bayesian theorem.
- It is statistical method & supervised learning method for classification.
- **It can solve problems involving both categorical and continuous valued attributes**.
- Bayesian classification is used to find conditional probabilities.



The Bayes Theorem:

- $P(H|X)=\dfrac{\mathbf{P(X|H)\ P(H)}}{\mathbf{P(X)}}$

P(H|X) : Probability that the customer will buy a computer given that we know his age, credit rating and income. (Posterior Probability of H)

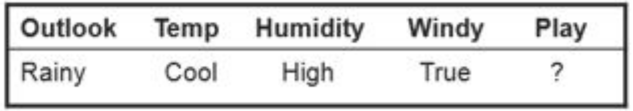
P(H) : Probability that the customer will buy a computer regardless of age, credit rating, income (Prior Probability of H)

P(X|H) : Probability that the customer is 35 years old, have fair credit rating and earns $40,000, given that he has bought computer (Posterior Probability of X)

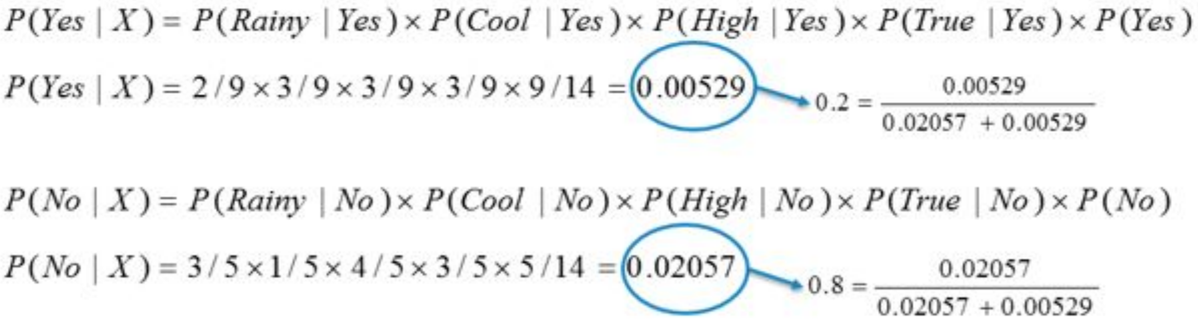
P(X) : Probability that a person from our set of customers is 35 years old, have fair credit rating and earns $40,000. (Prior Probability of X)

NAIVE BAYES CLASSIFIER EXAMPLE

In this example we have 4 inputs (predictors). The final posterior probabilities can be standardized between 0 and 1.



Outlook	Temp	Humidity	Windy	Play
Rainy	Cool	High	True	?

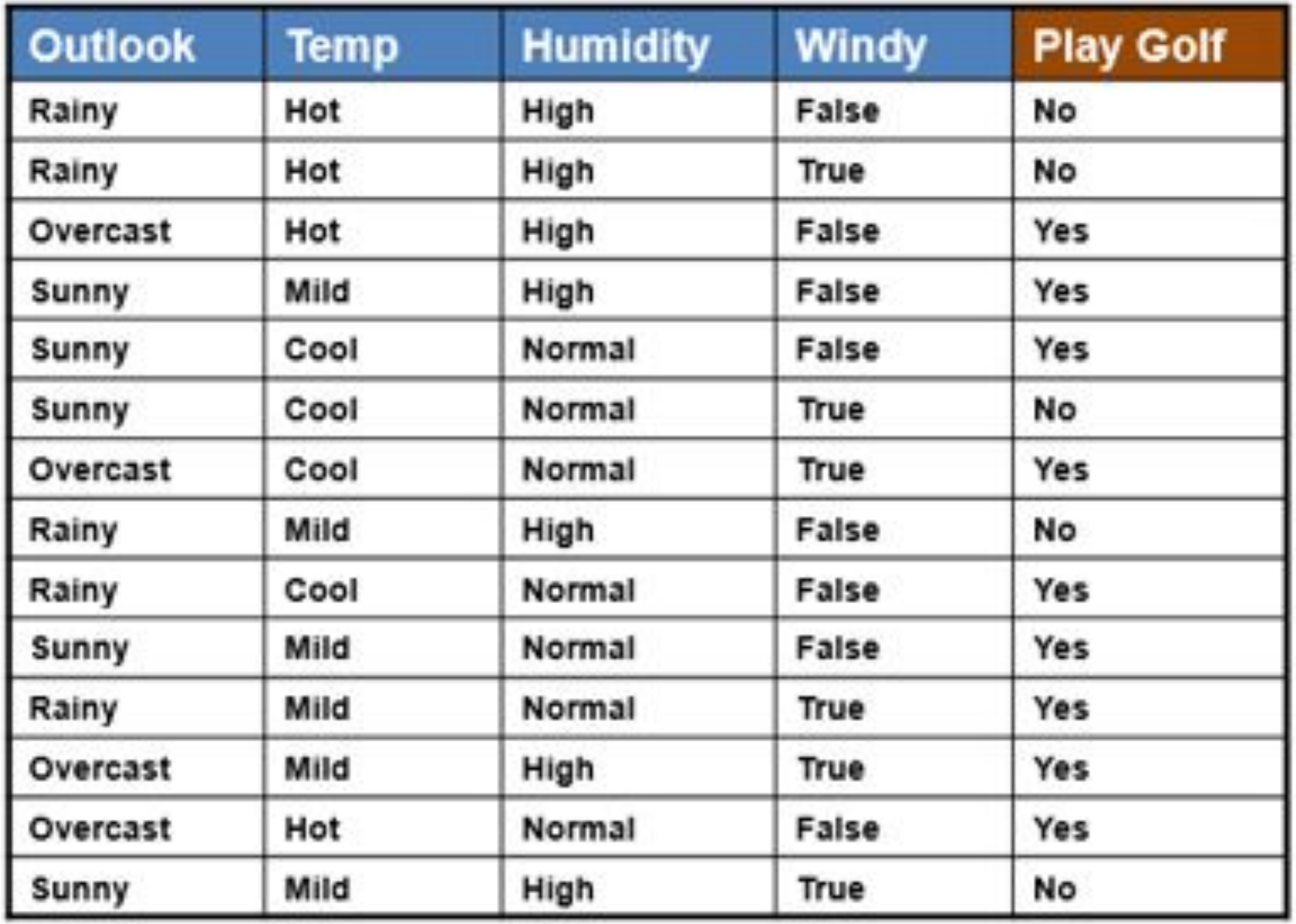


$$P(Yes \mid X) = P(Rainy \mid Yes) \times P(Cool \mid Yes) \times P(High \mid Yes) \times P(True \mid Yes) \times P(Yes)$$

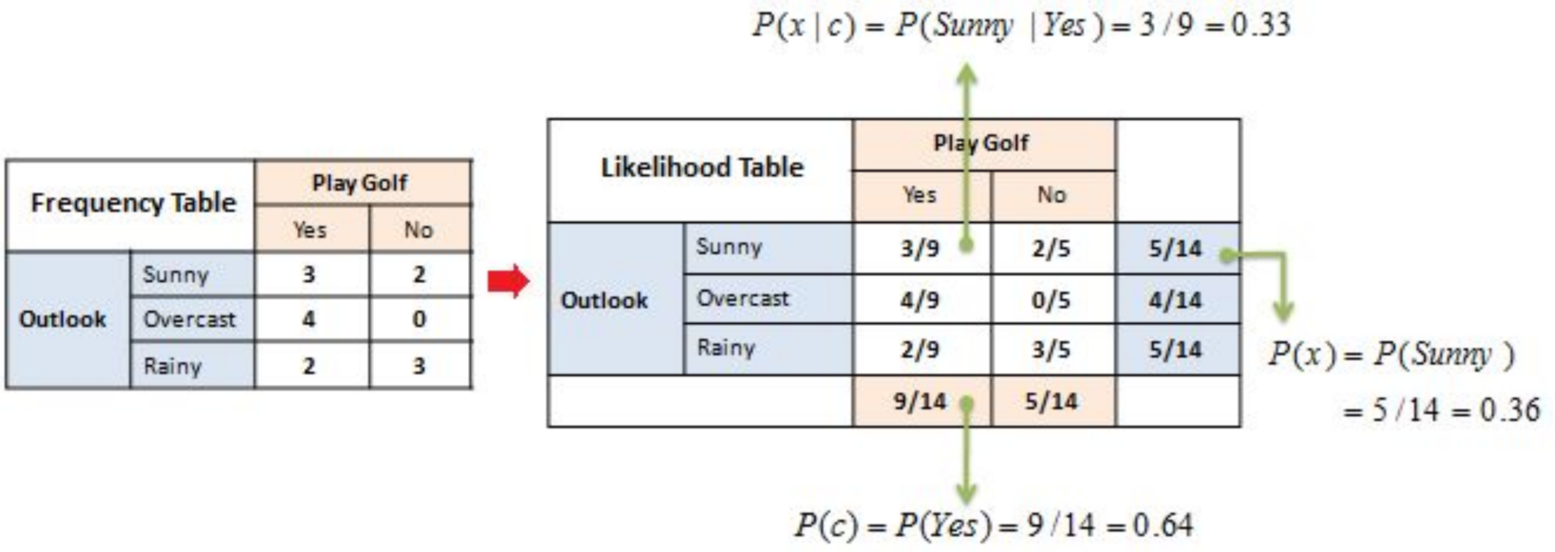
$$P(Yes \mid X) = 2/9 \times 3/9 \times 3/9 \times 3/9 \times 9/14 = \boxed{0.00529} \rightarrow 0.2 = \frac{0.00529}{0.02057 + 0.00529}$$

$$P(No \mid X) = P(Rainy \mid No) \times P(Cool \mid No) \times P(High \mid No) \times P(True \mid No) \times P(No)$$

$$P(No \mid X) = 3/5 \times 1/5 \times 4/5 \times 3/5 \times 5/14 = \boxed{0.02057} \rightarrow 0.8 = \frac{0.02057}{0.02057 + 0.00529}$$



Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No



Frequency Table		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

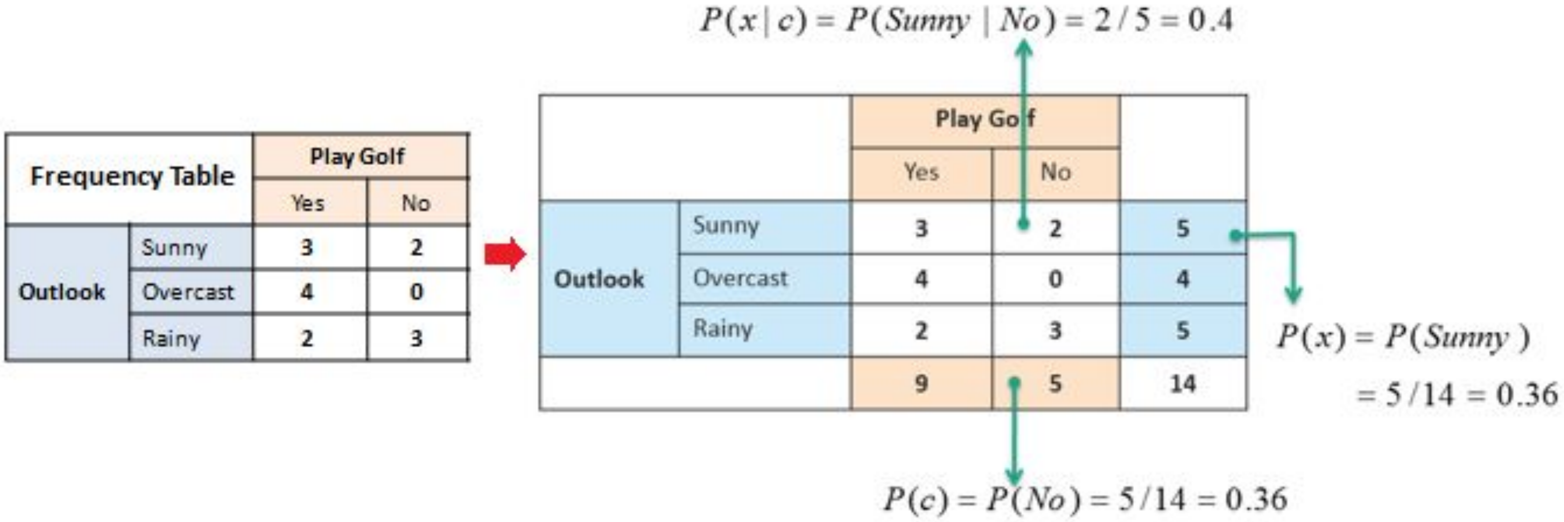
Likelihood Table		Play Golf		
		Yes	No	
Outlook	Sunny	3/9	2/5	5/14
	Overcast	4/9	0/5	4/14
	Rainy	2/9	3/5	5/14
		9/14	5/14	

$$P(x \mid c) = P(Sunny \mid Yes) = 3/9 = 0.33$$

$$P(x) = P(Sunny) = 5/14 = 0.36$$

$$P(c) = P(Yes) = 9/14 = 0.64$$

Posterior Probability: $P(c \mid x) = P(Yes \mid Sunny) = 0.33 \times 0.64 \div 0.36 = 0.60$



Frequency Table		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
		9	5	14

$$P(x \mid c) = P(Sunny \mid No) = 2/5 = 0.4$$

$$P(x) = P(Sunny)$$
$$= 5/14 = 0.36$$

$$P(c) = P(No) = 5/14 = 0.36$$

Posterior Probability: $P(c \mid x) = P(No \mid Sunny) = 0.40 \times 0.36 \div 0.36 = 0.40$

Frequency Table

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3

Likelihood Table

		Play Golf	
		Yes	No
Outlook	Sunny	3/9	2/5
	Overcast	4/9	0/5
	Rainy	2/9	3/5

		Play Golf	
		Yes	No
Humidity	High	3/9	4/5
	Normal	6/9	1/5

		Play Golf	
		Yes	No
Temp.	Hot	2/9	2/5
	Mild	4/9	2/5
	Cool	3/9	1/5

		Play Golf	
		Yes	No
Windy	False	6/9	2/5
	True	3/9	3/5

RULE BASED

It is featured by building rules based on an object attributes.

Rule-based classifier makes use of a set of **IF-THEN rules** for classification.

We can express a rule in the following from

IF condition THEN conclusion

Let us consider a rule R1,

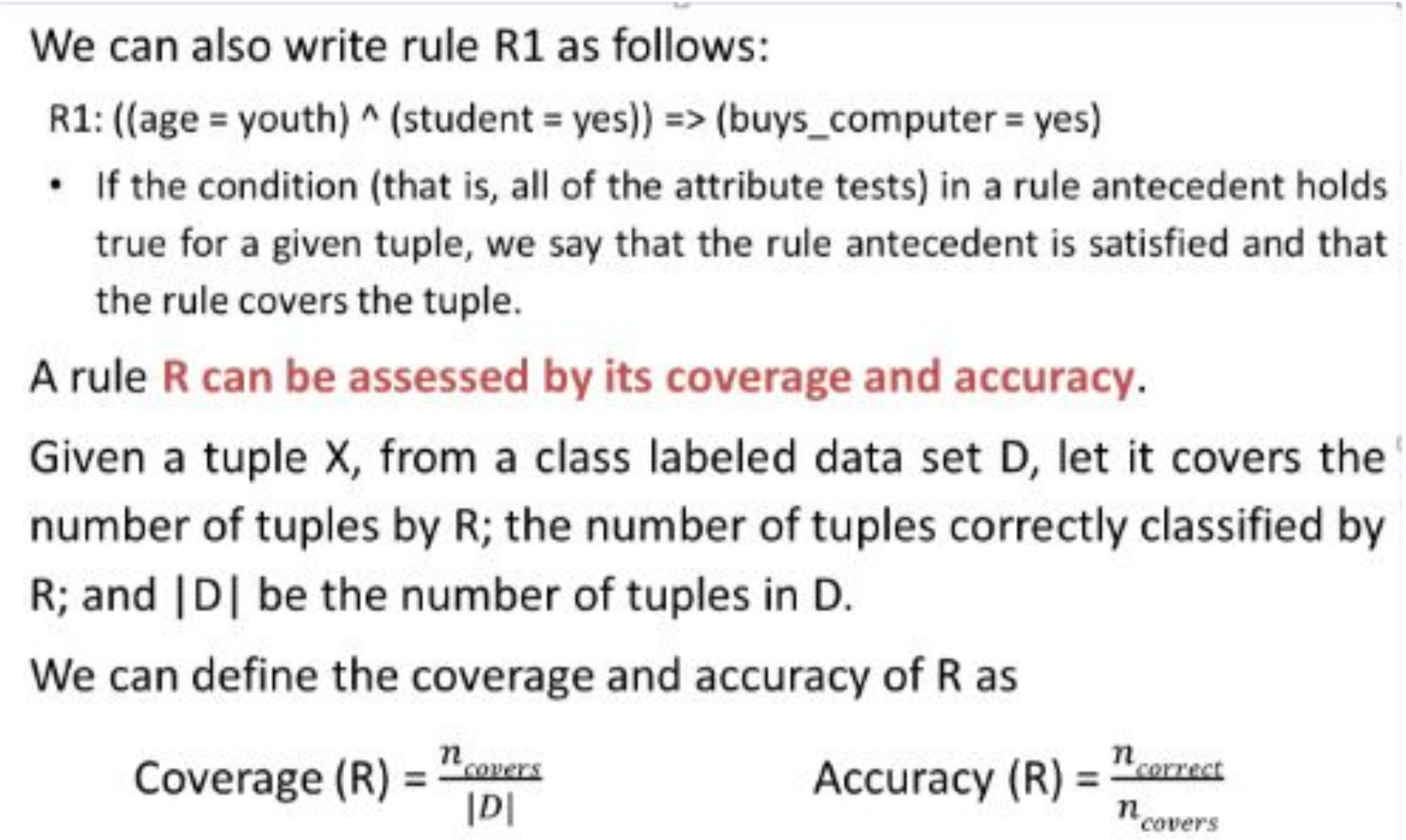
R1: **IF age=youth AND student=yes THEN buy_computer=yes**

The IF part of the rule is called rule antecedent or precondition.

The THEN part of the rule is called rule consequent (conclusion).

The antecedent (IF) part the condition consist of one or more attribute tests and these tests are logically ANDed.

The consequent (THEN) part consists of class prediction.



We can also write rule R1 as follows:

R1: ((age = youth) ^ (student = yes)) => (buys_computer = yes)

- If the condition (that is, all of the attribute tests) in a rule antecedent holds true for a given tuple, we say that the rule antecedent is satisfied and that the rule covers the tuple.

A rule **R can be assessed by its coverage and accuracy**.

Given a tuple X, from a class labeled data set D, let it covers the number of tuples by R; the number of tuples correctly classified by R; and |D| be the number of tuples in D.

We can define the coverage and accuracy of R as

$$\text{Coverage (R)} = \frac{n_{covers}}{|D|} \qquad \text{Accuracy (R)} = \frac{n_{correct}}{n_{covers}}$$

AMIRAJ
COLLEGE OF ENGINEERING & TECHNOLOGY

NEURAL NETWORK

The Artificial Neural Network (ANN) bases its assimilation of data on the way that the human brain processes information. The brain has billions of cells called neurons that process information in the form of electric signals. External information, or stimuli, is received, after which the brain processes it, and then produces an output.

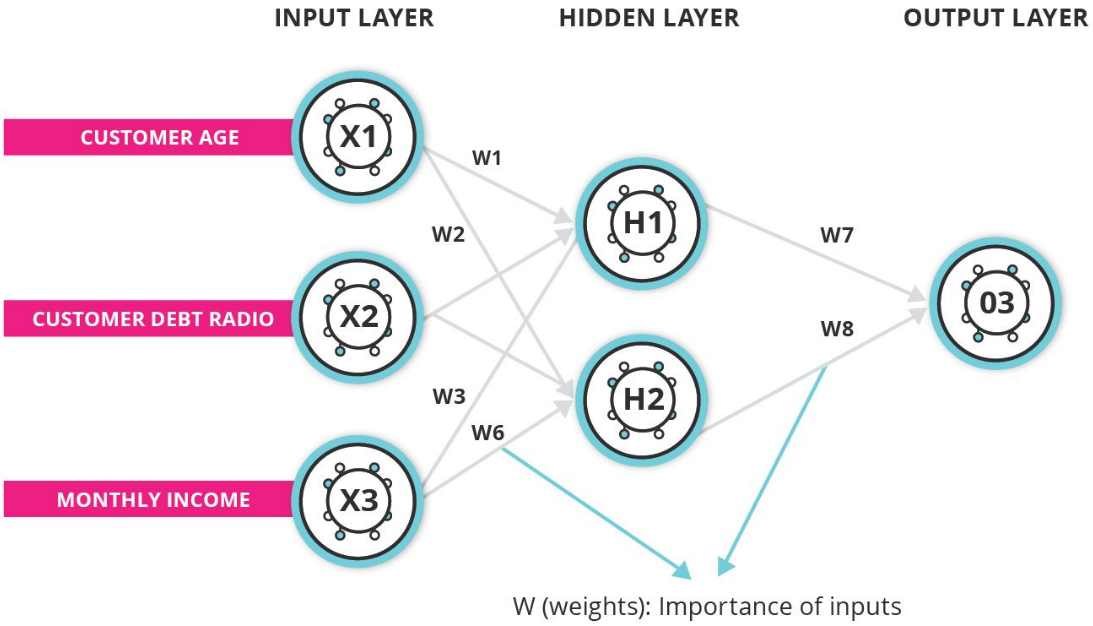
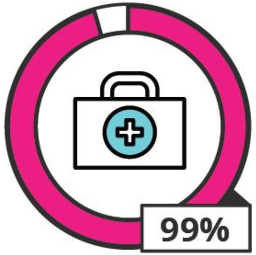


Image source: Mahanta, J. (Jul, 2017). 'Introduction to neural networks, advantages and applications'. Retrieved from Towards Data Science.

- In Professor Arackioraj’s paper, “Applications of neural networks in data mining”, he notes that finding information that is hidden in data is as difficult as it is important.22 Neural networks mine data in areas such as bioinformatics, banking, and retail. Using neural networks, data warehousing organisations can harvest information from datasets to help users make more informed decisions through neural network’s ability to handle complex relationships, cross-pollination of data, and machine learning. Neural networks and AI technologies can carry out many business purposes with unstructured data, from tracking and documenting real-time communications, to finding new customers that automate follow-ups and flag warm leads.23
- Until recently, decision-makers had to rely primarily on extracted data from structured, highly organised data sets, as these are easier to analyse. Unstructured data like emails and copy, are more difficult to analyse, and so have gone unutilised or simply ignored. Neural networks can now provide decision-makers with much deeper insight into the ‘why’ of a customer’s behaviour, which goes beyond what is provided in more structured data.24
- In healthcare, an example of how neural networks are successfully mining data is shown by Imperial College London, where ANNs are used to produce optimal patient care recommendations for patients with sepsis.



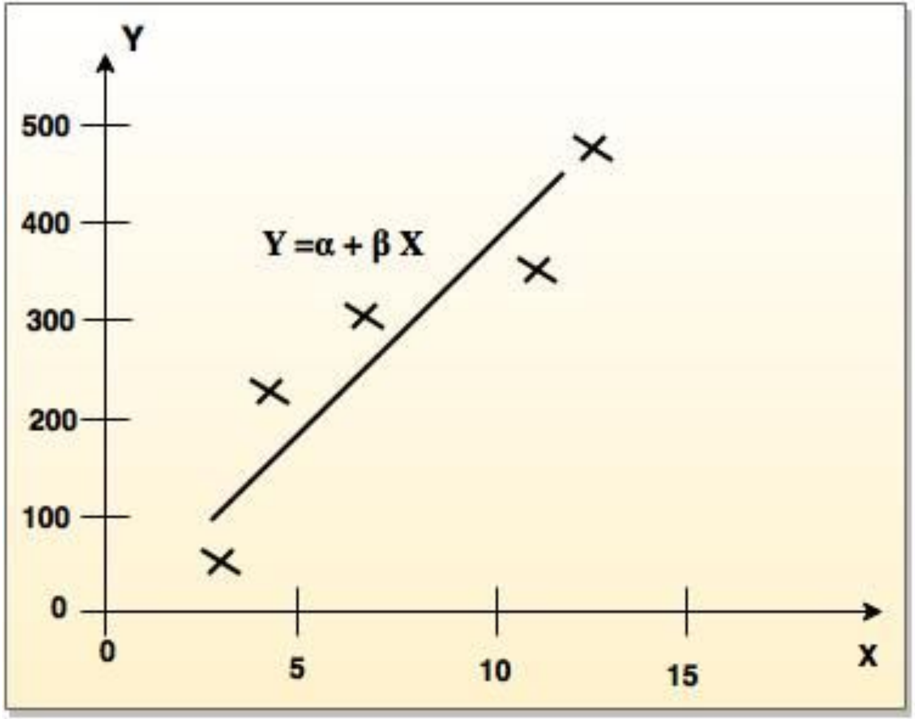
After analysing **100,000 records of intensive care unit patients**, the neural network learned to apply previous experience to diagnose the best course of treatment, with 99% of the recommendations matching or improving a human doctor’s decision.[25]

PREDICTION METHODS

LINEAR AND NONLINEAR REGRESSION

- It is simplest form of regression. Linear regression attempts to model the relationship between two variables by fitting a linear equation to observe the data.
- Linear regression attempts to find the mathematical relationship between variables.
- If outcome is straight line then it is considered as linear model and if it is curved line, then it is a non linear model.
- The relationship between dependent variable is given by straight line and it has only one independent variable.

  $$\mathbf{Y = \alpha + B\,X}$$
- Model **'Y'**, is a linear function of **'X'**.
- The value of 'Y' increases or decreases in linear manner according to which the value of 'X' also changes.



Linear Regression

MULTIPLE LINEAR REGRESSION

- ❖ Multiple linear regression is an extension of linear regression analysis.
- ❖ It uses two or more independent variables to predict an outcome and a single continuous dependent variable.

$\mathbf{Y = a_0 + a_1\ X_1 + a_2\ X_2 + ... + a_k\ X_k + e}$

where,

'Y' is the response variable.

$\mathbf{X_1 + X_2 + X_k}$ are the independent predictors.

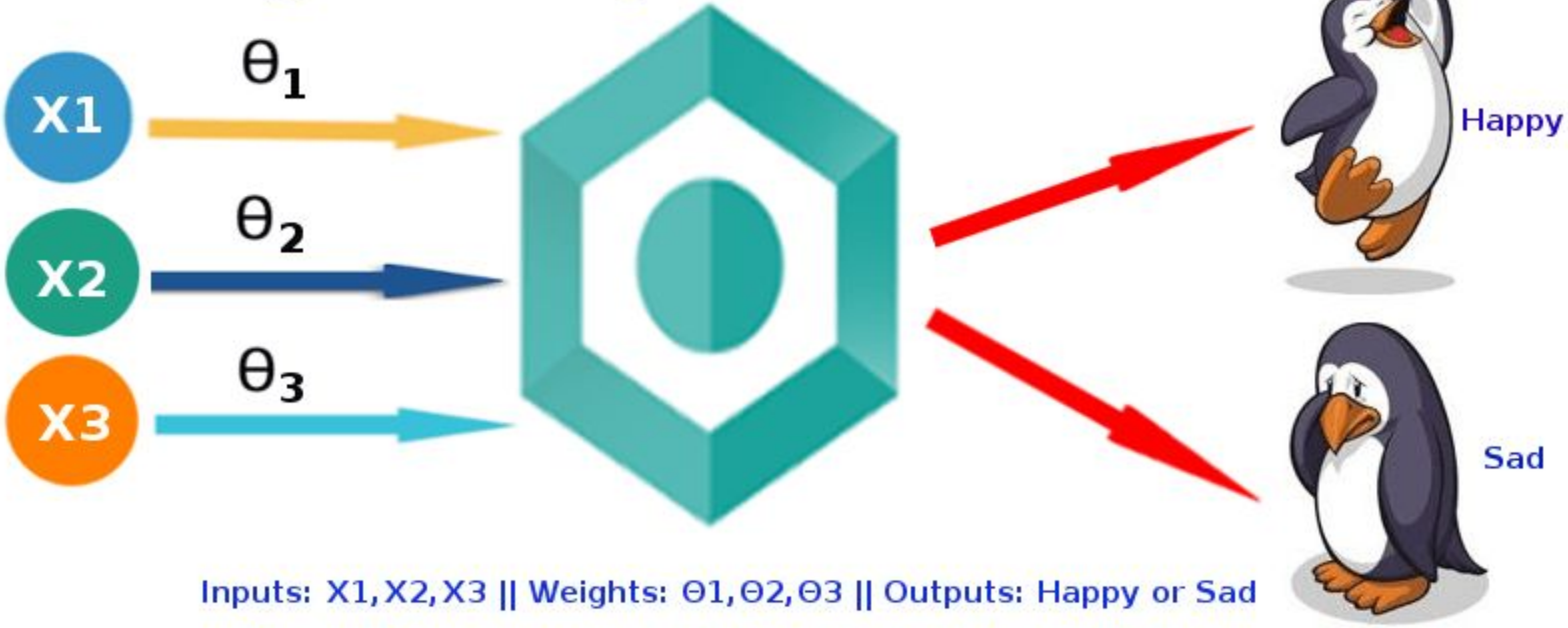
'e' is random error.

$\mathbf{a_0, a_1, a_2, a_k}$ are the regression coefficients.

LOGISTIC REGRESSION

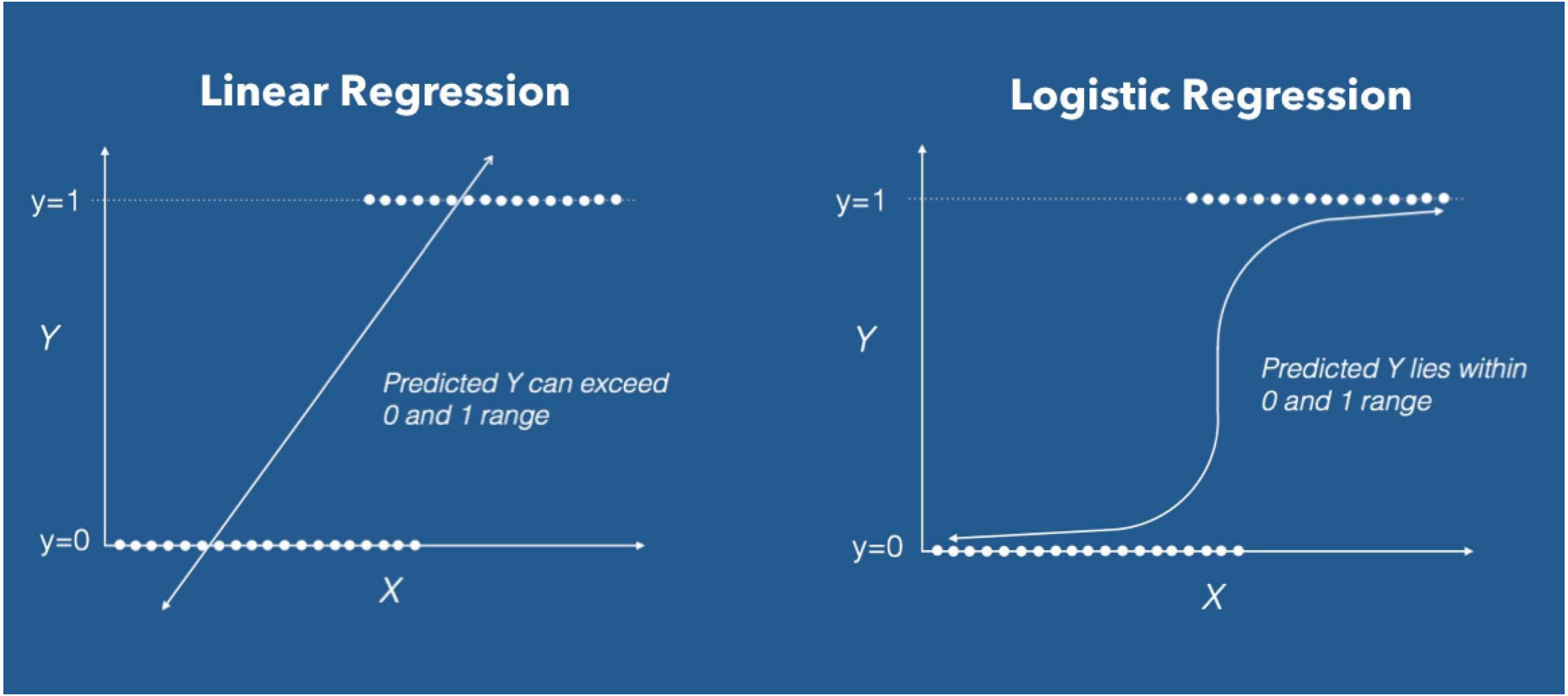
- Logistic Regression was used in the biological sciences in early twentieth century. It was then used in many social science applications. Logistic Regression is used when the dependent variable(target) is categorical.
- For example,
  - To predict whether an email is spam (1) or (0)
  - Whether the tumor is malignant (1) or not (0)

Logistic Regression Model



Inputs: X1,X2,X3 || Weights: Θ1,Θ2,Θ3 || Outputs: Happy or Sad

@dataaspirant.com



Linear Regression

Predicted Y can exceed 0 and 1 range

Logistic Regression

Predicted Y lies within 0 and 1 range

INTRODUCTION OF TOOLS

AMIRAJ
COLLEGE OF ENGINEERING & TECHNOLOGY

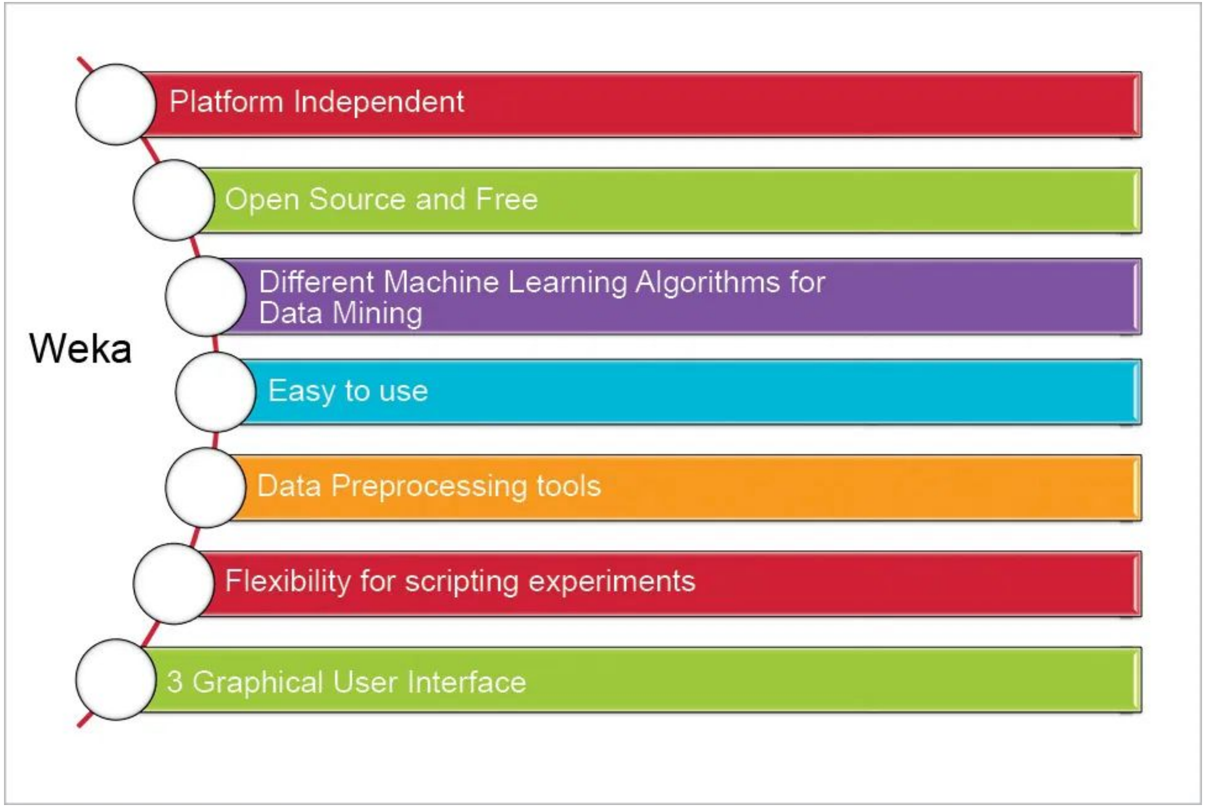
RAPID|MINER
WEKA
The University of Waikato
impact analytix
Microsoft SQL Server
SAP HANA
KNIME
Office Excel
Spotfire
R
tableau
Greenplum
PARACCEL
ORACLE
Predixion Software
sas
SPSS
AN IBM COMPANY
alteryx
TERADATA
NETEZZA
an IBM Company
ASTER

DATA MINING TOOLS

php
oscommerce
php
open cart
drupal
Open Source
Wordpress
zencart
joomla
Magento

OPEN SOURCE DATA MINING TOOLS

WEKA

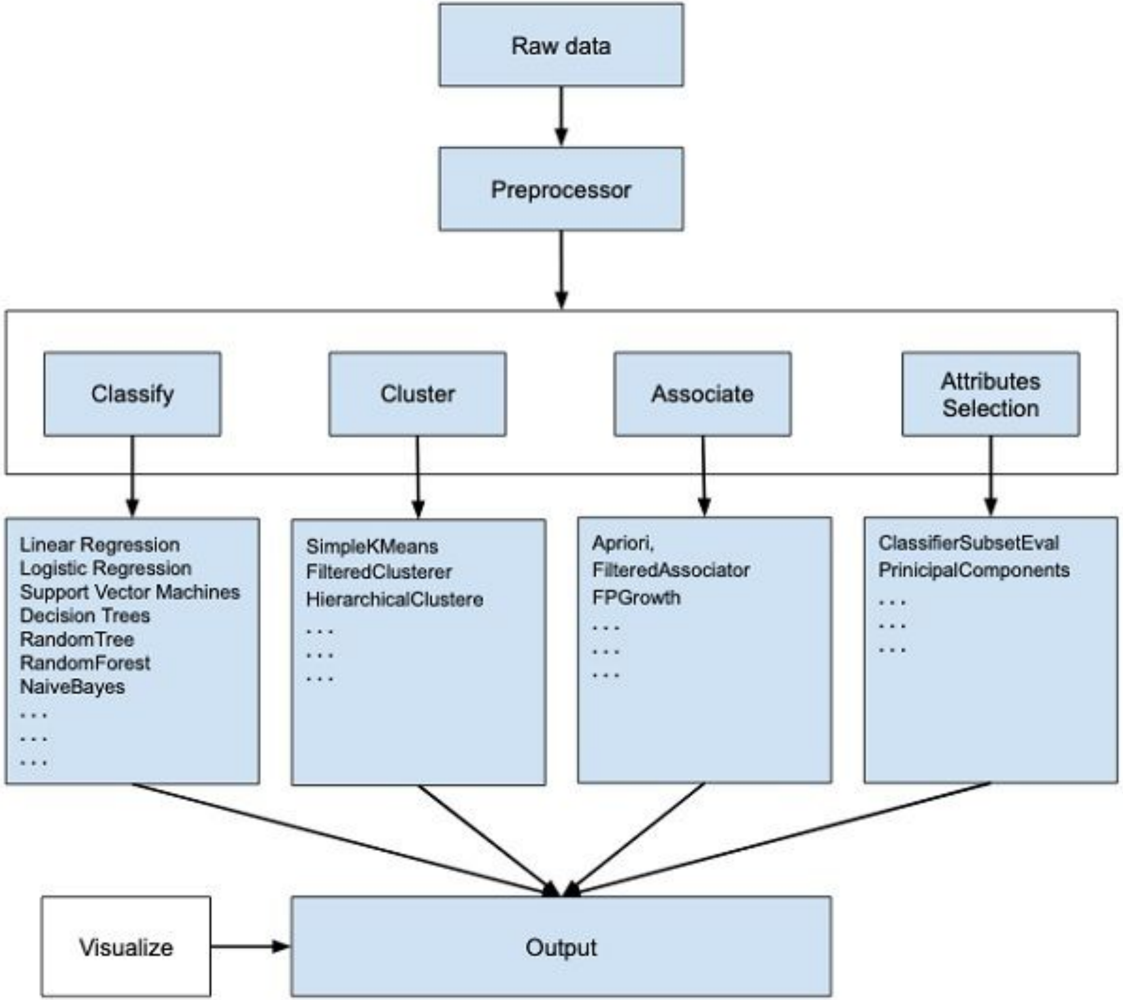


Weka

- Platform Independent
- Open Source and Free
- Different Machine Learning Algorithms for Data Mining
- Easy to use
- Data Preprocessing tools
- Flexibility for scripting experiments
- 3 Graphical User Interface

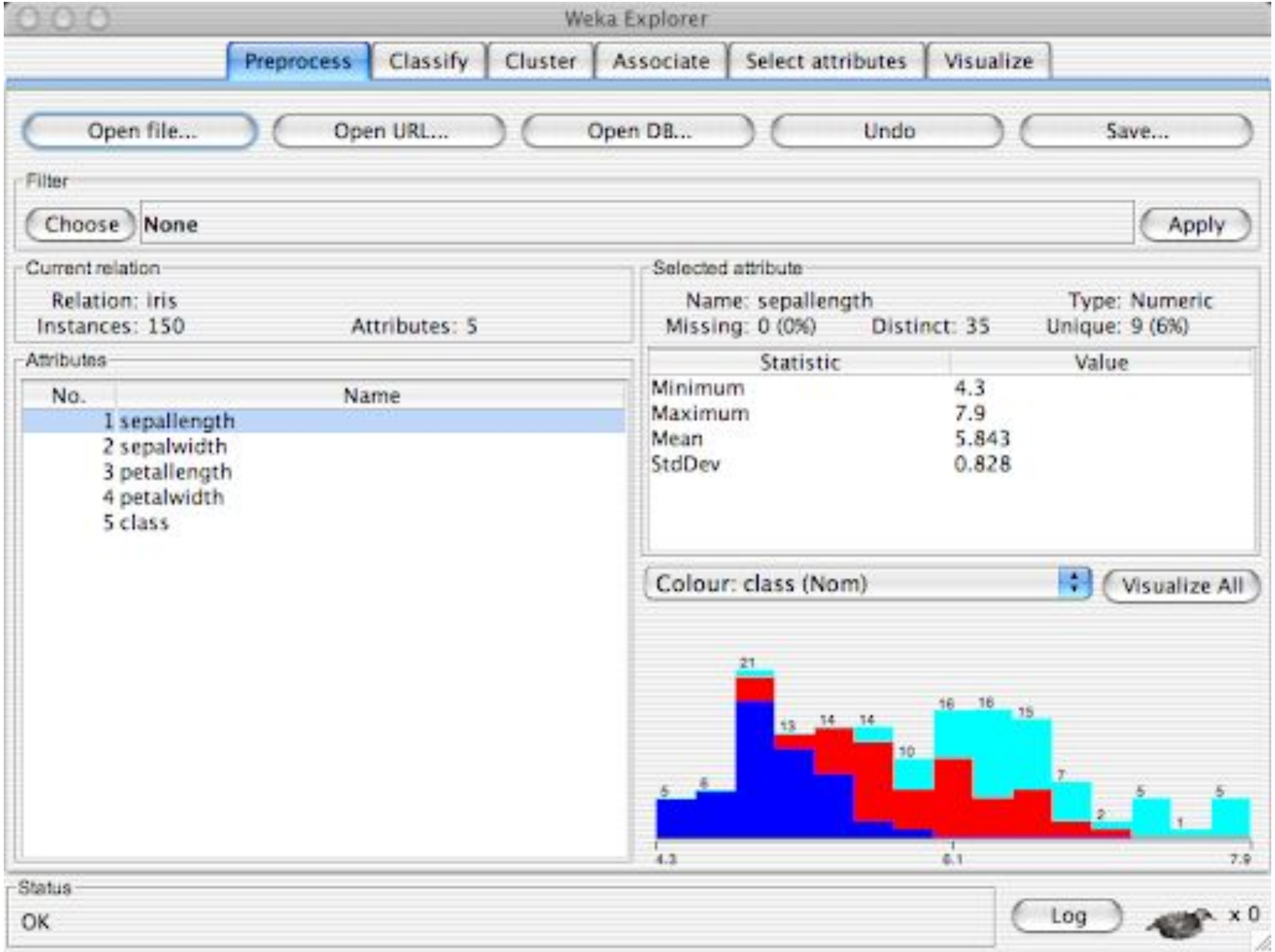
WEKA

- WEKA - an open source software provides tools for data preprocessing, implementation of several Machine Learning algorithms, and visualization tools so that you can develop machine learning techniques and apply them to real-world data mining problems. What WEKA offers is summarized in the following diagram −


Raw data
Preprocessor
Classify
Cluster
Associate
Attributes Selection
Linear Regression
Logistic Regression
Support Vector Machines
Decision Trees
RandomTree
RandomForest
NaiveBayes
. . .
. . .
. . .
SimpleKMeans
FilteredClusterer
HierarchicalClustere
. . .
. . .
. . .
Apriori,
FilteredAssociator
FPGrowth
. . .
. . .
. . .
ClassifierSubsetEval
PrinicipalComponents
. . .
. . .
. . .
Visualize
Output

WEKA

- If you observe the beginning of the flow of the image, you will understand that there are many stages in dealing with Big Data to make it suitable for machine learning −
- First, you will start with the raw data collected from the field. This data may contain several null values and irrelevant fields. You use the data preprocessing tools provided in WEKA to cleanse the data.
- Then, you would save the preprocessed data in your local storage for applying ML algorithms.
- Next, depending on the kind of ML model that you are trying to develop you would select one of the options such as Classify, Cluster, or Associate. The Attributes Selection allows the automatic selection of features to create a reduced dataset.



Filter

None

Current relation

Relation: iris
Instances: 150 Attributes: 5

Attributes

No.	Name
1	sepallength
2	sepalwidth
3	petallength
4	petalwidth
5	class

Selected attribute

Name: sepallength Type: Numeric
Missing: 0 (0%) Distinct: 35 Unique: 9 (6%)

Statistic	Value
Minimum	4.3
Maximum	7.9
Mean	5.843
StdDev	0.828

Colour: class (Nom)

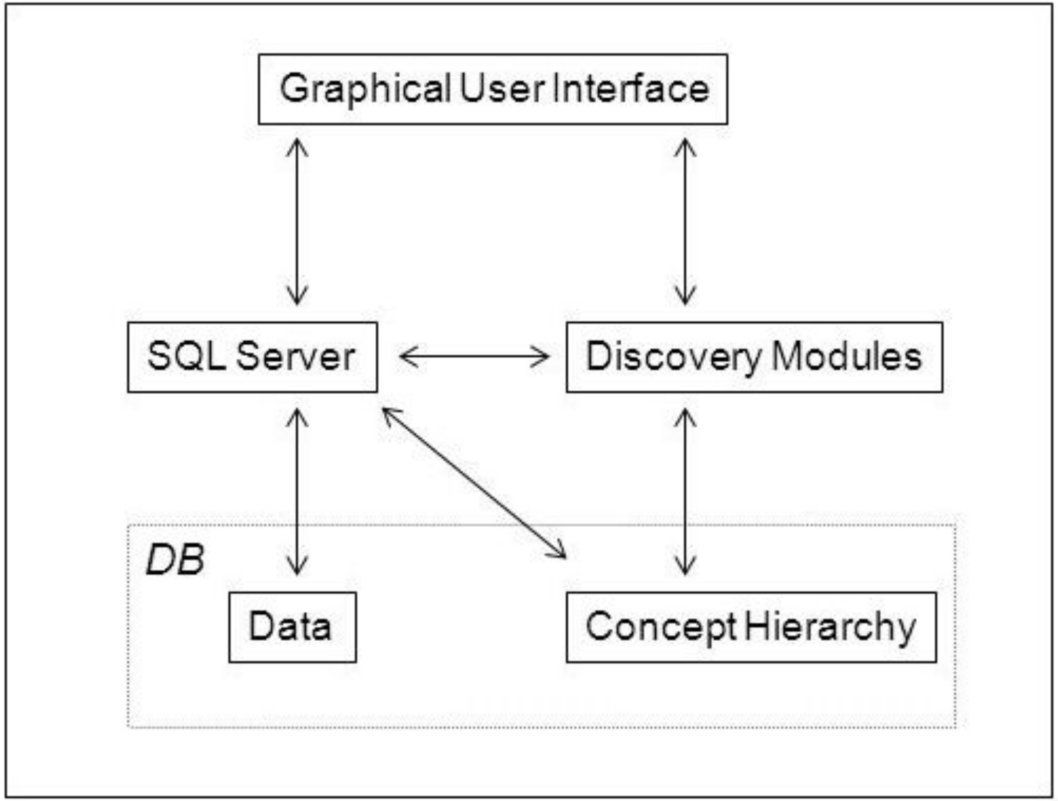
Status

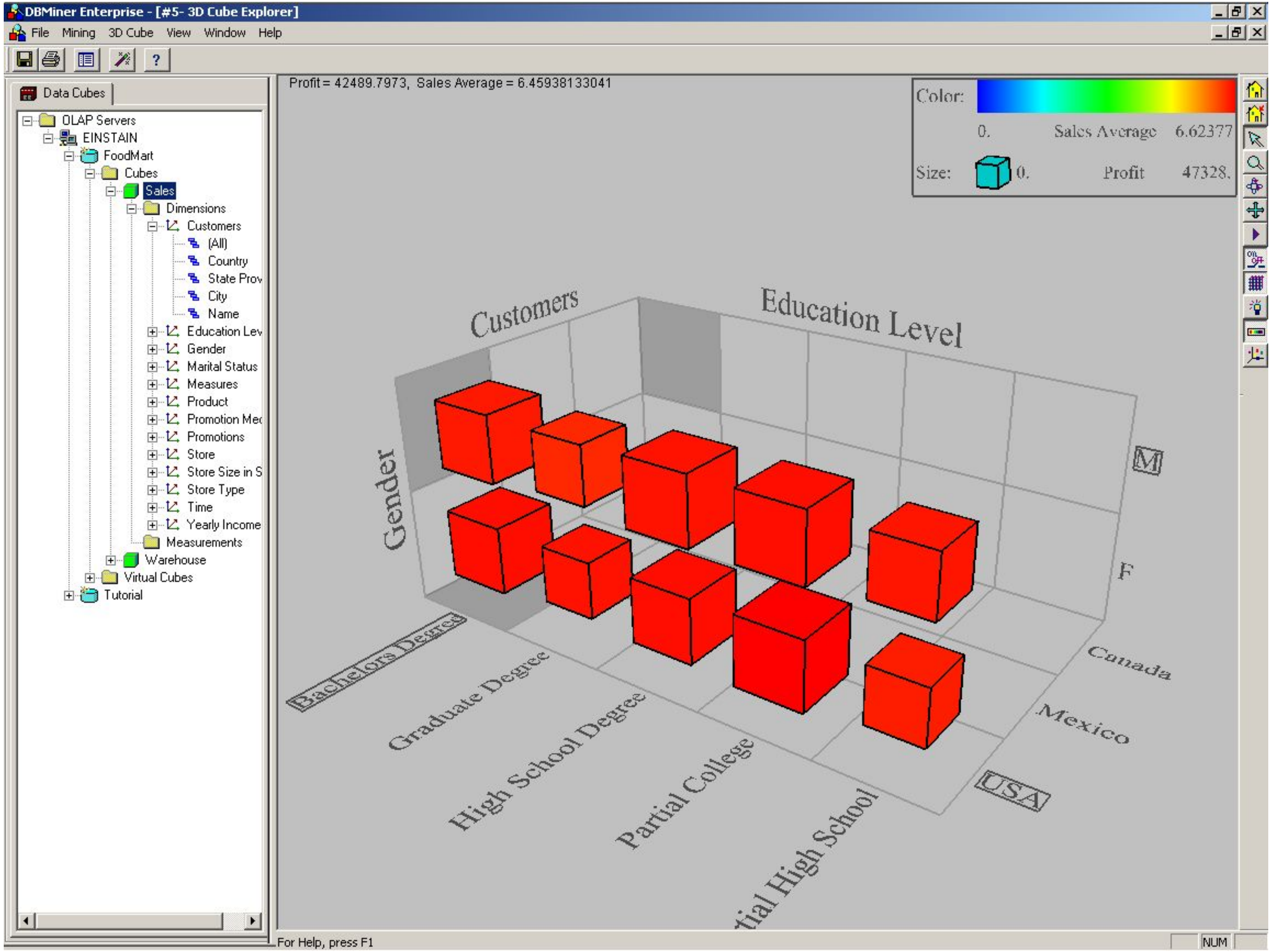
OK

x 0

DBMINER

- A data mining system, DBMiner, has been developed for interactive mining of multiple-level knowledge in large relational databases. The system implements a wide spectrum of data mining functions, including generalization, characterization, association, classification, and prediction.
- A data mining system, DBMiner, has been developed for interactive mining of multiple-level knowledge in large relational databases and data warehouses. The system implements a wide spectrum of data mining functions, including characterization, comparison, association, classification, prediction, and clustering. By incorporating several interesting data mining techniques, including OLAP and attribute-oriented induction, statistical analysis, progressive deepening for mining multiple-level knowledge, and meta-rule guided mining, the system provides a user-friendly, interactive data mining environment with good performance.


Graphical User Interface
SQL Server
Discovery Modules
DB
Data
Concept Hierarchy



Profit = 42489.7973, Sales Average = 6.45938133041

For Help, press F1

THANK
YOU