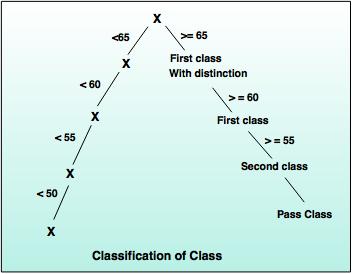
Introduction

* Classification constructs the classification model by using training data set.
* Classification predicts the value of classifying attribute or class label.  
  **For example:** Classification of credit approval on the basis of customer data.  
  **University gives class to the students based on marks.**
* If x >= 65, then First class with distinction.
* If 60<= x<= 65, then First class.
* If 55<= x<=60, then Second class.
* If 50<= x<= 55, then Pass class.



Classification Requirements

**The two important steps of classification are:**  
  
**1. Model construction**

* A predefine class label is assigned to every sample tuple or object. These tuples or subset data are known as training data set.
* The constructed model, which is based on training set is represented as classification rules, decision trees or mathematical formulae.

**2. Model usage**

* The constructed model is used to perform classification of unknown objects.
* A class label of test sample is compared with the resultant class label.
* Accuracy of model is compared by calculating the percentage of test set samples, that are correctly classified by the constructed model.
* Test sample data and training data sample are always different.

Classification vs Prediction

|  |  |
| --- | --- |
| **Classification** | **Prediction** |
| It uses the prediction to predict the class labels. | It is used to assess the values of an attribute of a given sample. |
| **For example:** If the patients are grouped on the basis of their known medical data and treatment outcome, then it is considered as classification. | **For example:** If a classification model is used to predict  the treatment outcome for a new patient, then it is prediction. |

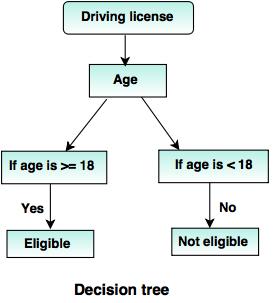
## Issues related to Classification and Prediction

**1. Data preparation**  
Data preparation consist of data cleaning, relevance analysis and  
data transformation.  
  
**2. Evaluation of classification methods**  
**i)  Predictive accuracy:** This is an ability of a model to predict the class label of a new      
     or previously unseen data.  
**ii) Speed and scalability:** It refers to the time required to construct and use the model and increase efficiency in disk- resident databases.  
  
**3. Inter-predictability:**  
It is an understanding and insight provided by the model.

## Decision Tree Induction Method

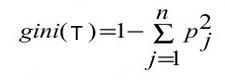
**Decision tree:**

* A decision tree performs the classification in the form of tree structure. It breaks down the dataset into small subsets and a decision tree can be designed simultaneously.
* The final result is a tree with decision node.

**For example:**  
The following decision tree can be designed to declare a result, whether an applicant is eligible or not eligible to get the driving license.  
  


## Attribute selection methods

#### 1. Gini Index (IBM intelligent Miner)

* Gini index is used in **CART** (Classification and Regression Trees), IBM's Intelligent Miner system, **SPRINT** (Scalable Parallelizable Induction of decision Trees).  
  **If a data set 'T' contains examples from 'n' classes, gini index, gini (T) is defined as:  
    
  After splitting T into two subsets T1,  T2  with sizes N1 and N2, the gini index of the split data is defined as:  
  ginisplit (T) = N1/ N2 gini (T1) + N2/ N gini (T2)**
* For each attribute, each of the possible binary splits is considered. For each attribute, the attribute providing smallest ginisplit (T) is chosen to split the node for continuous- valued attributes, where each possible split-point must be considered.

#### 2. ID3 (Algorithm for inducing a decision Tree)

* Ross Quinlin developed  ID3 algorithm in 1980.
* C4.5 is an extension of ID3.
* It avoids over-fitting  of the data.
* It determines the depth of decision tree and reduces the error pruning.
* It also handles continuous value attributes. **For example:** salary or temperature.
* It works for missing value attribute and handles suitable attribute selection measure.
* It gives better efficiency of computation.

**Algorithm to generate a decision tree from the training tuples of data partition, D.**  
**Step 1:** Create a node 'N':  
**Step 2:** If tuple in D are all of the same class, 'C', then go to step 3  
**Step 3:** Return 'N' as a leaf node labeled with the majority class in 'C'  
**Step 4:** If attribute list is empty, then return 'N' as a leaf node labeled with the majority class in D.  
**Step 5:** Apply attribute\_selection\_method (D, attribute \_list) to find the “best” splitting criteria.  
**Step 6:** If splitting\_attribute is discrete-valued and multi way, splits are allowed. Then follow step 7  
**Step 7:** Attribute\_list ← attribute\_list – splitting\_attribute;// remove splitting attribute.  
**Step 8:** For each outcome j of splitting creation, Let Dj be the set of data tuples in D that satisfies outcome j, If Dj is empty, then attached leaf is labeled with the majority class in D to node N;  
**Step 9:** Else, attach the node returned by Generate\_decision\_tree (Dj, attribute\_list ) to node N;  
**Step 10:** Return N;  
**Step 11:** Stop.

#### 3. Tree Pruning

* To avoid the overfitting problem, it is necessary to prune the tree.
* Generally, there are two possibilities while constructing a decision tree. Some record may contain noisy data, which increases the size of the decision tree. Another possibility is, if the number of training examples are too small to produce a representative sample of the true target function.
* Pruning can be possible in a top down or bottom up fashion.

**Some well known methods to perform pruning are:**  
  
**1. Reduced error pruning**  
This is simplest method of pruning. Start from the leaves. Each node is replaced with its most popular class to maintain accuracy.  
  
**2. Cost complexity pruning**

* It generates a series of trees.
* Consider 'T0' as the initial tree and 'Tm' as root.
* Consider that the tree is created by removing a subtree from tree **i- 1** and replacing it with a leaf node with value chosen as per the tree constructing algorithm.
* **The subtree which is removed can be chosen as follows:**
* Define the error rate of tree 'T' over data set 'S' as err (T,S).
* The subtree from tree that minimizes is chosen for removal.
* The function **(T,t)** defines the tree, which is obtained by pruning the subtrees **'t'** from the tree **'T'**. After creating series of tree, the best tree is chosen by measuring a training set or cross-validation.

**3.  Alpha-beta pruning**

* It is a search algorithm, which improves the minimax algorithm by eliminating branches which will not be able to give further outcome.
* Let **alpha (α)** be the value of the best choice along the path for higher value as **MAX**.
* Let **beta (β)** be the value of the best choice along the path for lower value as **MIN**.
* While working with decision tree, the problem of missing values (those values which are missing or wrong)  may occur.
* So, one of the most common solution is to label that missing value as **blank**.

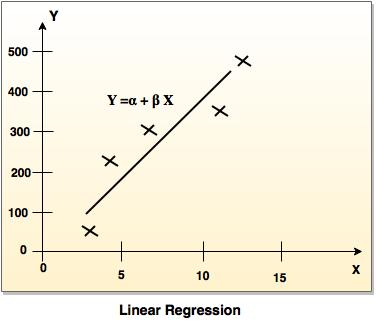
## Prediction

* Prediction deals with some variables or fields, which are available in the data set to predict unknown values regarding other variables of interest.
* Numeric prediction is the type of predicting continuous or ordered values for given input.  
  **For example:** The company may wish to predict the potential sales of a new product given with its price.
* The most widely used approach for numeric prediction is regression.

Regression involves **predictor variable** (the values which are known) and **response variable** (values to be predicted).  
  
**The two basic types of regression are:**

1. Linear 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.  
  **Y =  α + Β 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.



2. Multiple regression model

* 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.  
  **Y = a0 + a1 X1 + a2 X2 +.........+ak Xk +e**  
  **where,**  
  **'Y'** is the response variable.  
  **X1 + X2 + Xk** are the independent predictors.  
  **'e'** is random error.  
  **a0, a1, a2, ak** are the regression  coefficients.
* Naive Bays Classification Solved example
* **Bike Damaged example:** In the following table attributes are given such as color, type, origin and subject can be yes or no.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Bike No** | **Color** | **Type** | **Origin** | **Damaged?** |
| 10 | Blue | Moped | Indian | Yes |
| 20 | Blue | Moped | Indian | No |
| 30 | Blue | Moped | Indian | Yes |
| 40 | Red | Moped | Indian | No |
| 50 | Red | Moped | Japanese | Yes |
| 60 | Red | Sports | Japanese | No |
| 70 | Red | Sports | Japanese | Yes |
| 80 | Red | Sports | Indian | No |
| 90 | Blue | Sports | Japanese | No |
| 100 | Blue | Moped | Japanese | Yes |

* **Solution:**  
  **Required formula:**  
                **P (c | x) = P (c | x) P (c) / P (x) In the fields of science, engineering and statistics, the accuracy of a measurement system is the degree of closeness of measurements of a quantity to that quantity's true value.**  
    
  **Where,**     
  **P (c | x)** is the posterior of probability.  
  **P (c)**  is the prior probability.  
  **P (c | x)** is the likelihood.  
  **P (x)** is the prior probability.  
    
  It is necessary to classify a <Blue, Indian, Sports>, is unseen sample, which is not given in the data set.  
  **So the probability can be computed as:**  
         
  P (Yes) = 5/10  
  P (No) = 5/10

|  |  |
| --- | --- |
| **Color** |  |
| P(Blue|Yes) = 3/5 | P(Blue|No) = 2/5 |
| P(Red|Yes) = 2/5 | P(Red|No) = 3/5 |
| **Type** |  |
| P(Sports|Yes) = 1/5 | P(Sports|No) = 3/5 |
| P(Moped|Yes) = 4/5 | P(Moped|No) = 2/5 |
| **Origin** |  |
| P(Indian|Yes) = 2/5 | P(Indian|No) = 3/5 |
| P(Japanese|Yes) = 3/5 | P(Japanese|No) = 2/5 |

* So, unseen example X = <Blue, Indian, Sports>  
    
  P(X|Yes). P(Yes) = P(Blue|Yes). P(Indian|Yes). P(Sports|Yes). P(Yes)  
                               = 3/5\*2/5\*1/5\*5/10 = 0.024  
    
  P(X|No). P(No) = P(Blue|No). P(Indian|No). P(Sports|No).P(No)  
                             = 2/5\*3/5\*3/5\*5/10 = 0.072  
    
  So, 0.072 > 0.024 so example can be classified as **NO**.

Introduction

* It is a data mining technique used to place the data elements into their related groups.
* Clustering is the process of partitioning the data (or objects) into the same class, The data in one class is more similar to each other than to those in other cluster.
* The process of partitioning data objects into subclasses is called as cluster.
* A cluster consists of data object with high inter similarity and low intra similarity.
* The quality of cluster depends on the method used.
* Clustering is also called as data segmentation, because it partitions large data sets into groups according to their similarity

**Clustering can be helpful in many fields, such as:**  
**1. Marketing:**  
Clustering helps to find group of customers with similar behavior from a given data set customer record.  
  
**2. Biology:**  
Classification of plants and animal according to their features.  
  
**3. Library:**  
Clustering is very useful in book ordering.

Types of clustering

**Clustering methods can be classified into the following categories:**  
  
**1. Partitioning**  
In this approach, several partitions are created and then evaluated based on given criteria.  
  
**2.  Hierarchical method**  
In this method, the set of data objects are decomposed (multilevel) hierarchically by using certain criteria.  
  
**3. Density-based method**  
This method is based on density (density reachability and density connectivity).  
         
**4. Grid-based methods**  
This approach is based on multi-resolution grid data structure.

Classification vs Clustering

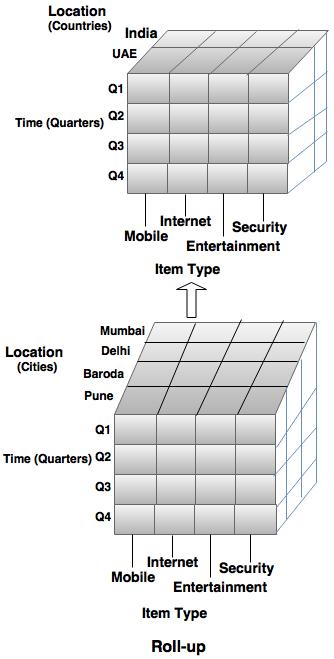
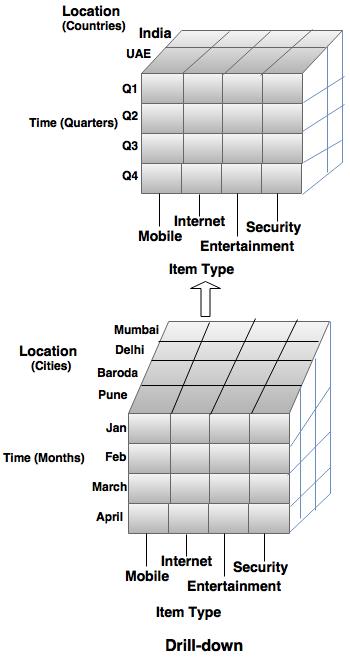
|  |  |
| --- | --- |
| **Classification** | **Clustering** |
| It is supervised learning. | It is unsupervised learning. |
| Classification contains previously categorized training set. | In clustering, the characteristics of similarity of data is not known. |
| Decision tree is used to partition and segment record. | There are a variety of algorithms for clustering, which generally share the same property of interactively assigning records to a cluster. |

* [K-means Clustering in Data Mining](https://www.tutorialride.com/data-mining/k-means-clustering-in-data-mining.htm)
* [Hierarchical Clustering in Data Mining](https://www.tutorialride.com/data-mining/hierarchical-clustering-in-data-mining.htm)
* OLAP (Online Analytical Processing) provides the support for the **multidimensional view** of the data.
* It provides easy and efficient access to the various views of information to the users.
* The complex queries are also processed by using OLAP. It is easy to analyze information by processing multidimensional views of the data.
* The data warehouse is used to analyze the information, where the ample amount of historical data is stored.

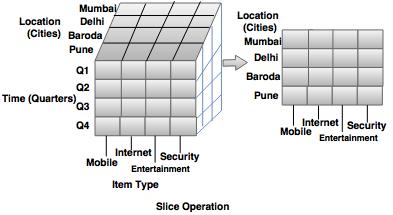
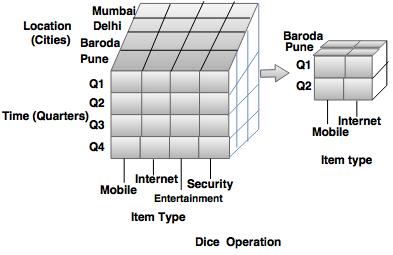
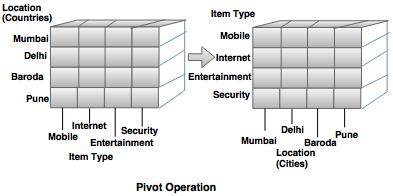
OLAP Operations

OLAP techniques are applied to retrieve the information from the data warehouse in the form of OLAP multidimensional databases.  
  
**Multi-dimensional model has two types of tables:**  
  
**1. Dimension tables:** It contains the attributes of dimensions.  
**2. Fact tables:** It consists of the facts or measures.  
The following OLAP operations are performed to implement OLAP:  
  
**1. Roll up or consolidation**

* Multi-dimensional databases have hierarchies with respect to the dimensions.
* Consolidation is rolling up or adding data relationship with respect to one or more dimensions.
* This hierarchy can be as,  the total order street<city<State<country.
* The roll up operation aggregates the data by city to country by location hierarchy as shown in the following diagram.

  
  
**2. Drill Down**  
In drill down operation, the view is changed to a greater level of detail.  
**For example:** In the diagram shown below, the drill operations are performed on the upper cube by stepping down a concept hierarchy for time. It can be defined as day<month<quarter<year.  
  
  
  
**3. Slice Operation**

* Slicing is referred to an ability to look at a database from various points of view.
* Slice operation selects one dimension of the given cube and creates a sub-cube, as  illustrated in the following diagram.

  
  
**4. Dice operation**  
Two or more dimensions are selected from the given cube to produce a sub cube as shown in the following diagram.  
  
  
  
**5. Pivot / Rotate**  
Pivot technique is used for visualization of data. It rotates the data axis to give  another presentation of the data, as shown in the following schematic.  
  


Approaches to OLAP

The OLAP systems are categorized using the different techniques mentioned below:  
  
**1. MOLAP**  
In MOLAP, data is stored in a multidimensional cube(array storage) rather than the relational database.  
  
**2. ROLAP**

* ROLAP works with relational databases. The base data and the dimension table are stored as relational tables and new table are created to hold an aggregated information.
* Each action of slicing and dicing operation is equivalent to adding a 'WHERE' clause in the SQL statement.

**3. HOLAP**

* HOLAP has the combine advantages of MOLAP and ROLAP.
* When detail information is required, the database is divided into relational database and specialized storage.

**For example:** The Microsoft SQL server 7.0 OLAP services support a hybrid OLAP server.

Market Basket Analysis

* It is a **modeling technique** that helps to identify which items should be purchased together.
* Assume that, there are large number of items  **like** “Tea”, “Coffee”, “Milk” “Sugar”. Among these, the customer buys the subset of items as per the requirement and market gets the information of items which customer has purchased together. So, the market uses this information to put the items on different positions (or locations).

Implementation of market based analysis

* The market basket analysis is used to decide the perfect location, where the items can be placed inside the store.  
  **For example:** If the customer buys a coffee, it is possible that the customer may buy milk or sugar along with coffee.
* So keeping the coffee and sugar next to each other in store will be helpful to customers to buy the items together and improves sales of the company.
* The problem of large volume transactions can be minimized by using **differential market basket analysis**, which is capable of finding interesting results and eliminates the large volume.

Frequent Item-sets, Closed item-sets and Association Rules

Frequent Item-sets

* Frequent item-sets are used in association rules.
* An itemset is said to be frequent, if X's support is no less than a minimum support threshold.
* A frequent item-set is a set of items that may appear at least in pre- defined number of transactions.

**Steps to find frequent item-sets:**  
**1.**A level wise search can be carried out to find the frequent-1 items (set of size 1), then proceed to find frequent- 2 items and so on.  
**2.**  All maximum frequent item-sets are searched in this step.

Closed Item-sets

* Consider two item-sets 'X' and 'Y', if each item of 'X' is in 'Y', but there is at least one item of 'Y', that is not in 'X' then 'Y' is not a proper super-item set of 'X'. In this case item set is closed.
* If 'X' is closed and frequent, then it is called as **closed frequent item-set.**

Association Rules

* The objects or items from relational databases, transactional databases or other information repositories are considered to find frequent patterns, associations and correlations.
* Association rules search for interesting relationships among the items in a given data set by examining transactions.  
  **For example -** Frequently purchased item- sets can be easily found out by examining the shop cart. This helps in advertising or for placement of goods in the store.
* Association rules have general form:  
  **I1  →  (where I1 ∩ I2 = 0)**
* The rule can be read as, Given that someone has purchased the items from the set**I1, then** they are likely to also buy the items in the set **I2.**

Large Item-sets

* It is a set of single items, from transactions.
* If some items occur together, then they can form an association rule.

**Support:**

* It is the percentage of the transactions in which the items appear.
* If **A => B**
* **Support (A → B) =  # \_ tuples containing both A and B / total #\_tuples**
* The support for an association X => Y is the percentage of transactions in the database that contains **X U Y.**

**Confidence:**

* The confidence or strength for an association rule **A => B** is the ratio of the number of transactions that contain **A.**
* Consider a rule **A => B, which** is a measure of ratio of the number of tuples containing both A and B to the number of tuples containing **'A'.**

**Confidence (A → B) = #\_tuples containing both A and B / #\_tuples\_containing A**

Solved example of apriori algorithm

**Example:** Find the frequent item sets in the given table, with minimum support of 50% and confidence 50%.

|  |  |
| --- | --- |
| **TID** | **List of items** |
| T2000 | A, B, C |
| T1000 | A, C |
| T4000 | A, D |
| T5000 | B, E, F |

**Solution:**  
**Step 1:** Scan D for count each candidate. The candidate list is {A, B, C, D, E, F}  
C1 =

|  |  |
| --- | --- |
| **Items** | **Support** |
| {A} | 3 |
| {B} | 2 |
| {C} | 2 |
| {D} | 1 |
| {E} | 1 |
| {F} | 1 |

**Step 2:** Compare the candidate support with minimum support count (50%)  
L1 =

|  |  |
| --- | --- |
| **Items** | **Support** |
| {A} | 3 |
| {B} | 2 |
| {C} | 2 |

**Step 3:** Generate candidate C2 from L1  
C2  =

|  |
| --- |
| **Items** |
| {A, B} |
| {A, C} |
| {B, C} |

**Step 4:** Scan D for count of each candidate in C2 and find the support.  
C2 =

|  |  |
| --- | --- |
| **Items** | **Support** |
| {A, B} | 1 |
| {A, C} | 2 |
| {B, C} | 1 |

**Step 5:** Compare candidate (C2) support count with the minimum support count.  
L2  =

|  |  |
| --- | --- |
| **Items** | **Support** |
| {A,C} | 2 |

**Step 6:**    
Data contains the frequent item 1 (A, C), so that the association rule that can be generated from 'L' are as shown in the following table with the support and confidence.

|  |  |  |  |
| --- | --- | --- | --- |
| **Association Rule** | **Support** | **Confidence** | **Confidence %** |
| A - > C | 2 | 2/3 = 0.66 | 66 % |
| C - > A | 2 | 2/2 = 1 | 100 % |

**So the final rules are:**  
**Rule 1:** A - > C  
**Rule 2:** C - > A

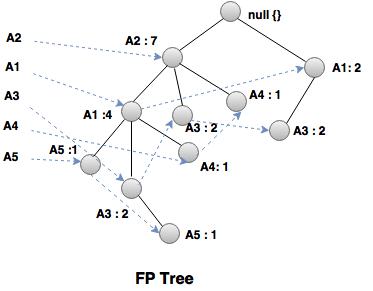
FP- Growth Algorithm by Jiawei Han et al.

* This algorithm allows to find frequent item-set without generation of candidate item-set.
* Once the F- P tree is generated, it is mined by calling FP\_growth (FP\_tree, null).  
  **Procedure FP\_ growth (Tree, α):**
* **If** tree consist of a single path 'P', then for each combination (which is denoted as 'β' ) of the path 'P' generates the pattern **β U , α** with support count = of nodes in **β;**
* **Else**, for each **ai** in the header of tree, generates the patterns  **β = ai.support\_count;**
* So, construct β's conditional pattern base first and then β's conditional FP \_Tree Treeβ;
* **If** Tree β ≠ Ф
* **Then**, call FP\_ growth (Treeβ, β); }

**Example:**  
**Given transaction database, find the frequent patterns with min support = 2.**

|  |  |
| --- | --- |
| **TID** | **List of item** |
| T1 | A1, A2, A5 |
| T2 | A2, A4 |
| T3 | A2, A3 |
| T4 | A1, A2, A4 |
| T5 | A1, A3 |
| T6 | A2, A3 |
| T7 | A1, A3 |
| T8 | A1, A2, A3, A5 |
| T9 | A1, A2, A3 |

|  |  |
| --- | --- |
| **Item** | **Support** |
| A1 | 6 |
| A2 | 7 |
| A3 | 6 |
| A4 | 2 |
| A5 | 2 |



|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Conditional pattern base** | **Conditional FP-tree** | **Frequent patterns generation** |
| A5 | {(A2 A1: 1)} | { A2: 2, A1: 2} | A2 A5: 2, A1 A5: 2, A2 A1A5: 2 |
| A4 | {A2 A1 : 1), (A2:1)} | {A2: 2} | A2 A4: 2 |
| A3 | {( A2, A1 :2), (A1: 2)} | (A2: 4, A1:2), (A1:2) | A2 A3: 4, A1, A3: 2, A2 A1A3: 2 |
| A1 | {(A2: 4)} | {(A2: 4)} | A2 A1: 4 |