MIT-WPU
|| विश्वशान्तिर्ध्रुवं ध्रुवा ||
Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

(Computer Engineering and Technology)
(TYB.Tech)

UNIT V- Data Mining_Classification and Clustering

Introduction to Data Mining, Data Mining Techniques, Supervised,Semi-Supervised, and Unsupervised Methods, Classification: BasicConcepts, Decision Tree Induction, Bayesian Classification

Clustering Techniques: Basic concepts, Partition based Clustering:k-Means

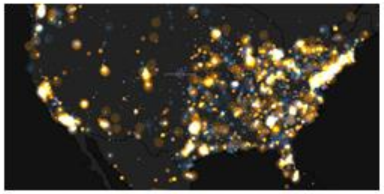
Large-scale Data is Everywhere!

- There has been enormous data growth in both commercial and scientific databases due to advances in data generation and collection technologies
- Gather whatever data you can whenever and wherever possible.
- Gathered data will have value either for the purpose collected or for a purpose not envisioned.
- Need of automated processing and identifying hidden knowledge from data

Cyber Security

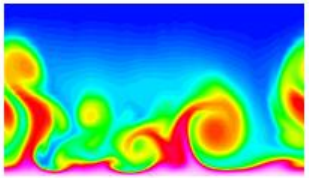
E-Commerce

Traffic Patterns



Social Networking: Twitter

Sensor Networks

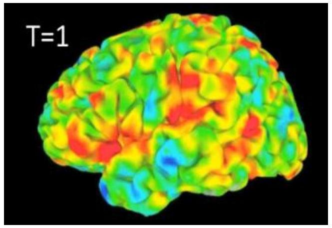


Computational Simulations

Why Data Mining? Scientific Viewpoint

- Data collected and stored at enormous speeds
  - remote sensors on a satellite
    - NASA EOSDIS archives over petabytes of earth science data / year
  - telescopes scanning the skies
    - Sky survey data
  - High-throughput biological data
  - scientific simulations
    - terabytes of data generated in a few hours

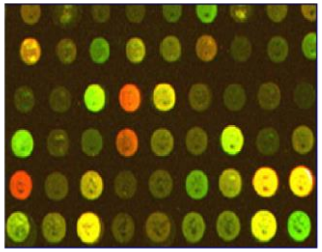
- Data mining helps scientists
  - in automated analysis of massive datasets
  - In finding hidden knowledge and inferences from data

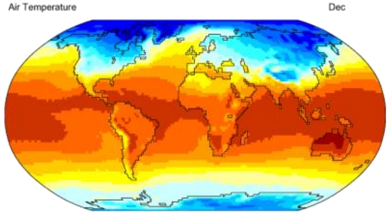


fMRI Data from Brain


Sky Survey Data

Gene Expression Data





Surface Temperature of Earth

Why Data Mining? Commercial Viewpoint

- Lots of data is being collected and warehoused
  - Web data
    - Yahoo has Peta Bytes of web data
    - Facebook has billions of active users
  - purchases at department/ grocery stores, e-commerce
    - Amazon handles millions of visits/day
  - Bank/Credit Card transactions

- Computers have become cheaper and more powerful

- Competitive Pressure is Strong
  - Need of Finding the inferences, ;likeliness , hidden knowledge from gathered data which Provide better, customized services for an edge (e.g. in Customer Relationship Management)

Need of Data Mining

- The Explosive Growth of Data: from terabytes to petabytes
  - Data collection and data availability
    - Automated data collection tools, database systems, Web, computerized society
  - Major sources of abundant data
    - Business: Web, e-commerce, transactions, stocks, …
    - Science: Remote sensing, bioinformatics, scientific simulation, …
    - Society and everyone: news, digital cameras, YouTube
- We are drowning in data, but starving for knowledge!
- "Necessity is the mother of invention"—Data mining—Automated analysis of massive data sets

An example application

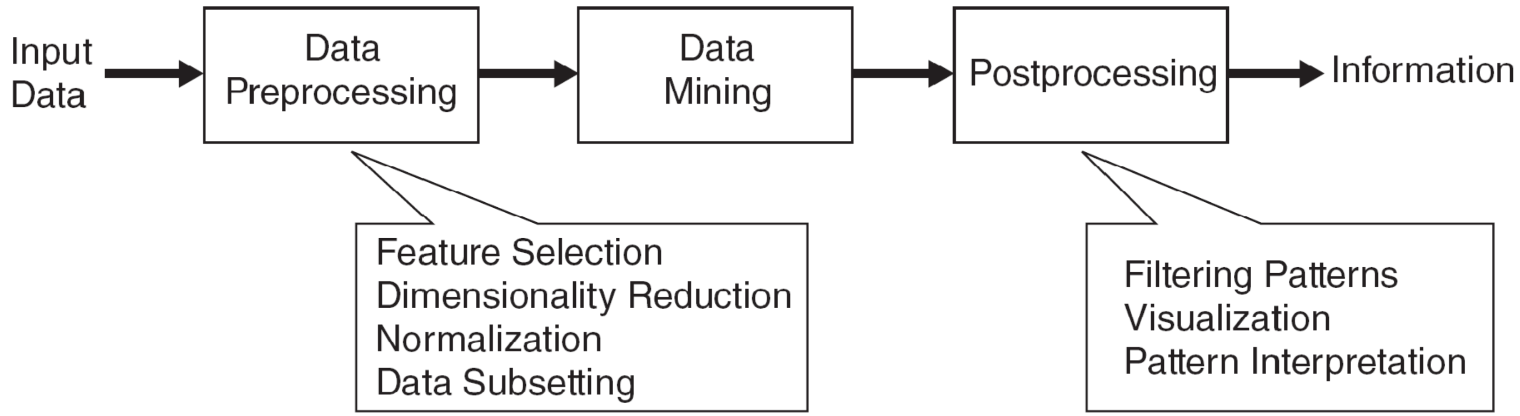
- An emergency room in a hospital measures 17 variables (e.g., blood pressure, age, etc) of newly admitted patients.
- A decision is needed: whether to put a new patient in an intensive-care unit.
- Due to the high cost of ICU, those patients who may survive less than a month are given higher priority.
- Problem: to predict high-risk patients and discriminate them from low-risk patients.

Another application

- A credit card company receives thousands of applications for new cards. Each application contains information about an applicant,
  - age
  - Marital status
  - annual salary
  - outstanding debts
  - credit rating
  - etc.
- Problem: to decide whether an application should approved, or to classify applications into two categories, approved and not approved.

What is Data Mining

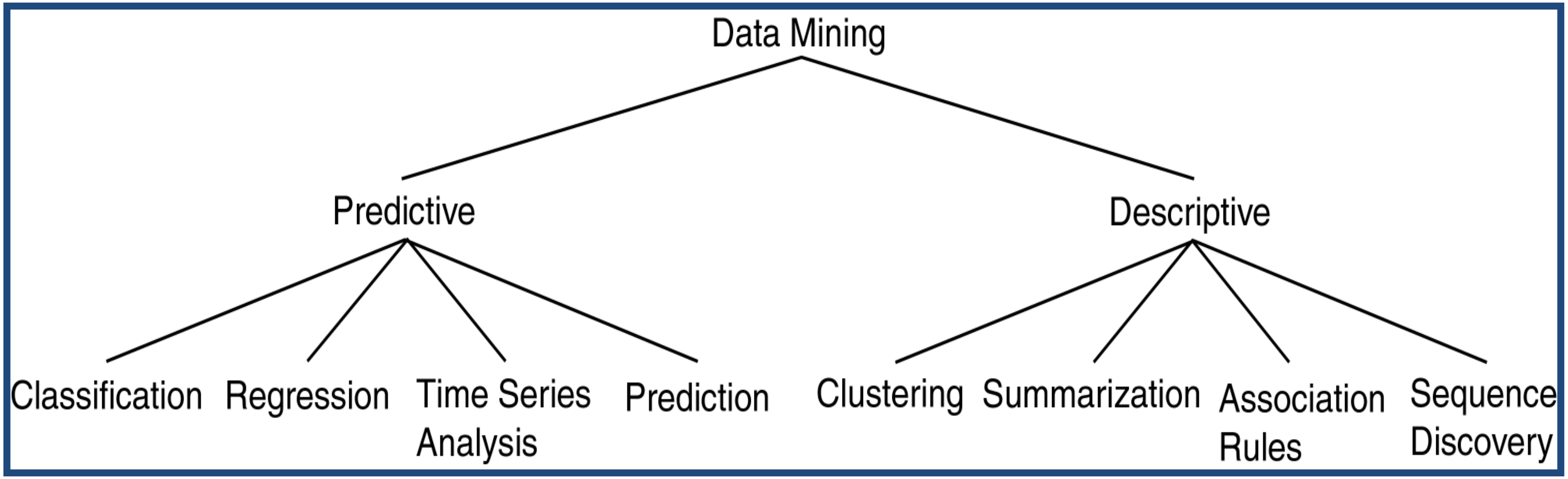
- Data Mining is Non-trivial extraction of implicit, previously unknown and potentially useful information from data
- Data Mining is Exploration & analysis, by automatic or semi-automatic means, of large quantities of data in order to discover meaningful patterns
-



Data Mining Task

- Predictive Methods
  - Use some variables to predict unknown or future values of other variables.
- Descriptive Methods
  - Find human-interpretable patterns that describe the data.

Data Mining Models and Tasks



Data Mining Task

- Predictive Methods
  - Use some variables to predict unknown or future values of other variables.
- Descriptive Methods
  - Find human-interpretable patterns that describe the data.

Supervised vs. Unsupervised Learning

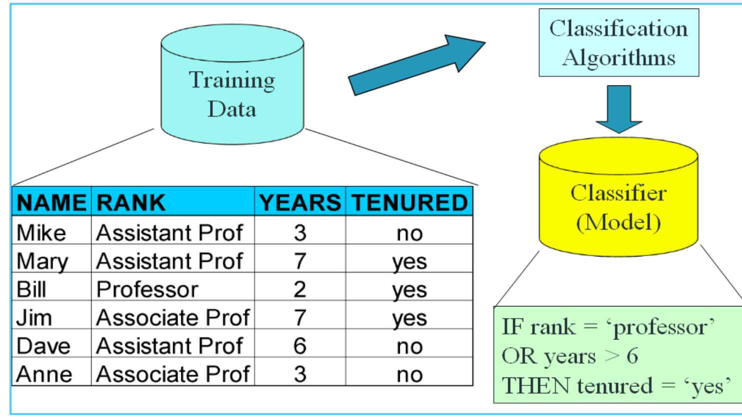
Supervision: The training data (observations, measurements, etc.) are accompanied by **labels** indicating the class of the observations
New data is classified based on the training set

Unsupervised learning (clustering)

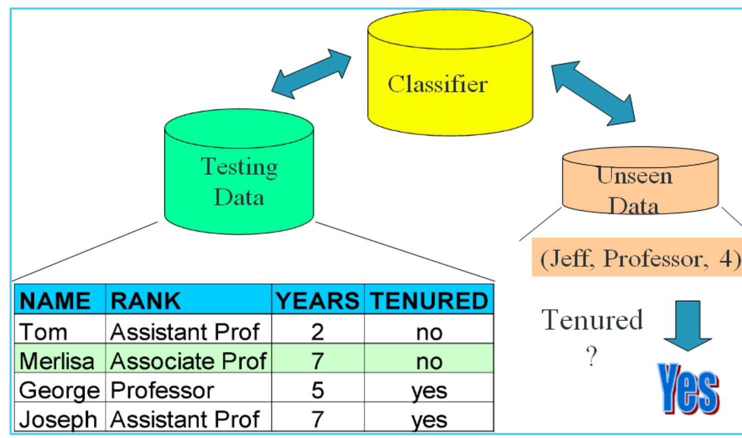
ning data is unknown
ements, observations, etc. with the aim of
nce of classes or clusters in the data

Supervised Learning

- Prediction methods are commonly referred to as supervised learning.
- Supervised methods are thought to attempt the discovery of the relationships between input attributes and a target attribute.
- A training set is given and the objective is to form a description that can be used to predict unseen examples.
- Methods:
  - Classification
    - The domain of the target attribute is finite and categorical.
    - A classifier must assign a class to a unseen example.
  - Regression
    - The target attribute is formed by infinite values.
    - To fit a model to learn the output target attribute as a function of input attributes.
  - Time Series Analysis
    - Making predictions in time.



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no



NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

Unsupervised Learning

- There is no supervisor and only input data is available.
- The aim is to find regularities, irregularities, relationships, similarities and associations in the input.

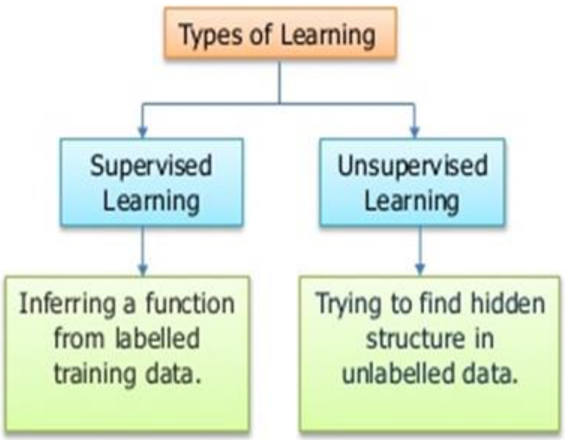
- Methods:
  - Clustering
  - Association Rules
  - Pattern Mining
    - It is adopted as a more general term than frequent pattern mining or association mining.
  - Outlier Detection
    - Process of finding data examples with behaviours that are very different from the expectation (outliers or anomalies).

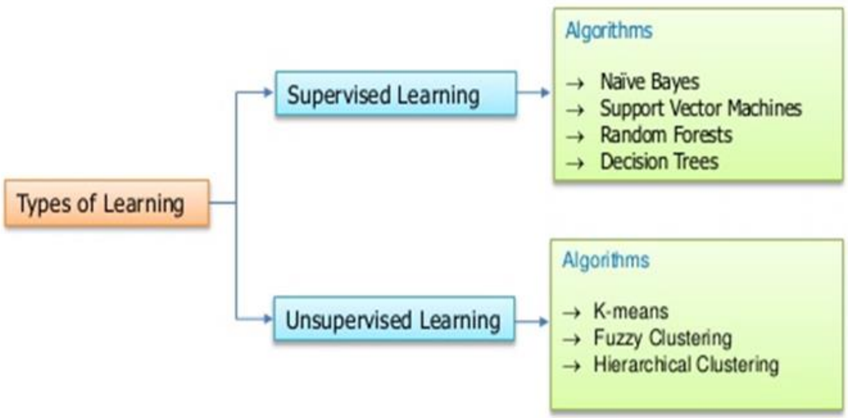
Semi Supervised Learning

- Semi-supervised learning is a class of machine learning tasks and techniques that also make use of unlabeled data for training – typically a small amount of labeled data with a large amount of unlabeled data

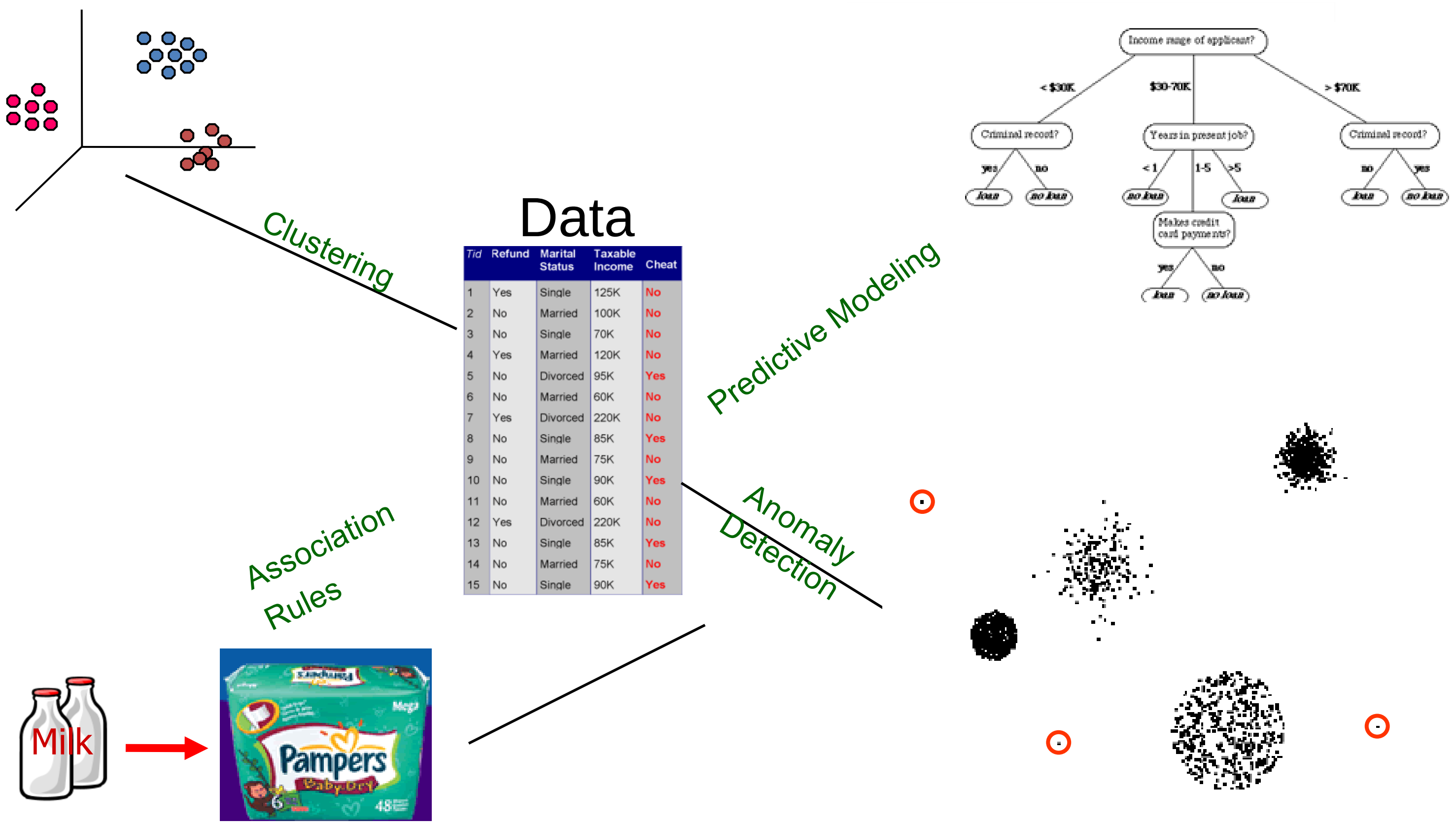
- Semi-supervised learning falls between unsupervised learning (without any labeled training data) and supervised learning (with completely labeled training data).

Supervised , Unsupervised Learning Algori



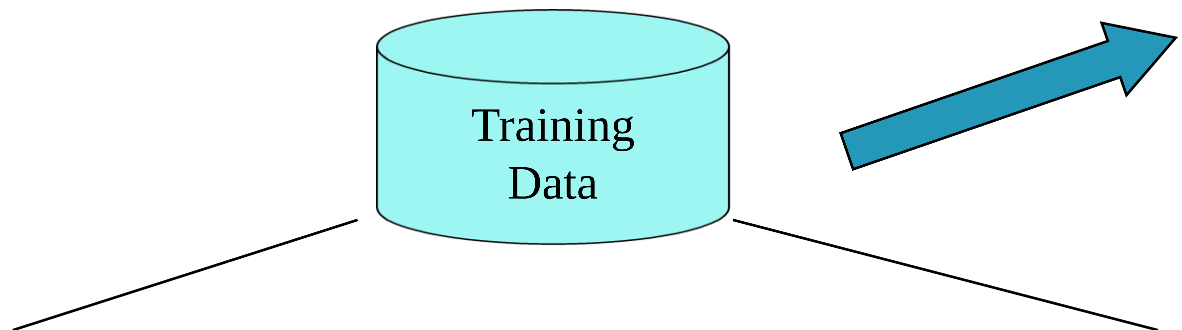


Data Mining Task..

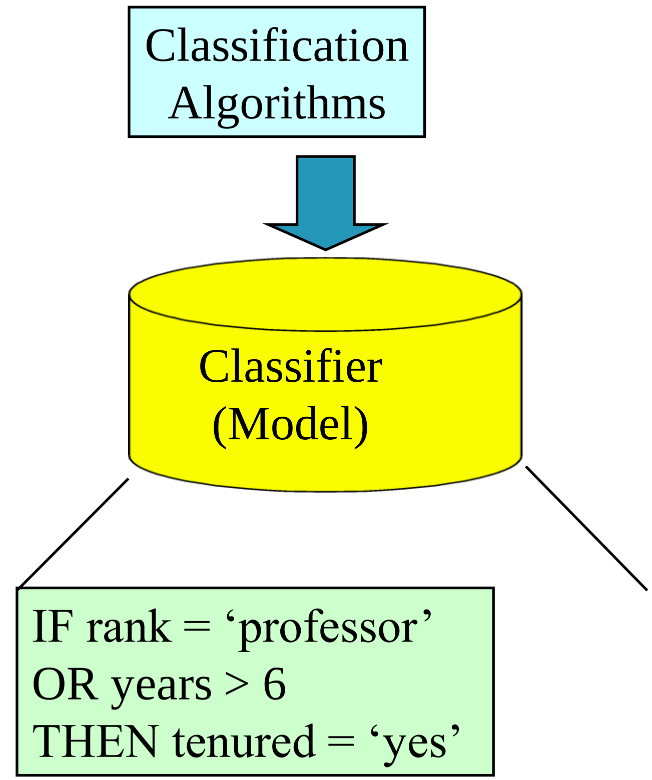


Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes
11	No	Married	60K	No
12	Yes	Divorced	220K	No
13	No	Single	85K	Yes
14	No	Married	75K	No
15	No	Single	90K	Yes

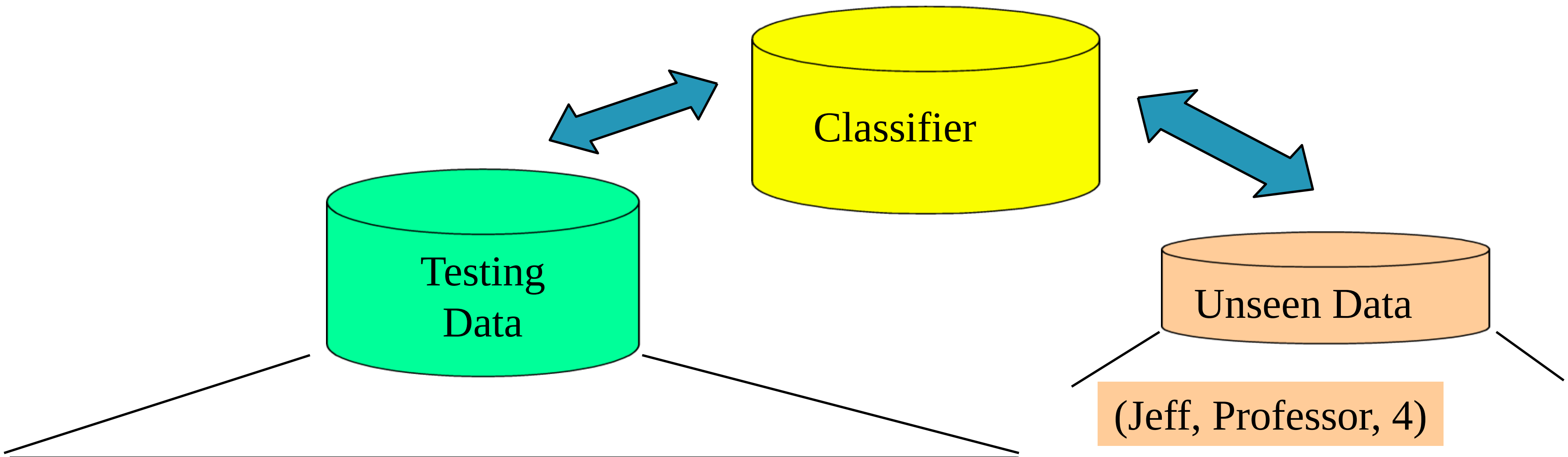
Process (1): Model Construction



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no



Process (2): Using the Model in Prediction



NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

Tenured?

Yes

Classification

Classification maps data into predefined groups or classes

It is useful in

- Supervised learning
- Pattern recognition
- Prediction

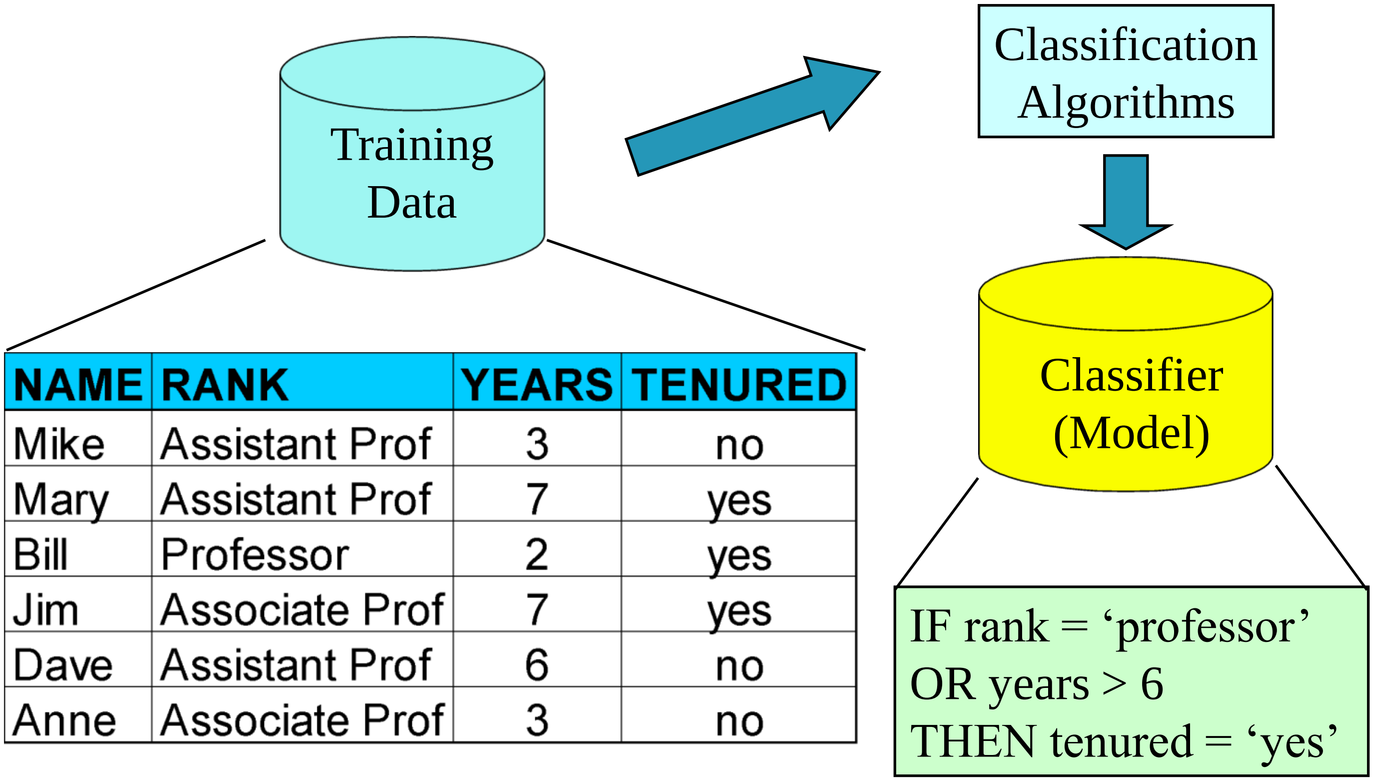
Classification Vs Prediction

- Classification
  - Predicts categorical class labels (discrete or nominal)
  - Classifies data (constructs a model) based on the training set and the values (class labels) in a classifying attribute and uses it in classifying new data
- Prediction
  - Models continuous-valued functions, i.E., Predicts unknown or missing values
- Typical applications
  - Credit approval
  - Target marketing
  - Medical diagnosis
  - Fraud detection

Classification Process

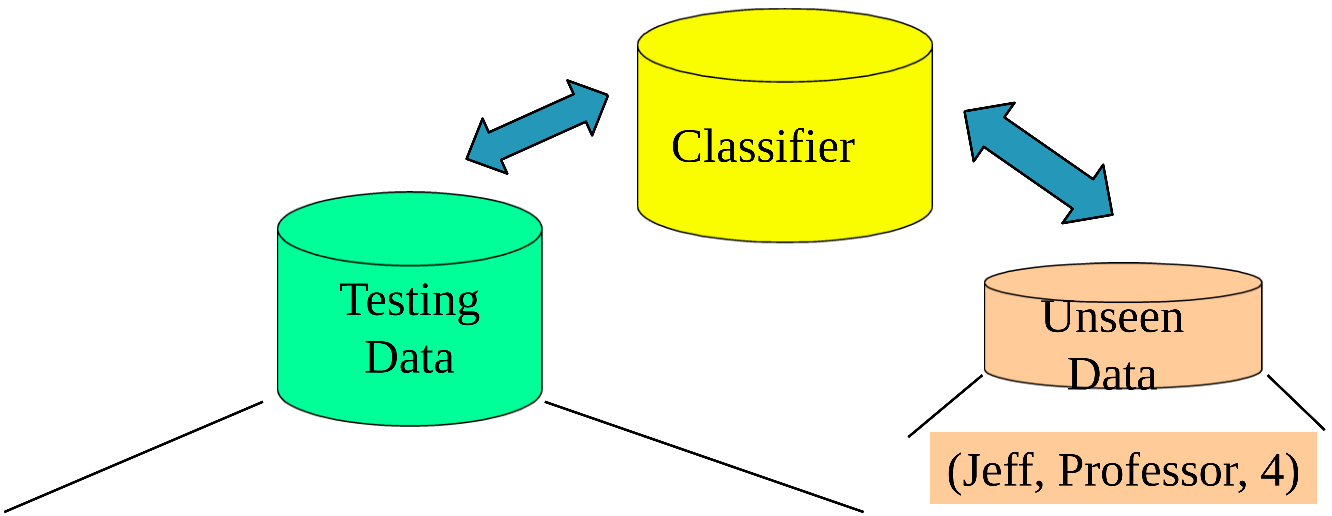
- Model construction: describing a set of predetermined classes
  - Each tuple/sample is assumed to belong to a predefined class, as determined by the class label attribute
  - The set of tuples used for model construction is training set
  - The model is represented as classification rules, decision trees, or mathematical formulae
- Model usage: for classifying future or unknown objects
  - Estimate accuracy of the model
    - The known label of test sample is compared with the classified result from the model
    - Accuracy rate is the percentage of test set samples that are correctly classified by the model
    - Test set is independent of training set, otherwise over-fitting will occur
  - If the accuracy is acceptable, use the model to classify data tuples whose class labels are not known

Step 1: Model Construction



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

Step 2: Model Usage



NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

Tenured?

Yes

Bayesian Classification

- A statistical classifier: performs *probabilistic prediction, i.e.,* predicts class membership probabilities
- Foundation: Based on Bayes' Theorem.
- Performance: A simple Bayesian classifier, *naïve Bayesian classifier*, has comparable performance with decision tree and selected neural network classifiers
- Incremental: Each training example can incrementally increase/decrease the probability that a hypothesis is correct — prior knowledge can be combined with observed data
- Standard: Even when Bayesian methods are computationally intractable, they can provide a standard of optimal decision making against which other methods can be measured

Bayesian Classification - Example

age	income	student	credit rating	com
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Bayesian Classification - Probability

- *Probability:* How likely something is to happen
- Probability of an event happening =

$$\frac{\textit{Number of ways it can happen}}{\textit{Total number of outcomes}}$$

Bayesian Classification Theorem

- Let **X** be a data sample ("*evidence*"): class label is unknown
- Let H be a *hypothesis* that X belongs to class C
- Classification is to determine P(H|**X**), the probability that the hypothesis holds given the observed data sample **X**
- P(H) (*prior probability*), the initial probability
  - E.g., **X** will buy computer, regardless of age, income, …
- P(**X**): probability that sample data is observed
- P(**X**|H) (*posteriori probability*), the probability of observing the sample **X**, given that the hypothesis holds
  - E.g., Given that **X** will buy computer, the prob. that X is 31..40, medium income

Bayesian Classification Theorem

- Given training data **X**, *posteriori probability of a hypothesis* H, P(H|**X**), follows the Bayes theorem
-

$$P(H\,|\,\mathbf{X}) = \frac{P(\mathbf{X}\,|\,H)P(H)}{P(\mathbf{X})}$$

- Informally, this can be written as
  - posteriori = likelihood x prior/evidence
- ***Predicts X belongs to C_2 iff the probability $P(C_i|X)$ is the highest among all the $P(C_k|X)$ for all the k classes***
- Practical difficulty: require initial knowledge of many probabilities, significant computational cost

Naïve Bayesian Classification Theorem

- Let D be a training set of tuples and their associated class labels, and each tuple is represented by an n-D attribute vector **X** = $(x_1, x_2, \ldots, x_n)$
- Suppose there are *m* classes $C_1, C_2, \ldots, C_m$.
- Classification is to derive the maximum posteriori, i.e., the maximal $P(C_i|\mathbf{X})$
- This can be derived from Bayes' theorem

$$P(C_i|\mathbf{X}) = \frac{P(\mathbf{X}|C_i)P(C_i)}{P(\mathbf{X})}$$

- Since P(X) is constant for all classes, only
  - $P(C_i|\mathbf{X}) = P(\mathbf{X}|C_i)P(C_i)$ needs to be maximized

Naïve Bayesian Classification - Example

Class:
C1:buys_computer = 'yes'
C2:buys_computer = 'no'

Data sample
X = (age <=30,
Income = medium,
Student = yes
Credit_rating = Fair)

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Naïve Bayesian Classification - Example

Test for X = (age <= 30 , income = medium, student = yes, credit_rating = fair)

- $P(C_i)$: **P(buys_computer = "yes") = 9/14 = 0.643 P(buys_computer = "no") = 5/14= 0.357**

Compute $P(X|C_i)$ for each class

P(age = "<=30" | buys_computer = "yes") = 2/9 = 0.222
P(age = "<= 30" | buys_computer = "no") = 3/5 = 0.6
P(income = "medium" | buys_computer = "yes") = 4/9 = 0.444
P(income = "medium" | buys_computer = "no") = 2/5 = 0.4
P(student = "yes" | buys_computer = "yes) = 6/9 = 0.667
P(student = "yes" | buys_computer = "no") = 1/5 = 0.2
P(credit_rating = "fair" | buys_computer = "yes") = 6/9 = 0.667
P(credit_rating = "fair" | buys_computer = "no") = 2/5 = 0.4

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

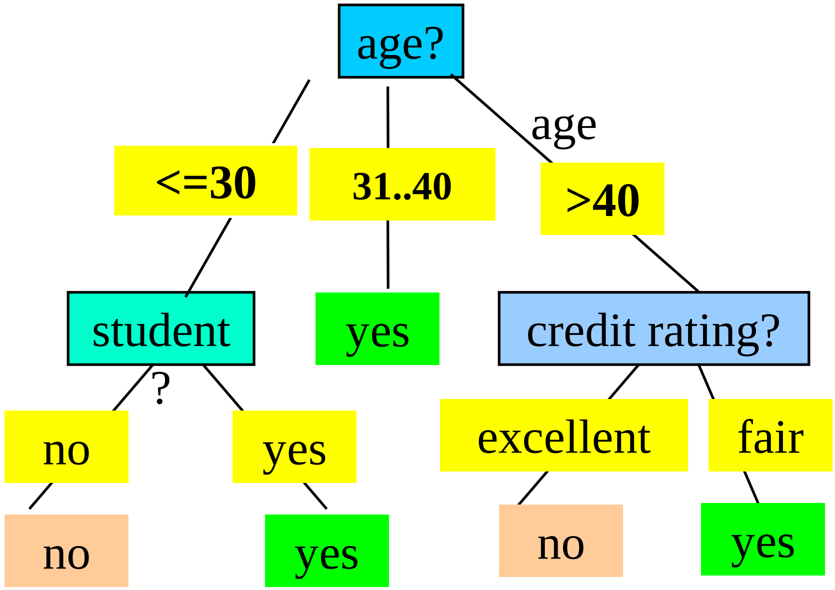
- **$P(X|C_i)$:** P(X|buys_computer = "yes") = 0.222 x 0.444 x 0.667 x 0.667 = 0.044
  P(X|buys_computer = "no") = 0.6 x 0.4 x 0.2 x 0.4 = 0.019
- **$P(X|C_i)*P(C_i)$:** P(X|buys_computer = "yes") * P(buys_computer = "yes") = **0.028**
  P(X|buys_computer = "no") * P(buys_computer = "no") = **0.007**

0.028>0.007 ..
Therefore, X belongs to class ("buys_computer = yes")

Comment on Naive Bayes Classification

- Advantages
  - Easy to implement
  - Good results obtained in most of the cases
- Disadvantages
  - Assumption: class conditional independence, therefore loss of accuracy

Decision Tree Classification

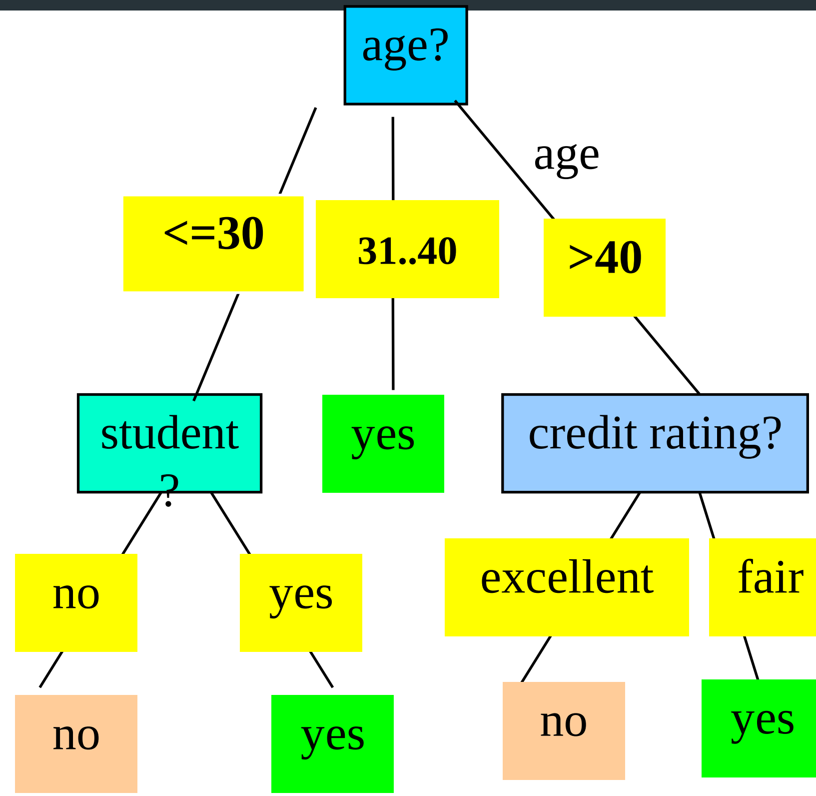


age	income	student	credit rating	buys computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Decision Tree Terminologies

- Decision tree may be n-ary, $n \geq 2$.
- There is a special node called root node.
- Internal nodes are test attribute/decision attribute
- leaf nodes are class labels
- Edges of a node represent the outcome for a value of the test node.
- In a path, a node with same label is never repeated.
- Decision tree is not unique, as different ordering of internal nodes can give different decision tree

Rule Extraction From decision Tree



- Rules are easier to understand than large trees
- One rule is created for each path from the root to a leaf
- Each attribute-value pair along a path forms a conjunction: the leaf holds the class prediction
- Rules are mutually exclusive and exhaustive

Example: Rule extraction from our *buys_computer* decision-tree

IF *age* = young AND *student = no* THEN *buys_computer = no*

IF *age* = young AND *student = yes* THEN *buys_computer = yes*

IF *age* = mid-age THEN *buys_computer = yes*

IF *age* = old AND *credit_rating = excellent* THEN *buys_computer = yes*

IF *age* = young AND *credit_rating = fair* THEN *buys_computer = no*

Decision Tree Algorithm

- Basic algorithm (a greedy algorithm)
  - Tree is constructed in a top-down recursive divide-and-conquer manner
  - At start, all the training examples are at the root
  - Attributes are categorical (if continuous-valued, they are discretized in advance)
  - Examples are partitioned recursively based on selected attributes
  - Test attributes are selected on the basis of a heuristic or statistical measure (e.g., information gain)
- Conditions for stopping partitioning
  - All samples for a given node belong to the same class
  - There are no remaining attributes for further partitioning – majority voting is employed for classifying the leaf
  - There are no samples left

Decision Tree Algorithm

Test Attribute Selection

- **Select the attribute with the highest information gain**
- Let p_i be the probability that an arbitrary tuple in D belongs to class C_i, estimated by $|C_{i,D}|/|D|$
- Expected information (entropy) needed to classify a tuple in D:

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

- Information needed (after using A to split D into v partitions) to classify D:

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times I(D_j)$$

- Information gained by branching on attribute A

$$Gain(A) = Info(D) - Info_A(D)$$

Attribute Selection : Information gain

Class P: buys_computer = “yes”
Class N: buys_computer = “no”

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

1 : Calculate Entropy for Class Labels

$$Info(D) = I(9,5) = -\frac{9}{14}\log_2(\frac{9}{14}) - \frac{5}{14}\log_2(\frac{5}{14}) = 0.940$$

2 : Calculate Information of Each Attribute (one by one)

age	p_i	n_i	$I(p_i, n_i)$
<=30	2	3	0.971
31…40	4	0	0
>40	3	2	0.971

$$Info_{age}(D) = \frac{5}{14}I(2,3) + \frac{4}{14}I(4,0) + \frac{5}{14}I(3,2) = 0.694$$

$\frac{5}{14}I(2,3)$ means “age <=30” has 5 out of 14 samples, with 2 yes’es and 3 no’s.

3 : Calculate Gain of Each Attribute (one by one)

$$Gain(age) = Info(D) - Info_{age}(D) = 0.246$$

$$Gain(income) = 0.029$$

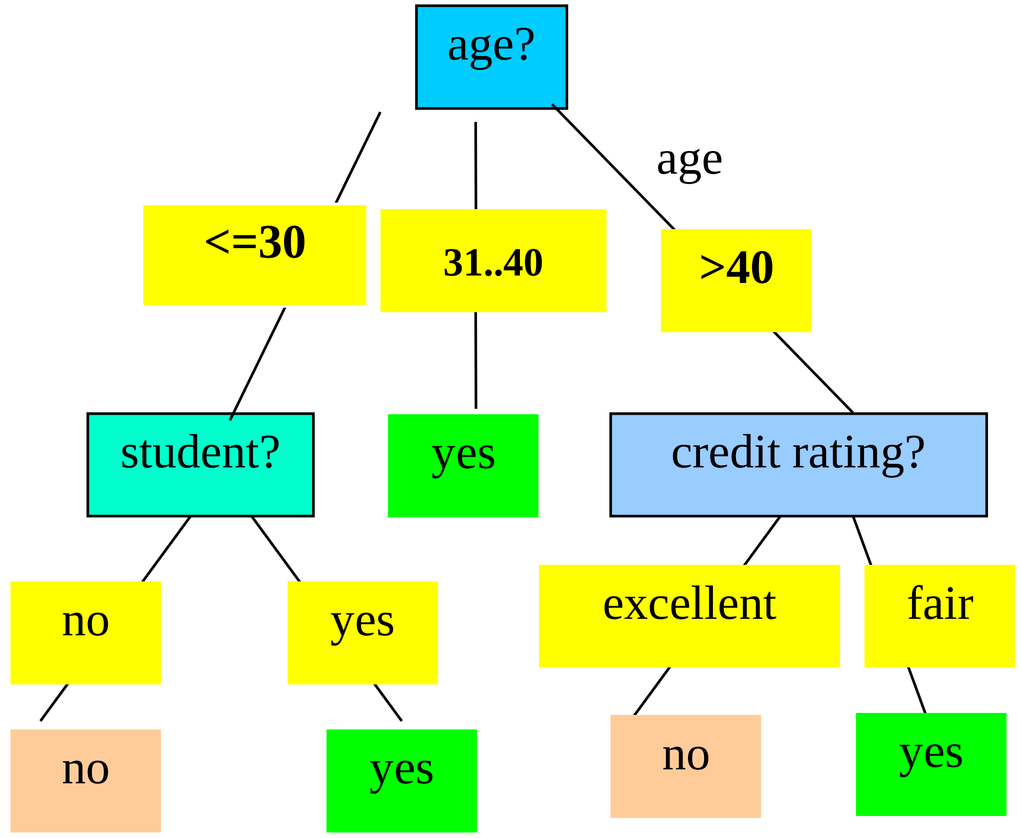
$$Gain(student) = 0.151$$

$$Gain(credit\_rating) = 0.048$$

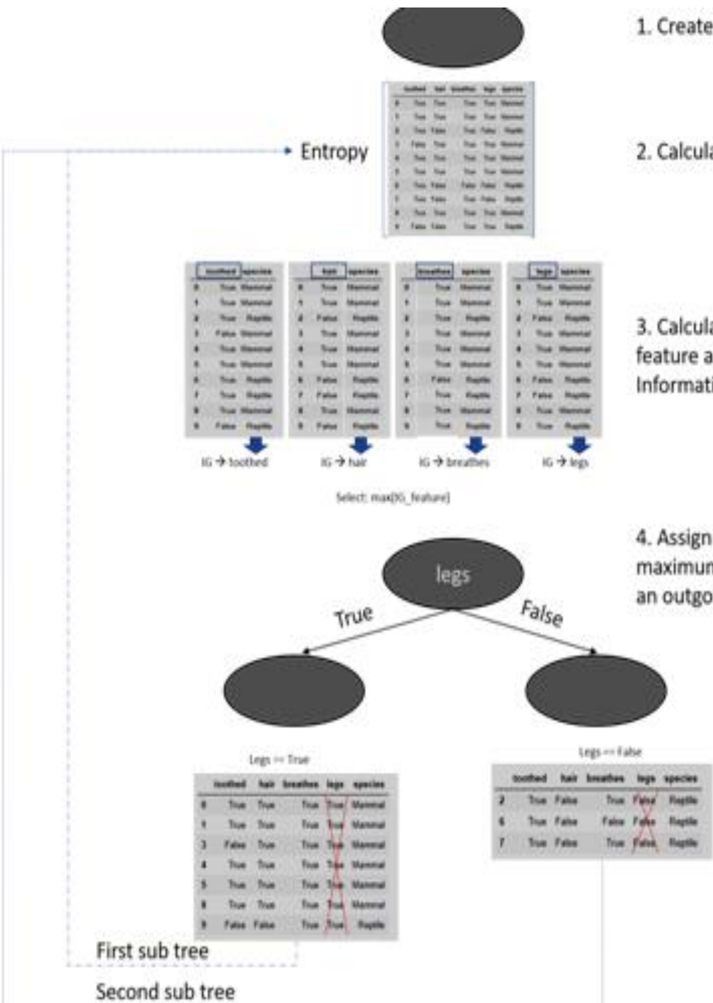
4: Select Attribute with max gain as a root attribute

Gain(age)> any other gain. so age is selected as root node

Final DT of the buys_computer dataset



Example : Usage of Information Gain and Entropy in DT Creation



1. Create a root node

2. Calculate the entropy of the whole (sub) dataset

3. Calculate the Information gain of each single feature and pick that feature with the larges Information gain

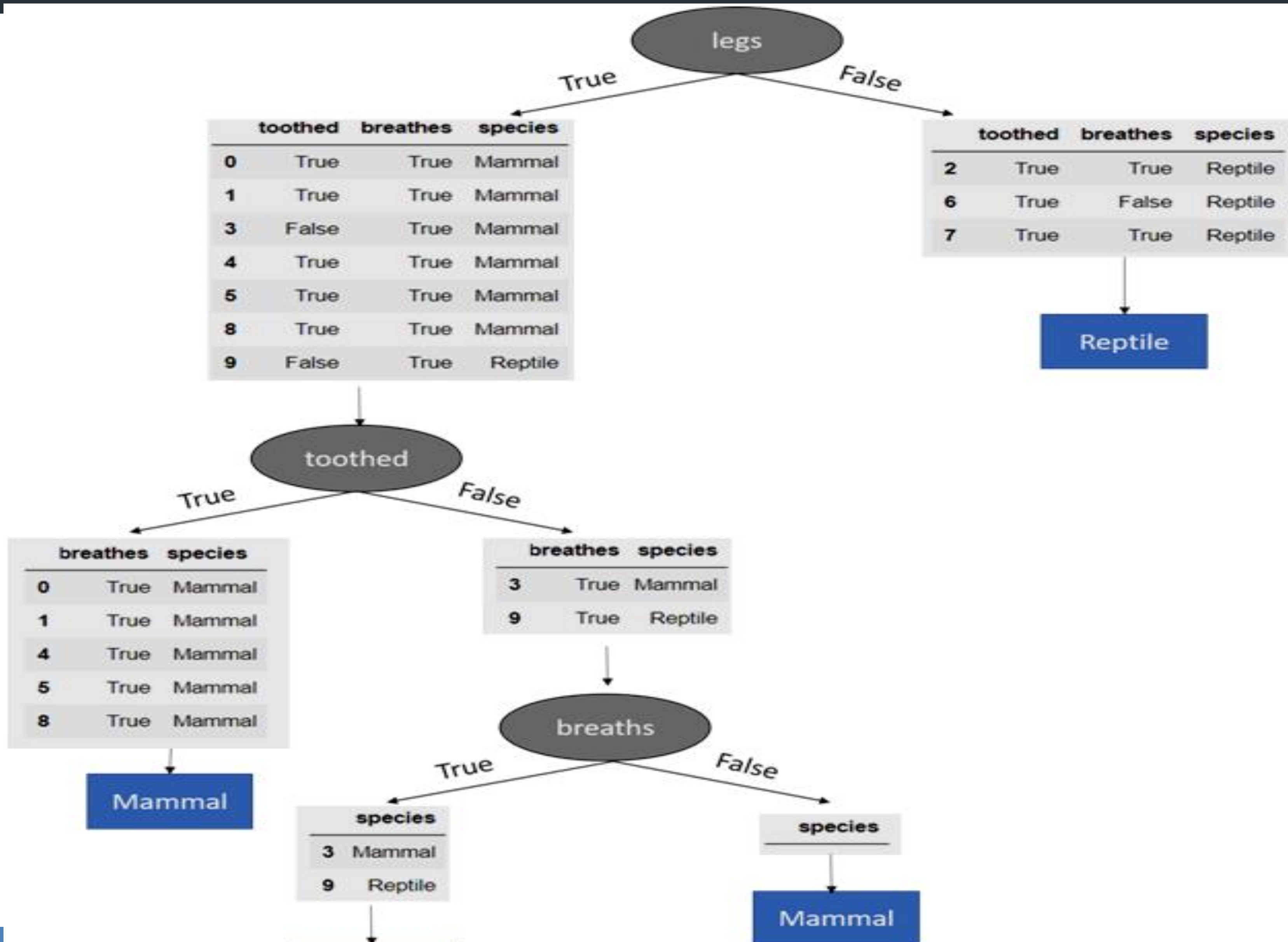
4. Assign the (root) node the label of the feature with the maximum information gain. Grow for each feature value an outgoing branch and add unlabelled nodes at the end

5. Split the dataset along the values of the maximum information gain feature and remove this feature from the dataset

6. For each of the sub_datasets, repeat steps 2 to 5 until a stopping criteria is satisfied →Here the recursion kicks in

	toothed	hair	breathes	legs	species
0	True	True	True	True	Mammal
1	True	True	True	True	Mammal
2	True	False	True	False	Reptile
3	False	True	True	True	Mammal
4	True	True	True	True	Mammal
5	True	True	True	True	Mammal
6	True	False	False	False	Reptile
7	True	False	True	False	Reptile
8	True	True	True	True	Mammal
9	False	False	True	True	Reptile

Decision Tree Algorithm- Example



Summary- Classification Algorithm

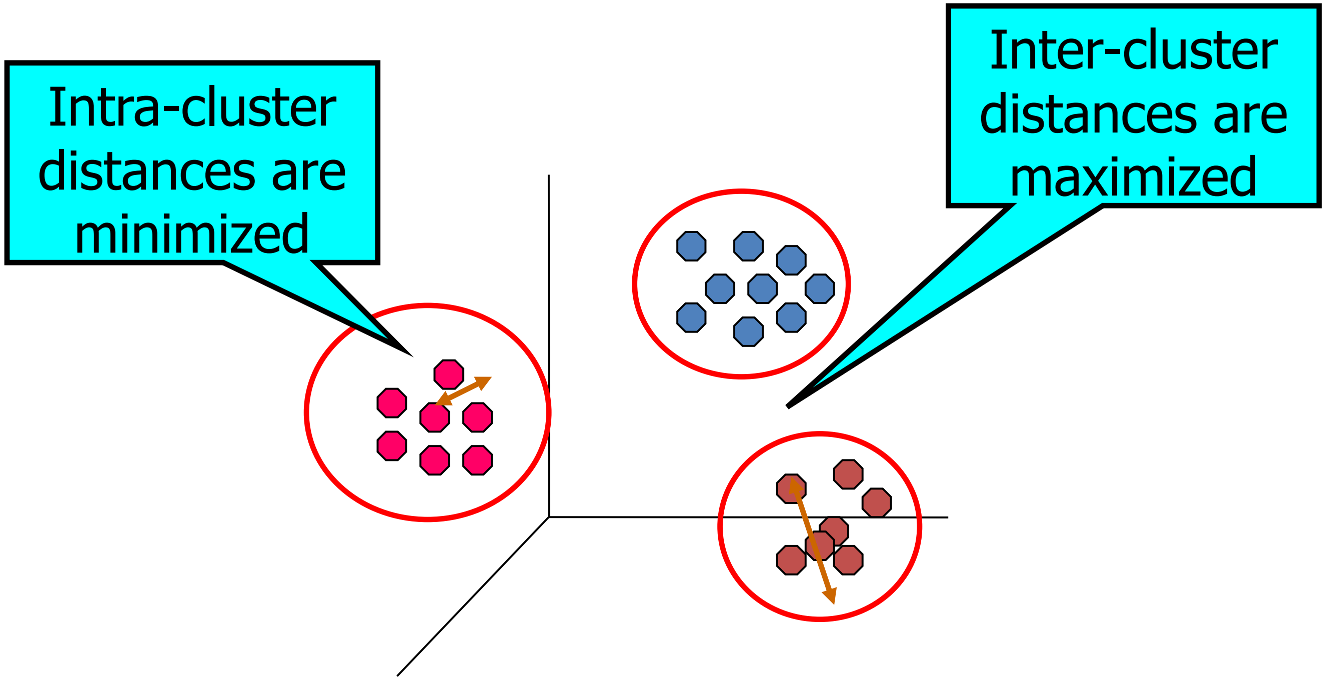
- Classification is a form of data analysis that extracts models describing important data classes.
- Effective and scalable methods have been developed for decision tree induction, Naive Bayesian classification, rule-based classification, and many other classification methods.

Clustering

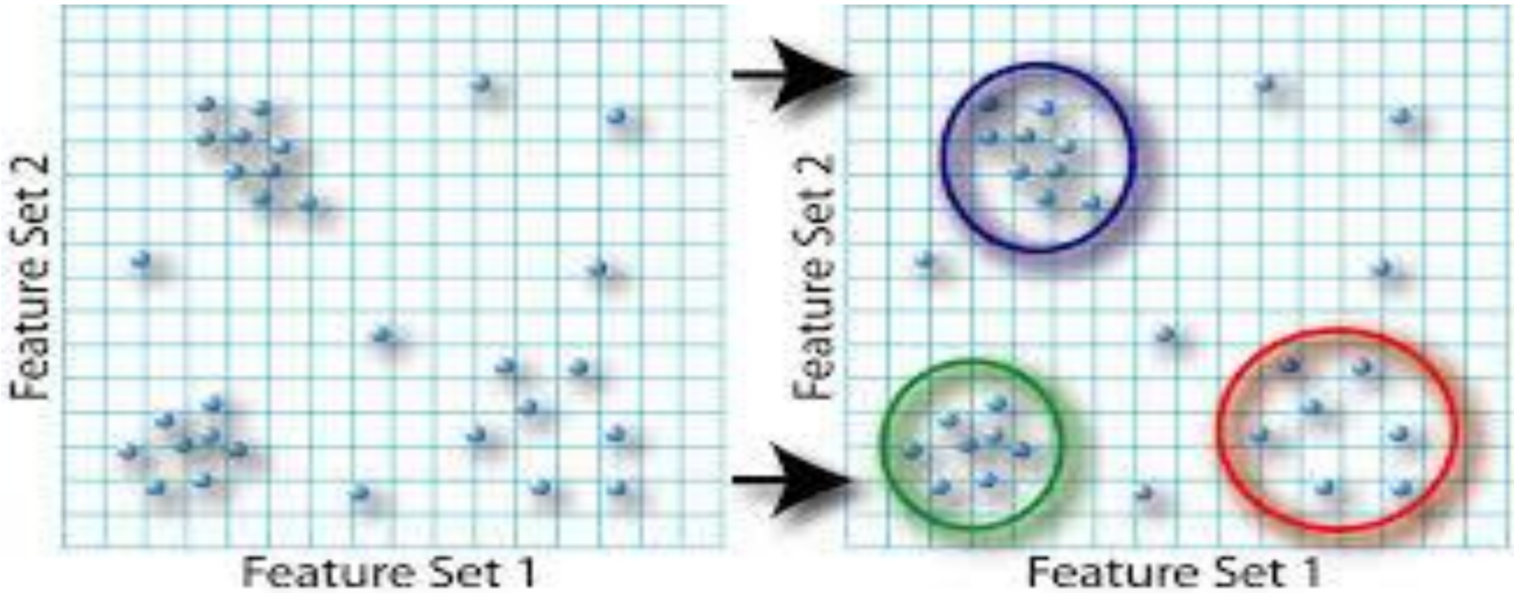
- Organizing data into classes such that there is
  - high intra-class similarity
  - low inter-class similarity
- Finding the class labels and the number of classes directly from the data (in contrast to classification).
- More informally, finding natural groupings among objects.
- Also called unsupervised learning, sometimes called classification by statisticians and sorting by psychologists and segmentation by people in marketing

Definition

- Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups



Clustering Example



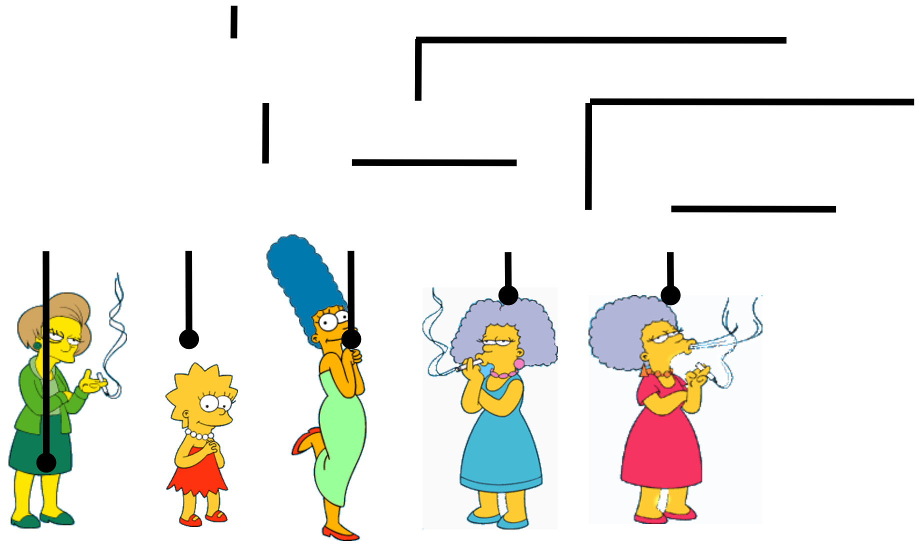
Types of Clustering

- Hierarchical clustering(BIRCH)
  - A set of nested clusters organized as a hierarchical tree
- Partitional Clustering(k-means,k-mediods)
  - A division data objects into non-overlapping (distinct) subsets (i.e., clusters) such that each data object is in exactly one subset
- Density – Based(DBSCAN)
  - Based on density functions
- Grid-Based(STING)
  - Based on nultiple-level granularity structure
- Model-Based(SOM)
  - Hypothesize a model for each of the clusters and find the best fit of the data to the given model

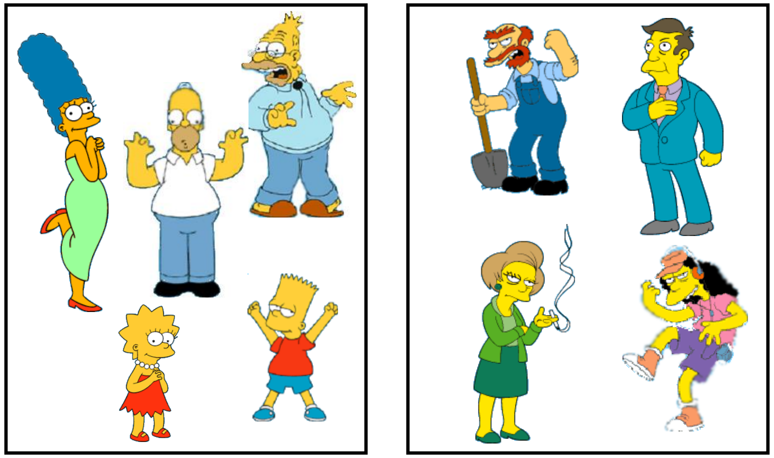
Two types of Clustering

- Partitional algorithms: Construct various partitions and then evaluate them by some criterion (we will see an example called BIRCH)
- Hierarchical algorithms: Create a hierarchical decomposition of the set of objects using some criterion

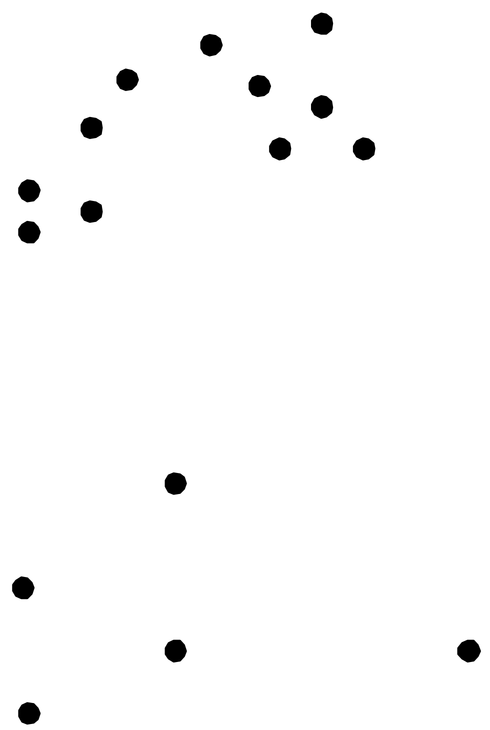
Hierarchical



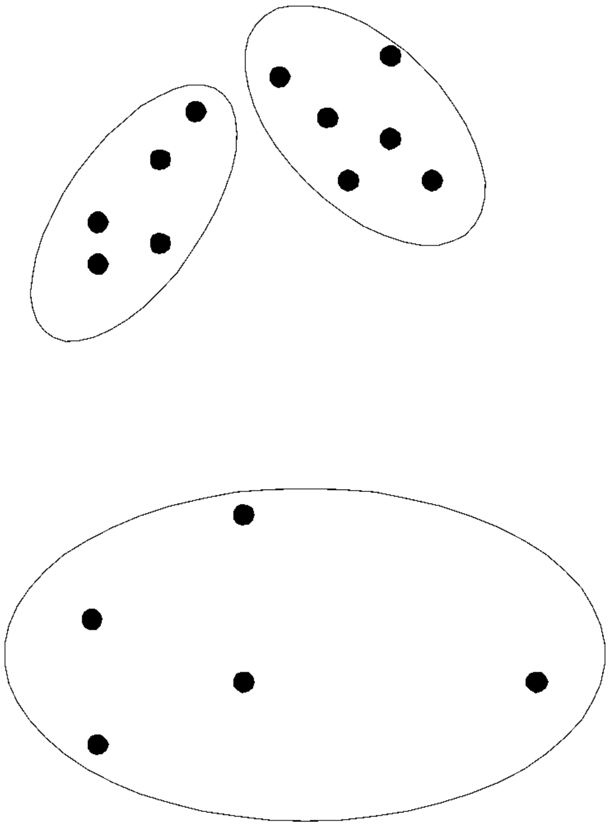
Partitional



Partitional Clustering



Original Points



A Partitional Clustering

Clustering Algorithms

- Partitional
  - K-means

Desirable Properties of Clustering

- Scalability (in terms of both time and space)
- Ability to deal with different data types
- Minimal requirements for domain knowledge to determine input parameters
- Able to deal with noise and outliers
- Insensitive to order of input records
- Incorporation of user-specified constraints
- Interpretability and usability

K-MEANS CLUSTERING

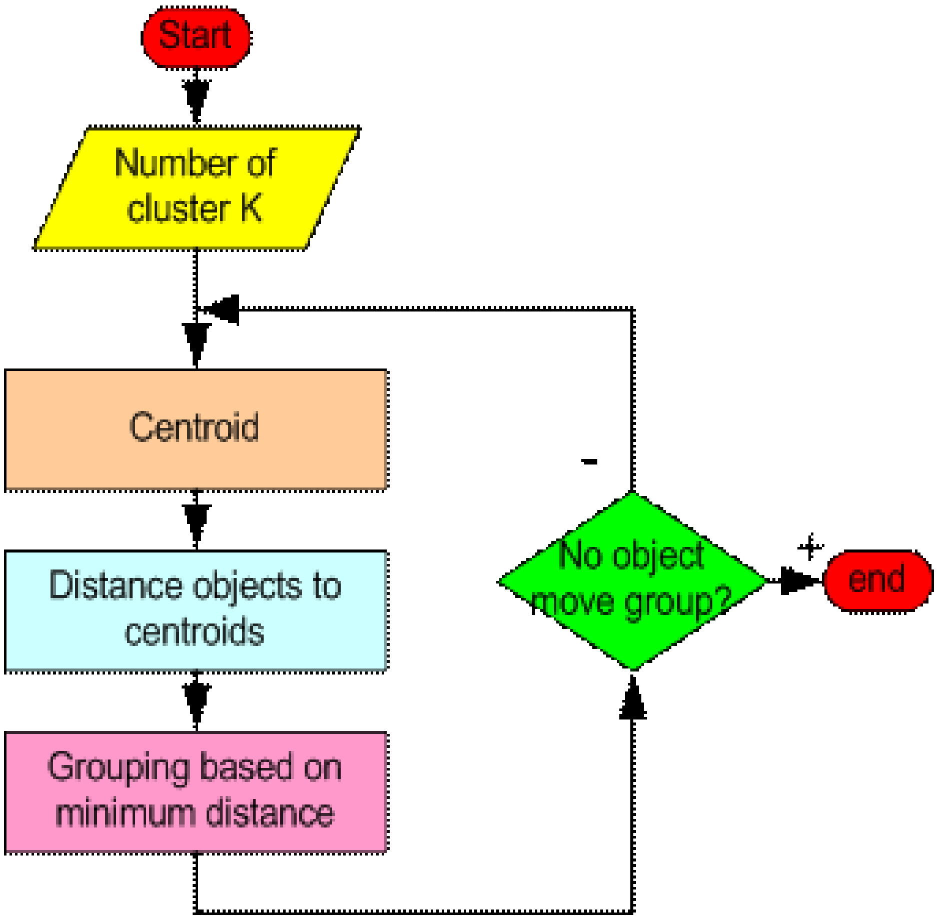
- The **k-means algorithm** is an algorithm to cluster *n* objects based on attributes into *k* partitions, where $k < n$.
- It assumes that the object attributes form a vector space.
- An algorithm for partitioning (or clustering) N data points into K disjoint subsets S_j containing data points so as to minimize the sum-of-squares criterion
- Where x_n is a vector representing the the n^{th} data point and u_j is the geometric centroid of the data points in S_j.

$$J = \sum_{j=1}^{K} \sum_{n \in S_j} \left| x_n - \mu_j \right|^2$$

K-MEANS CLUSTERING

- Simply speaking k-means clustering is an algorithm to classify or to group the objects based on attributes/features into K number of group.
- *K* is positive integer number.
- The grouping is done by minimizing the sum of squares of distances between data and the corresponding cluster centroid.

Working of K-MEANS CLUSTERING



Classical Partitioning Method- K mean

- First, it randomly selects K of the objects, each of which initially means represents cluster mean or center.
- For each of the remaining objects, an object is assigned to the cluster to which it is most similar, based on the distance between the object and the cluster mean.
- It then computes the new mean for each cluster.
- This process iterates until the criterion function converges.
- Typically, the Square error criterion is used, defines as

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} dist(p, c_i)^2,$$

- ***E***- Sum c r all objects in the data set
- ***P*** is the p presenting given object;
- ***mi-*** is the mean of cluster Ci

Classical Partitioning Method- K mean

- In other words, for each object in each cluster , the distance from the object to its cluster center is squared, and the distances are summed.
- This criterion tries to make the resulting k clusters as compact and as separate as possible.

Working of K-MEANS CLUSTERING

- Begin with a decision on the value of k = number of clusters
- Arbitrarily assign k objects from D as the initial cluster centers
- Each object is distributed to a cluster based on the cluster center to which it is the nearest.
- Next, the cluster centers are updated i.e. mean valu of each cluster is recalculated based on the current objects in the cluster
- Using the new cluster centers, the objects are redistributed to the clusters based on which cluster center is the nearest.
- This process iterates
- Eventually, no redistribution of the objects in any occurs , and so the process terminates
- Resulting clusters are returned by the clustering process.

Algorithm of K-MEANS CLUSTERING

Algorithm: *k*-means. The *k*-means algorithm for partitioning, where each cluster's center is represented by the mean value of the objects in the cluster.

Input:

- *k*: the number of clusters,
- *D*: a data set containing *n* objects.

Output: A set of *k* clusters.

Method:

(1) arbitrarily choose *k* objects from *D* as the initial cluster centers;
(2) **repeat**
(3) (re)assign each object to the cluster to which the object is the most similar, based on the mean value of the objects in the cluster;
(4) update the cluster means, that is, calculate the mean value of the objects for each cluster;
(5) **until** no change;

Example 1

Given: {2,3,6,8,9,12,15,18,22} Assume k=3.

- Solution:
  - Randomly partition given data set:
    - K1 = 2,8,15 mean = 8.3
    - K2 = 3,9,18 mean = 10
    - K3 = 6,12,22 mean = 13.3
  - Reassign
    - K1 = 2,3,6,8,9 mean = 5.6
    - K2 = mean = 0
    - K3 = 12,15,18,22 mean = 16.75

MIT-WPU
Dr. Vishwanath Karad MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

Example 1

- Reassign
  - K1 = 3,6,8,9 mean = 6.5
  - K2 = 2 mean = 2
  - K3 = 12,15,18,22 mean = 16.75
- Reassign
  - K1 = 6,8,9 mean = 7.6
  - K2 = 2,3 mean = 2.5
  - K3 = 12,15,18,22 mean = 16.75
- Reassign
  - K1 = 6,8,9 mean = 7.6
  - K2 = 2,3 mean = 2.5
  - K3 = 12,15,18,22 mean = 16.75
- STOP

Example 1

Given {2,4,10,12,3,20,30,11,25}
Assume k=2.

Solution:
K1 = 2,3,4,10,11,12
K2 = 20, 25, 30

A Simple example

A Simple example showing the implementation of k-means algorithm (using K=2)

Individual	Variable 1	Variable 2
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

A Simple example

Step 1:

Initialization: Randomly we choose following two centroids (k=2) for two clusters.

In this case the 2 centroid are: m1=(1.0,1.0) and m2=(5.0,7.0).

Individual	Variable 1	Variable 2
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

	Individual	Mean Vector
Group 1	1	(1.0, 1.0)
Group 2	4	(5.0, 7.0)

A Simple example

Step 2:

Thus, we obtain two clusters containing:
{1,2,3} and {4,5,6,7}.

Their new centroids are:

$$m_1 = \left(\frac{1}{3}(1.0+1.5+3.0), \frac{1}{3}(1.0+2.0+4.0)\right) = (1.83, 2.33)$$

$$m_2 = \left(\frac{1}{4}(5.0+3.5+4.5+3.5), \frac{1}{4}(7.0+5.0+5.0+4.5)\right)$$

$$= (4.12, 5.38)$$

Individual	Centroid 1	Centroid 2
1	0	7.21
2 (1.5, 2.0)	1.12	6.10
3	3.61	3.61
4	7.21	0
5	4.72	2.5
6	5.31	2.06
7	4.30	2.92

$$d(m_1, 2) = \sqrt{|1.0-1.5|^2 + |1.0-2.0|^2} = 1.12$$

$$d(m_2, 2) = \sqrt{|5.0-1.5|^2 + |7.0-2.0|^2} = 6.10$$

A Simple example

Step 3:

Now using these centroids we compute the Euclidean distance of each object, as shown in table.

Therefore, the new clusters are: {1,2} and {3,4,5,6,7}

Next centroids are:

m1=(1.25,1.5) and m2 = (3.9,5.1)

Individual	Centroid 1	Centroid 2
1	1.57	5.38
2	0.47	4.28
3	2.04	1.78
4	5.64	1.84
5	3.15	0.73
6	3.78	0.54
7	2.74	1.08

A Simple example

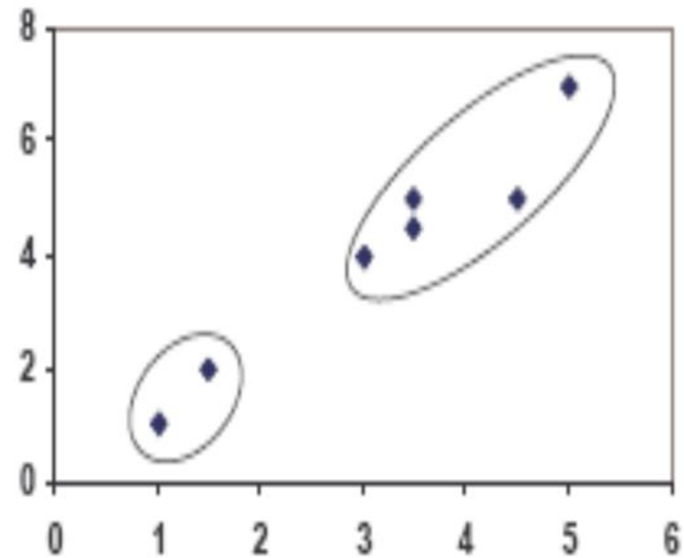
Step 4:

The clusters obtained are:
{1,2} and {3,4,5,6,7}

Therefore, there is no change in the cluster.

Thus, the algorithm comes to a halt here and final result consist of 2 clusters {1,2} and {3,4,5,6,7}.

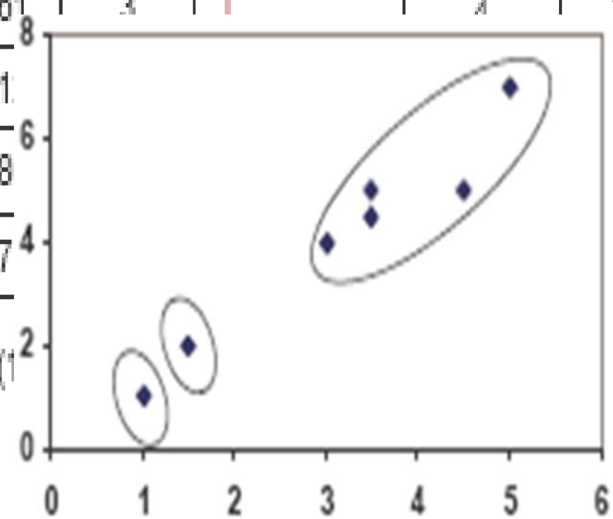
Individual	Centroid 1	Centroid 2
1	0.56	5.02
2	0.56	3.92
3	3.05	1.42
4	6.66	2.20
5	4.16	0.41
6	4.78	0.61
7	3.75	0.72



K=3

Individual	$m_1 = 1$	$m_2 = 2$	$m_3 = 3$	cluster
1	0	1.11	3.61	1
2	1.12	0	2.5	2
3	3.61	2.5	0	3
4	7.21	6.10	3.61	3
5	4.72	3.61	1.1	
6	5.31	4.24	1.8	
7	4.30	3.20	0.7	

clustering with initial centroids (1



Individual	m_1 (1.0, 1.0)	m_2 (1.5, 2.0)	m_3 (3.9,5.1)	cluster
1	0	1.11	5.02	1
2	1.12	0	3.92	2
3	3.61	2.5	1.42	3
	7.21	6.10	2.20	3
	4.72	3.61	0.41	3
	5.31	4.24	0.61	3
	4.30	3.20	0.72	3

Step 1

Step 2

Applications of K-Mean Clustering

- It is relatively efficient and fast. It computes result at O(tkn), where n is number of objects or points, k is number of clusters and t is number of iterations.
- k-means clustering can be applied to machine learning or data mining
- Used on acoustic data in speech understanding to convert waveforms into one of k categories (known as Vector Quantization or Image Segmentation).
- Also used for choosing color palettes on old fashioned graphical display devices and Image Quantization.

Weaknesses of K-Mean Clustering

- When the numbers of data are not so many, initial grouping will determine the cluster significantly.
- The number of cluster, K, must be determined before hand. Its disadvantage is that it does not yield the same result with each run, since the resulting clusters depend on the initial random assignments.
- We never know the real cluster, using the same data, because if it is inputted in a different order it may produce different cluster if the number of data is few.
- It is sensitive to initial condition. Different initial condition may produce different result of cluster. The algorithm may be trapped in the local optimum.

Advantages & Disadvantage of K means-Clustering

- Advantages
  - K-means is relatively scalable and efficient in processing large data sets
  - The computational complexity of the algorithm is O(nkt)
    - n: the total number of objects
    - k: the number of clusters
    - t: the number of iterations
    - Normally: k<<n and t<<n
- Disadvantage
  - Can be applied only when the mean of a cluster is defined
  - Users need to specify k
  - K-means is not suitable for discovering clusters with non convex
  - Shapes or clusters of very different size
  - It is sensitive to noise and outlier data points

Dr. Vishwanath Karad MIT WORLD PEACE UNIVERSITY | PUNE

Conclusion of Clustering

- K-means algorithm is useful for undirected knowledge discovery and is relatively simple. K-means has found wide spread usage in lot of fields, ranging from unsupervised learning of neural network, Pattern recognitions, Classification analysis, Artificial intelligence, image processing, machine vision, and many others.

References

- *J. A. Hartigan (1975) "Clustering Algorithms". Wiley.*
- *J. A. Hartigan and M. A. Wong (1979) "A K-Means Clustering Algorithm", Applied Statistics, Vol. 28, No. 1, p100-108.*
- *www.wikipedia.com*