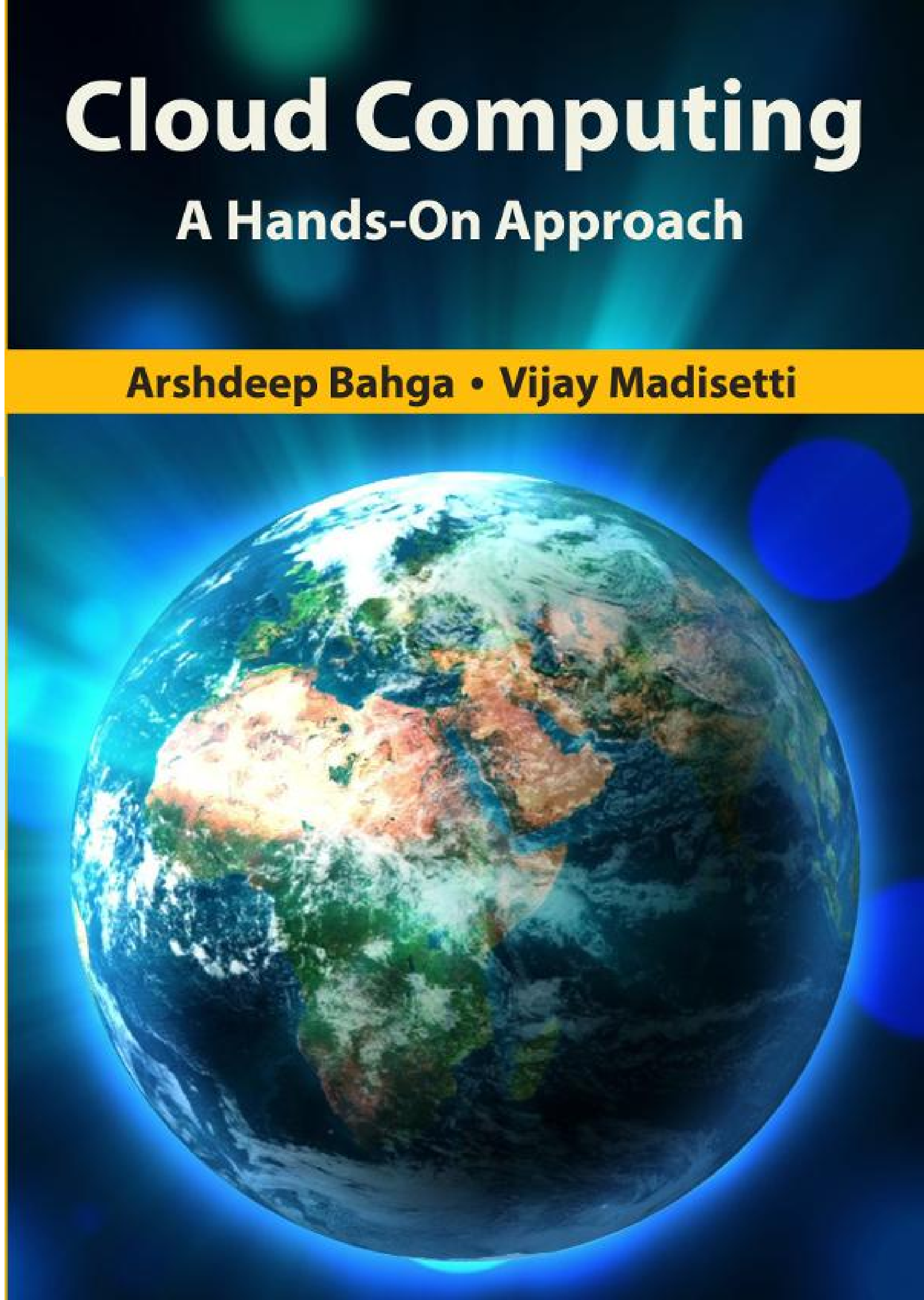
Chapter 9

Big Data Analytics



Outline

- Big Data analytics approaches
- Approaches for clustering big data
- Approaches for classification of big data
- Recommendation Systems

Big Data

- Big data is defined as collections of data sets whose volume, velocity in terms of time variation, or variety is so large that it is difficult to store, manage, process and analyze the data using traditional databases and data processing tools.
- Characteristics of big data:
  - Volume
    - Though there is no fixed threshold for the volume of data to be considered as big data, however, typically, the term big data is used for massive scale data that is difficult to store, manage and process using traditional databases and data processing architectures. The volumes of data generated by modern IT, industrial, healthcare and systems is growing exponentially driven by the lowering costs of data storage and processing architectures and the need to extract valuable insights from the data to improve business processes, efficiency and service to consumers.
  - Velocity
    - Velocity is another important characteristic of big data and the primary reason for exponential growth of data. Velocity of data refers to how fast the data is generated. Modern IT, industrial and other systems are generating data at increasingly higher speeds generating big data.
  - Variety
    - Variety refers to the forms of the data. Big data comes in different forms such as structured or unstructured data, including text data, image, audio, video and sensor data.

Clustering Big Data

- Clustering is the process of grouping similar data items together such that data items that are more similar to each other (with respect to some similarity criteria) than other data items are put in one cluster.
- Clustering big data is of much interest, and happens in applications such as:
  - Clustering social network data to find a group of similar users
  - Clustering electronic health record (EHR) data to find similar patients.
  - Clustering sensor data to group similar or related faults in a machine
  - Clustering market research data to group similar customers
  - Clustering clickstream data to group similar users
- Clustering is achieved by clustering algorithms that belong to a broad category algorithms called unsupervised machine learning.
- Unsupervised machine learning algorithms find the patterns and hidden structure in data for which no training data is available.

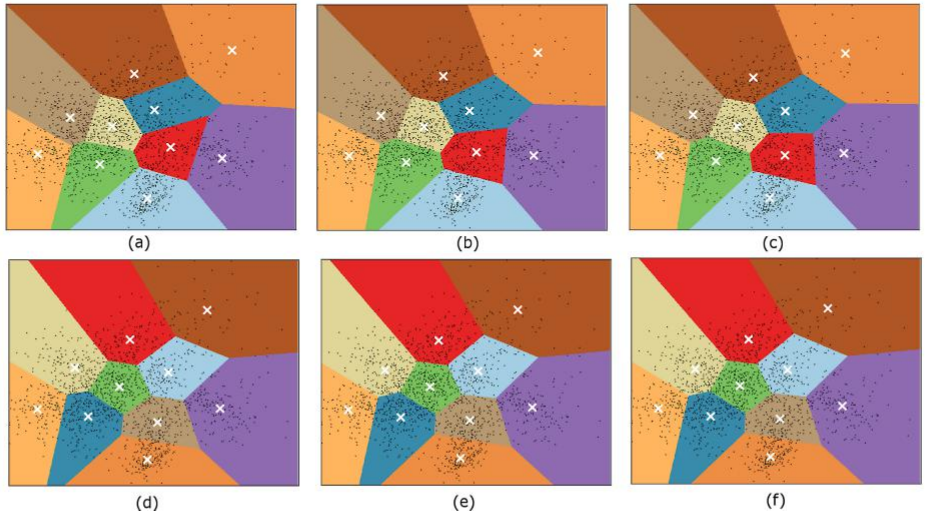
k-means Clustering

- k-means is a clustering algorithm that groups data items into k clusters, where k is user defined.
- Each cluster is defined by a centroid point.
- k-means clustering begins with a set of k centroid points which are either randomly chosen from the dataset or chosen using some initialization algorithm such as canopy clustering.
- The algorithm proceeds by finding the distance between each data point in the data set and the centroid points.
- Based on the distance measure, each data point is assigned to a cluster belonging to the closest centroid.
- In the next step the centroids are recomputed by taking the mean value of all the data points in a cluster.
- This process is repeated till the centroids no longer move more than a specified threshold.

k-means Clustering

k-means Clustering Algorithm

```
Start with k centroid points
while the centroids no longer move beyond a threshold or maximum number of iterations reached:
    for each point in the dataset:
        for each centroid:
            find the distance between the point and the centroid
            assign the point to the cluster belonging to the nearest centroid
        for each cluster:
            recompute the centroid point by taking mean value of all points in the cluster
```



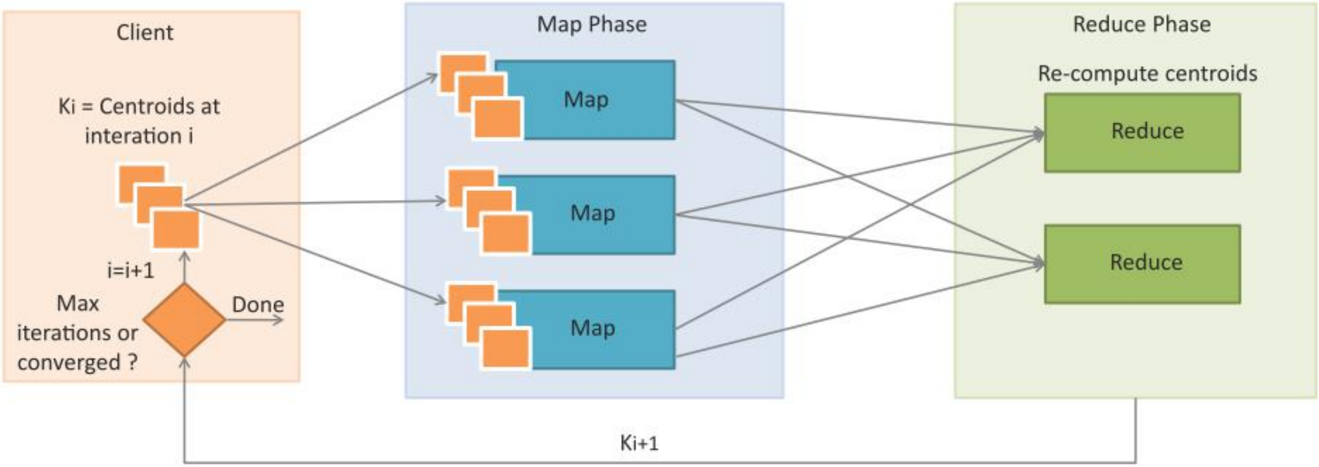
Example of clustering 300 points with k-means: (a) iteration 1, (b) iteration 2, (c) iteration 3, (d) iteration 5, (e) iteration 10, (f) iteration 100.

Clustering Documents with k-means

- Document clustering is the most commonly used application of k-means clustering algorithm.
- Document clustering problem occurs in many big data applications such as finding similar news articles, finding similar patients using electronic health records, etc.
- Before applying k-means algorithm for document clustering, the documents need to be vectorized. Since documents contain textual information, the process of vectorization is required for clustering documents.
- The process of generating document vectors involves several steps:
  - A dictionary of all words used in the tokenized records is generated. Each word in the dictionary has a dimension number assigned to it which is used to represent the dimension the word occupies in the document vector.
  - The number of occurrences or term frequency (TF) of each word is computed.
  - Inverse Document Frequency (IDF) for each word is computed. Document Frequency (DF) for a word is the number of documents (or records) in which the word occurs.
  - Weight for each word is computed. The term weight Wi is used in the document vector as the value for the dimension-i.
  - Similarity between documents is computed using a distance measure such as Euclidean distance measure.

k-means with MapReduce

- The data to be clustered is distributed on a distributed file system such as HDFS and split into blocks which are replicated across different nodes in the cluster.
- Clustering begins with an initial set of centroids. The client program controls the clustering process.
- In the Map phase, the distances between the data samples and centroids are calculated and each sample is assigned to the nearest centroid.
- In the Reduce phase, the centroids are recomputed using the mean of all the points in each cluster.
- The new centroids are then fed back to the client which checks whether convergence is reached or maximum number of iterations are completed.

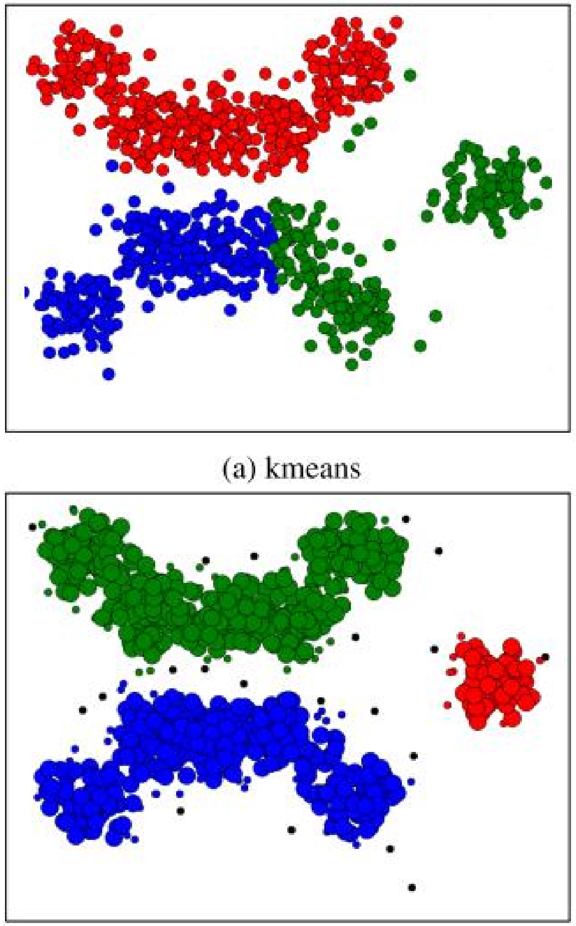


DBSCAN clustering

- DBSCAN is a density clustering algorithm that works on the notions of density reachability and density connectivity.
- Density Reachability
  - Is defined on the basis of *Eps*-neighborhood, where *Eps*-neighborhood means that for every point *p* in a cluster *C* there is a point *q* in *C* so that *p* is inside of the *Eps*-neighborhood of *q* and there are at least a minimum number (*MinPts*) of points in an *Eps*-neighborhood of that point.
  - A point *p* is called directly density-reachable from a point *q* if it is not farther away than a given distance (*Eps*) and if it is surrounded by at least a minimum number (*MinPts*) of points that may be considered to be part of a cluster.
- Density Connectivity
  - A point *p* is density connected to a point *q* if there is a point *o* such that both, *p* and *q* are density-reachable from *o* wrt. *Eps* and *MinPts*.
- A cluster, is then defined based on the following two properties:
  - Maximality: For all point *p, q* if *p* belongs to cluster *C* and *q* is density-reachable from *p* (wrt. *Eps* and *MinPts*), then *q* also belongs to the cluster *C*.
  - Connectivity: For all point *p, q* in cluster *C, p* is density-connected to *q* (wrt. *Eps* and *MinPts*).

DBSCAN vs K-means

- DBSCAN can find irregular shaped clusters as seen from this example and can even find a cluster completely surrounded by a different cluster.
- DBSCAN considers some points as noise and does not assign them to any cluster.



(a) kmeans

(b) DBSCAN

Classification of Big Data

- Classification is the process of categorizing objects into predefined categories.
- Classification is achieved by classification algorithms that belong to a broad category of algorithms called supervised machine learning.
- Supervised learning involves inferring a model from a set of input data and known responses to the data (training data) and then using the inferred model to predict responses to new data.
- Binary classification
  - Binary classification involves categorizing the data into two categories. For example, classifying the sentiment of a news article into positive or negative, classifying the state of a machine into good or faulty, classifying the heath test into positive or negative, etc.
- Multi-class classification
  - Multi-class classification involves more than two classes into which the data is categorized. For example, gene expression classification problem involves multiple classes.
- Document classification
  - Document classification is a type of multi-class classification approach in which the data to the classified is in the form of text document. For classifying news articles into different categories such as politics, sports, etc.

Performance of Classification Algorithms

- Precision: Precision is the fraction of objects that are classified correctly.

$$Precision = \frac{TruePositive}{(TruePositive + FalsePositive)}$$

- Recall: Recall is the fraction of objects belonging to a category that are classified correctly.

$$Recall = \frac{TruePositive}{(TruePositive + FalseNegative)}$$

- Accuracy:

$$Accuracy = \frac{(TruePositive + TrueNegative)}{(TruePositive + TrueNegative + FalsePositive + FalseNegative)}$$

- F1-score: F1-score is a measure of accuracy that considers both precision and recall. F1-score is the harmonic means of precision and recall given as,

$$F1 - Score = \frac{2(Precision)(Recall)}{(Precision + Recall)}$$

Naive Bayes

- Naive Bayes is a probabilistic classification algorithm based on the Bayes theorem with a naive assumption about the independence of feature attributes. Given a class variable C and feature variables F1,...,Fn , the conditional probability (posterior) according to Bayes theorem is given as,

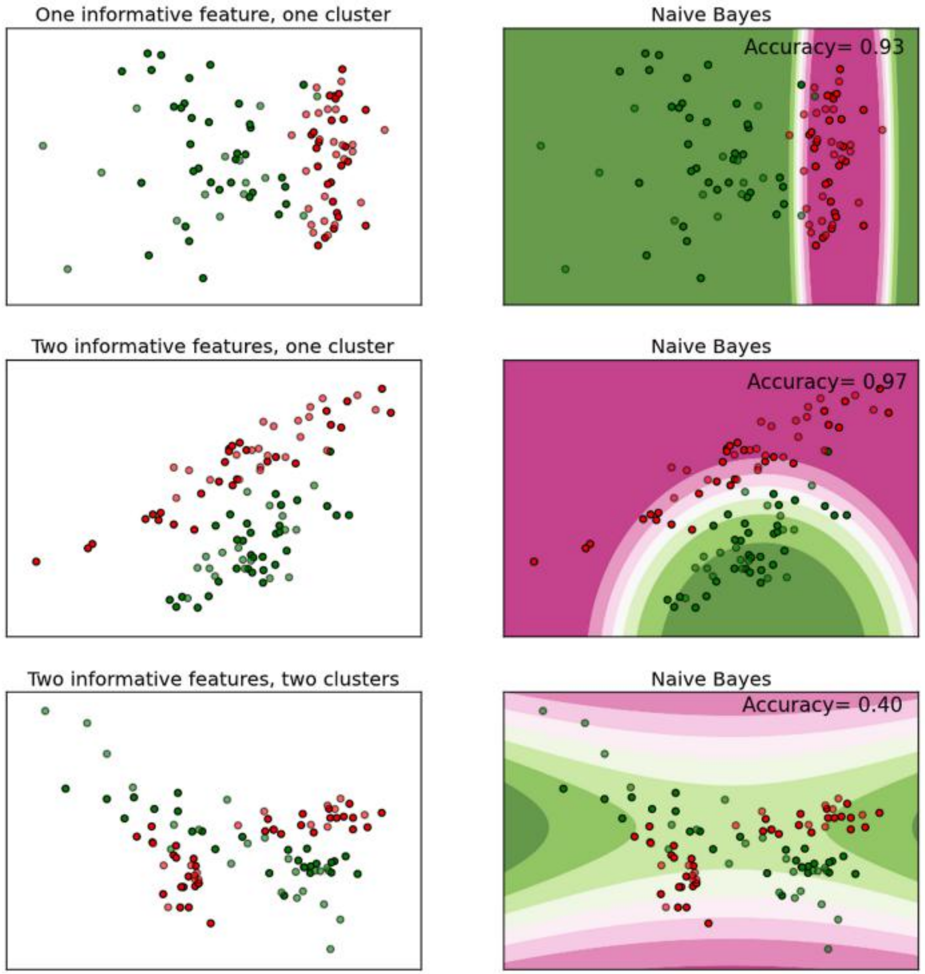
$$P(C|F_1,...,F_n) = \frac{P(F_1,...,F_n|C)P(C)}{P(F_1,...,F_n)}$$

- where, P(C|F1,...,Fn) is the posterior probability, P(F1,...,Fn |C) is the likelihood and P(C) is the prior probability and P(F1,...,Fn) is the evidence. Naive Bayes makes a naïve assumption about the independence every pair of features given as,

$$P(F_1,...,F_n|C) = \prod_{i=1}^{n} P(F_i|C)$$

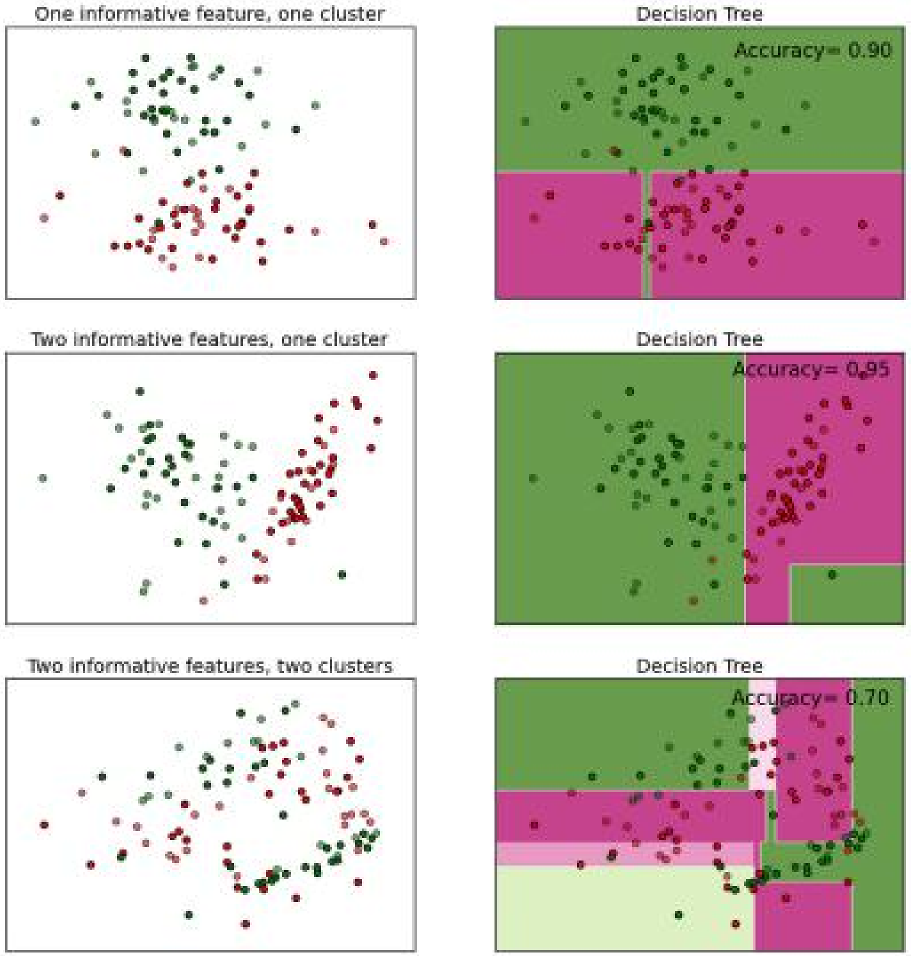
- Since the evidence P(F1,...,Fn) is constant for a given input and does not depend on the class variable C, only the numerator of the posterior probability is important for classification.
- With this simplification, classification can then be done as follows,

$$C = argmax_C P(C) \prod_{i=1}^{n} P(F_i|C)$$



Decision Trees

- Decision Trees are a supervised learning method that use a tree created from simple decision rules learned from the training data as a predictive model.
- The predictive model is in the form of a tree that can be used to predict the value of a target variable based on a several attribute variables.
- Each node in the tree corresponds to one attribute in the dataset on which the “split” is performed.
- Each leaf in a decision tree represents a value of the target variable.
- The learning process involves recursively splitting on the attributes until all the samples in the child node have the same value of the target variable or splitting further results in no further information gain.
- To select the best attribute for splitting at each stage, different metrics can be used.



Splitting Attributes in Decision Trees

To select the best attribute for splitting at each stage, different metrics can be used such as:

- Information Gain
  - Information content of a discrete random variable X with probability mass function (PMF), P(X), is defined as,

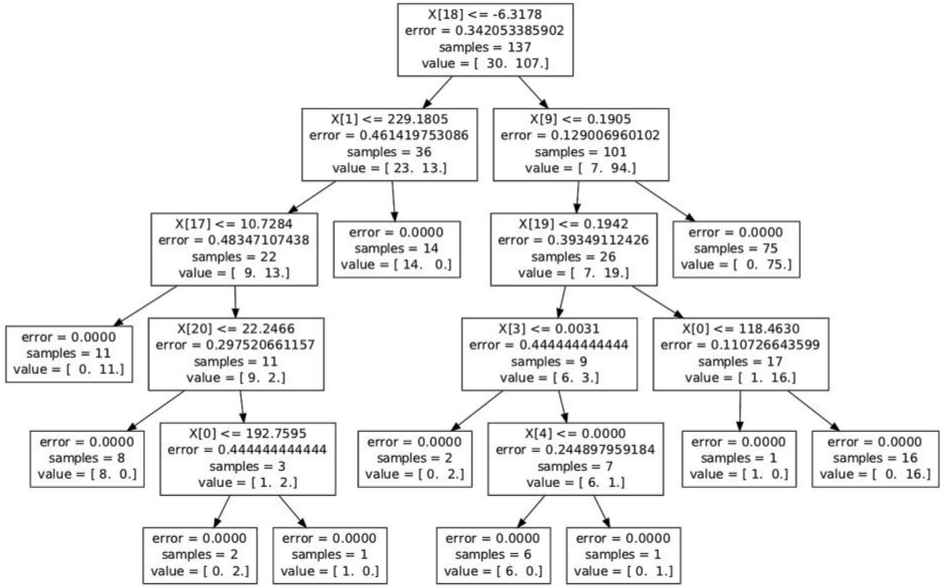
$$I(X) = -\log_2 P(X)$$

  - Information gain is defined based on the entropy of the random variable which is defined as,

$$H(X) = E[I(X)] = E[-\log_2 P(X)] = -\sum_i \log_2 P(x_i)$$

  - Entropy is a measure of uncertainty in a random variable and choosing the attribute with the highest information gain results in a split that reduces the uncertainty the most at that stage.
- Gini Coefficient
  - Gini coefficient measures the inequality, i.e. how often a randomly chosen sample that is labeled based on the distribution of labels, would be labeled incorrectly. Gini coefficient is defined as,

$$G(X) = 1 - \sum_i P(x_i)^2$$

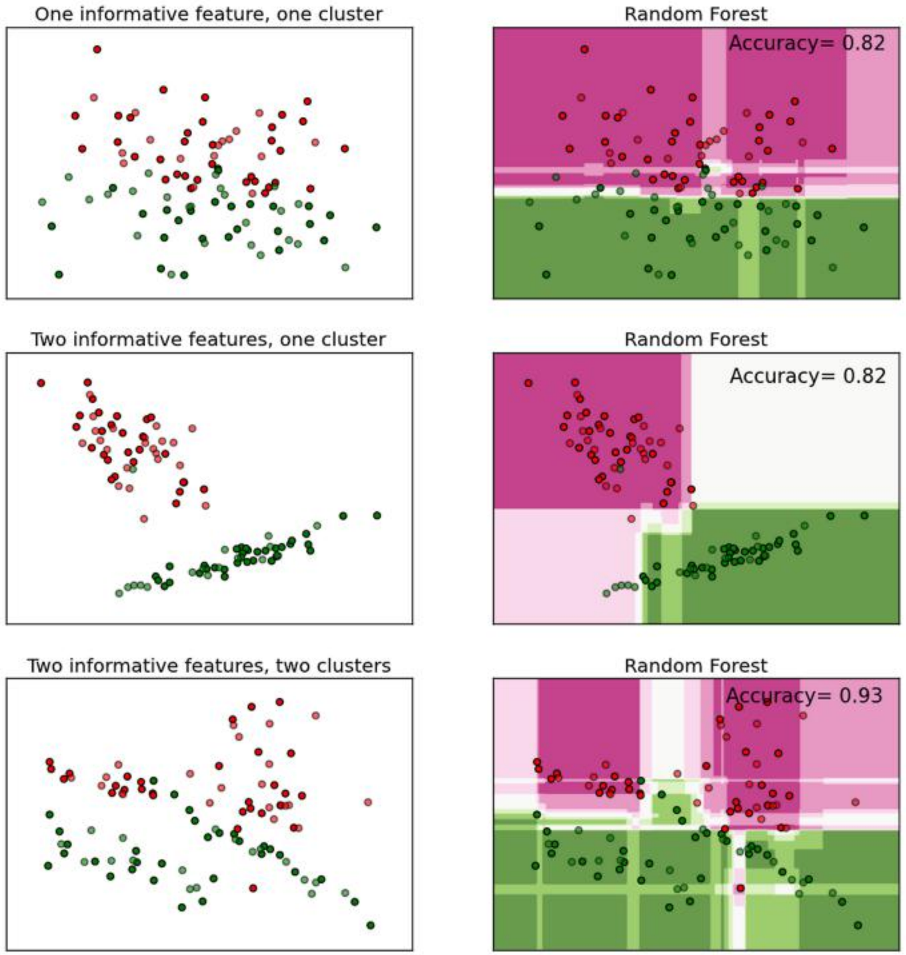


Decision Tree Algorithms

- There are different algorithms for building decisions trees, popular ones being ID3 and C4.5.
- ID3:
  - Attributes are discrete. If not, discretize the continuous attributes.
  - Calculate the entropy of every attribute using the dataset.
  - Choose the attribute with the highest information gain.
  - Create branches for each value of the selected attribute.
  - Repeat with the remaining attributes.
- The ID3 algorithm can be result in over-fitting to the training data and can be expensive to train especially for continuous attributes.
- C4.5
  - The C4.5 algorithm is an extension of the ID3 algorithm. C4.5 supports both discrete and continuous attributes.
  - To support continuous attributes, C4.5 finds thresholds for the continuous attributes and then splits based on the threshold values. C4.5 prevents over-fitting by pruning trees after they have been created.
  - Pruning involves removing or aggregating those branches which provide little discriminatory power.

Random Forest

- Random Forest is an ensemble learning method that is based on randomized decision trees.
- Random Forest trains a number decision trees and then takes the majority vote by using the mode of the class predicted by the individual trees.

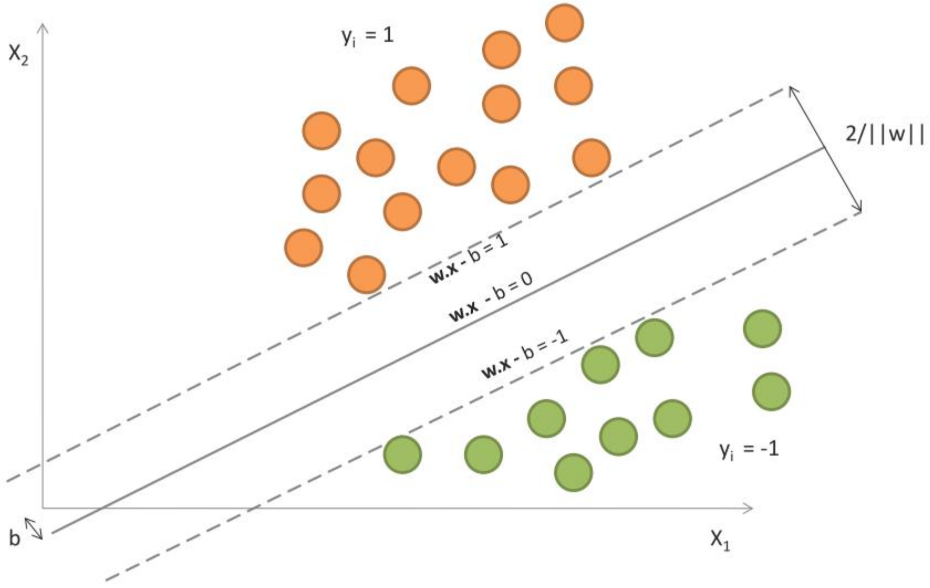


Breiman’s Algorithm

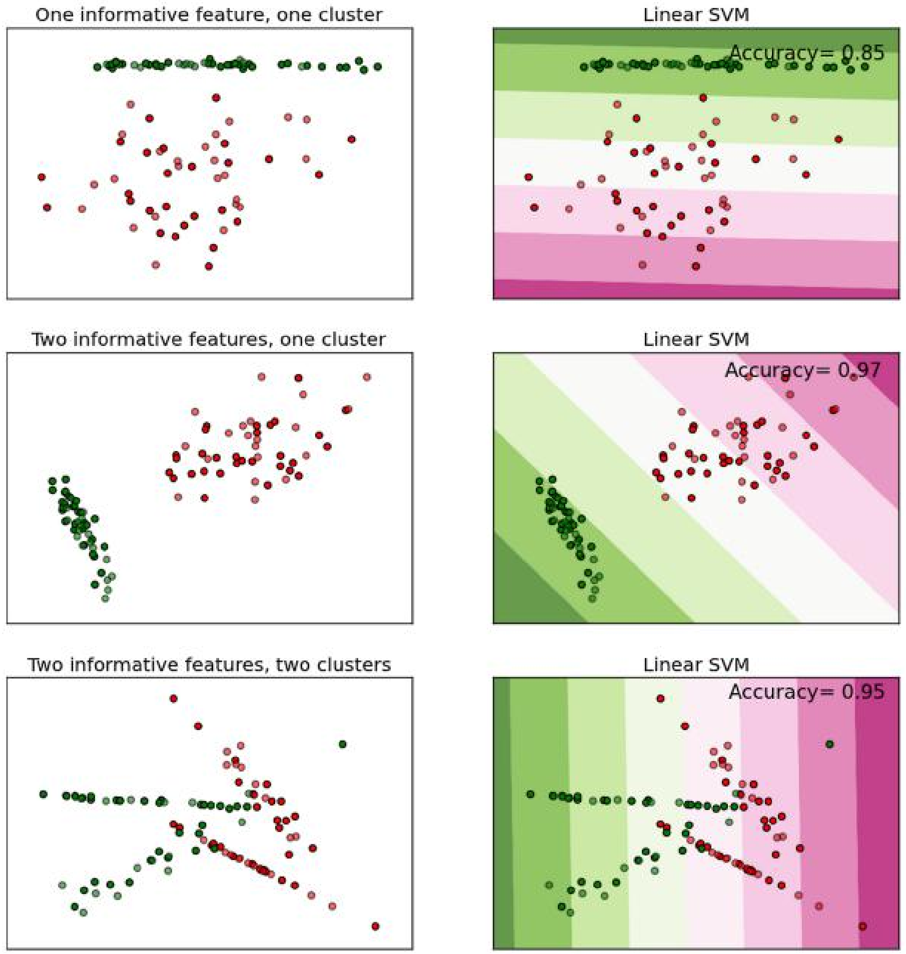
1. Draw a bootstrap sample (n times with replacement from the N samples in the training set) from the dataset
2. Train a decision tree
    - Until the tree is fully grown (maximum size)
    - Choose next leaf node
    - Select m attributes (m is much less than the total number of attributes M) at random.
    - Choose the best attribute and split as usual
3. Measure out-of-bag error
    - Use the rest of the samples (not selected in the bootstrap) to estimate the error of the tree, by predicting their classes.
4. Repeat steps 1-3 k times to generate k trees.
5. Make a prediction by majority vote among the k trees

Support Vector Machine

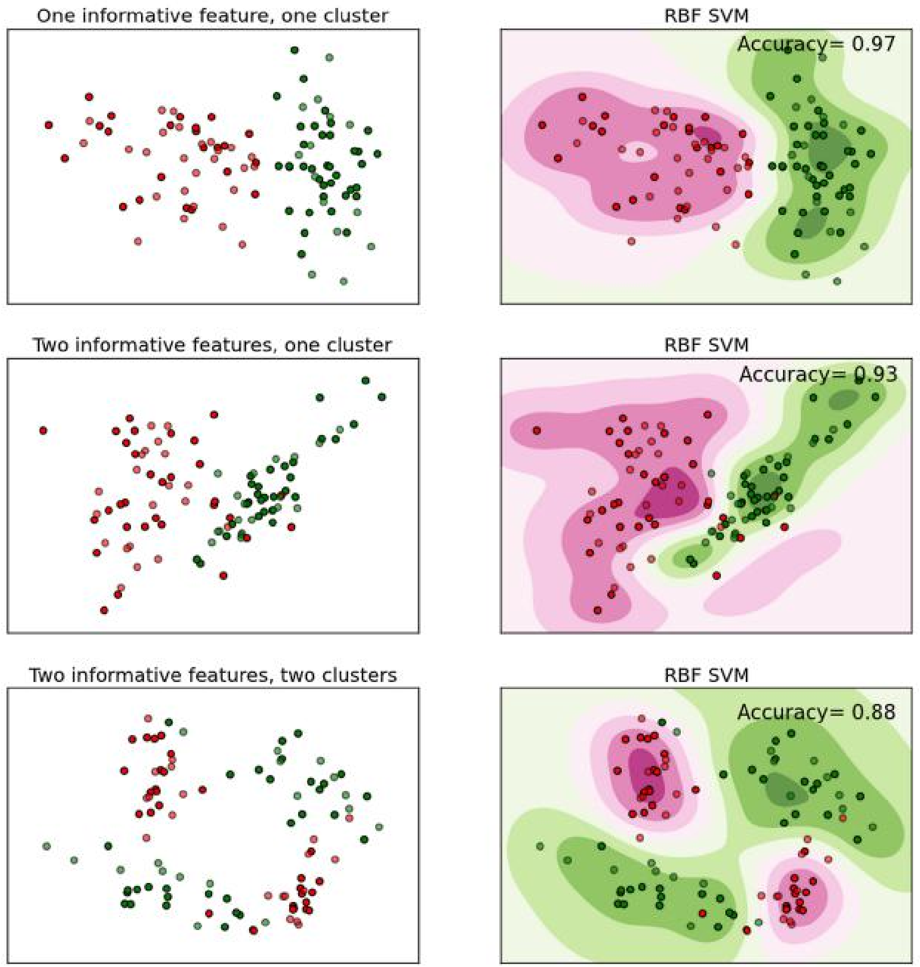
- Support Vector Machine (SVM) is a supervised machine learning approach used for classification and regression.
- The basic form is SVM is a binary classifier that classifies the data points into one of the two classes.
- SVM training involves determining the maximum margin hyperplane that separates the two classes.
- The maximum margin hyperplane is one which has the largest separation from the nearest training data point.
- Given a training data set (x_i ,y_i) where x_i is an n dimensional vector and y_i = 1 if x_i is in class 1 and y_i = -1 if x_i is in class 2.
- A standard SVM finds a hyperplane **w.x**-b = 0, which correctly separates the training data points and has a maximum margin which is the distance between the two hyperplanes **w.x**-b = 1 and **w.x**-b = -1



Support Vector Machine



Binary classification with Linear SVM



Binary classification with RBF SVM

Recommendation Systems

- Recommendation systems are an important part of modern cloud applications such as e-Commerce, social networks, content delivery networks, etc.
- Item-based or Content-based Recommendation
  - Provides recommendations to users (for items such as books, movies, songs, or restaurants) for unrated items based on the characteristics of the item.
- Collaborative Filtering
  - Provides recommendations based on the ratings given by the user and other users to similar items.

Further Reading

- Martin Ester, Hans-Peter Kriegel, Jorg Sander, Xiaowei Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), 1996.
- Scikit-learn, http://scikit-learn.org/stable
- Apache Mahout, http://mahout.apache.org
- Corinna Cortes, Vladimir N. Vapnik, "Support-Vector Networks", Machine Learning, 20, 1995.