A Study on Selective Data Mining Algorithms

[1] A.N.Pathak, [2] Manu Sehgal and [3] Divya Christopher

[1] Dr.A.N.Pathak, Professor & Head Applied Science, NITTTR
Chandigarh -160019, India

[2] Department of Information Technology, GGDSD College.
Chandigarh -160019, India

[3] Divya Christopher Senoir Lecturer, Amity university
Noida, India

Abstract

Data mining is a field of database application that searches for unknown patterns in data that can be used to predict future behavior. Basically data mining is a technique not to change the presentation but to discover unknown relationships between the data. Data mining is termed as software, which is used to describe data in a new way, which is not true.

Keywords: (k)-***means clustering, (****k*)-***nearest neighbor***

1. Introduction

Data mining, a branch of computer science and artificial intelligence is the process of extracting patterns from data. Data mining is seen as an increasingly important tool by modern business to transform data into bussiness intelligence giving an informational advantage. It is currently used in a wide range of profiling practices, such as marketing, surveillance, fraud detection, and scientific discovery.

1.1) Data mining commonly involves four classes of tasks

- **Clustering** - is the task of discovering groups and structures in the data that are in some way or another "similar", without using known structures in the data.

- **Classification**- is the task of generalizing known structure to apply to new data. For example, an email program might attempt to classify an email as legitimate or spam. Common algorithms include decision tree learning, nearest neighor, naïve Baysian classification, neural networks and support vector machines.

- **Regression** - Attempts to find a function which models the data with the least error. Regression analysis helps us understand how the typical value of the dependent variable changes when any one of the independent variables is varied, while the other independent variables are held fixed

- **Association rule learning** - Searches for relationships between variables. For example a supermarket might gather data on customer purchasing habits. Using association rule learning, the supermarket can determine which products are frequently bought together and use this information for marketing purposes. This is sometimes referred to as market basket analysis..

2. Cluster analysis or **clustering** is the assignment of a set of observations into subsets (called *clusters*) so that observations in the same cluster are similar in some sense. Clustering is a method of unsupervised learning, and a common technique for statistical data analysis used in many fields, including machine learning, data mining,pattern recognition, image analysis,information retrieval, and bioinformatics.

2.1) Types of clustering algorithms

A) Hierarchical algorithms find successive clusters using previously established clusters. These algorithms usually are either agglomerative ("bottom-up") or divisive ("top-

down"). Agglomerative algorithms begin with each element as a separate cluster and merge them into successively larger clusters. Divisive algorithms begin with the whole set and proceed to divide it into successively smaller clusters.

B) Partitional algorithms typically determine all clusters at once, but can also be used as divisive algorithms in the *hierarchical clustering*.

C) Density-based clustering algorithms are devised to discover arbitrary-shaped clusters. In this approach, a cluster is regarded as a region in which the density of data objects exceeds a threshold. *DBSCAN* and *OPTICS* are two typical algorithms of this kind.

Subspace clustering methods look for clusters that can only be seen in a particular projection (subspace, manifold) of the data. These methods thus can ignore irrelevant attributes. The general problem is also known as *Correlation clustering* while the special case of axis-parallel subspaces is also known as *Two-way clustering*, *co-clustering* or *biclustering* in these methods not only the objects are clustered but also the features of the objects, i.e., if the data is represented in a *data matrix*, the rows and columns are clustered simultaneously. They usually do not however work with arbitrary feature combinations as in general subspace methods. But this special case deserves attention due to its applications in bioinformatics.

Many clustering algorithms require the specification of number of clusters to produce in the input data set, prior to execution of the algorithm. Barring knowledge of the proper value beforehand, the appropriate value must be determined, a problem on its own for which a number of techniques have been developed.

2.2) Partitioned clustering Algorithm (PCA)

***k*-means clustering**

The *k*-means algorithm assigns each point to the cluster whose center (also called centroid) is nearest. The center is the average of all the points in the cluster — that is, its coordinates are the arithmetic mean for each dimension separately over all the points in the cluster.

Example: The data set has three dimensions and the cluster has two points: $X = (x_1,x_2,x_3)$ and $Y = (y_1,y_2,y_3)$. Then the centroid Z becomes $Z = (z_1, z_2, z_3)$, where $z_1 = \frac{x_1 + y_1}{2}$, $z_2 = \frac{x_2 + y_2}{2}$ and $z_3 = \frac{x_3 + y_3}{2}$.

The algorithm steps are:

- *Choose the number of clusters*, *k*.
- Randomly generate *k* clusters and determine the cluster centers, or directly generate *k* random points as cluster centers.
- Assign each point to the nearest cluster center, where "nearest" is defined with respect to one of the distance measures discussed above.
- Recompute the new cluster centers.

Repeat the two previous steps until some convergence criterion is met (usually that the assignment hasn't changed).

The main advantages of this algorithm are its simplicity and speed which allows it to run on large datasets. Its disadvantage is that it does not yield the same result with each run, since the resulting clusters depend on the initial random assignments (the *k-means ++* algorithm addresses this problem by seeking to choose better starting clusters). It minimizes intra-cluster variance, but does not ensure that the result has a global minimum of variance. Another disadvantage is the requirement for the concept of a mean to be definable which is not always the case. For such datasets the *k-medoids* variants is appropriate. An alternative, using a different criterion for which points are best assigned to which centre is *K-medians clustering*.

3) Statistical classification is the problem in statistics of identifying the sub-population to which new observations belong, where the identify of the sub-population is unknown, on the basis of a training set of data containing observations whose sub-population is known. Thus the requirement is that new individual items are placed into groups based on quantitative information on one or more measurements, traits or characteristics, etc) and based on the training set in which previously decided groupings are already established.

The problem here may be contrasted with that for cluster *cluster analysis*, where the problem is to

analyse a single data-set and decide how and whether the observations in the data-set can be divided into groups. In certain terminology, particularly that of machine learning, the classification problem is known as supervised learning, while clustering is known as *unsupervised learning*.

3.1) Problem statement

The data-set used for training consists of information *x* and *y* for each data-point, where *x* denotes what is generally a vector of observed characteristics for the data-item and *y* denotes a group-label. The label *y* can take only a finite number of values. The classification problem can be stated as follows: given training data $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ produce a rule (or "classifier") *h*, such that $h(x)$ can be evaluated for any possible value of *x* (not just those included in the training data) and such that the group attributed to any new observation, specifically

$$\hat{y} = h(x),$$

is as close as possible to the true group label *y*. For the training data-set, the true labels y_i are known but will not necessarily match their in-sample approximations

$$\hat{y}_i = h(x_i).$$

For new observations, the true labels y_j are unknown, but it is a prime target for the classification procedure that the approximation

$$\hat{y}_j = h(x_j) \approx y_j$$

as well as possible, where the quality of this approximation needs to be judged on the basis of the statistical or probabilistic properties of the overall population from which future observations will be drawn.

3.2) Classification Algorithm (CA)

K-nearest neighbor algorithm

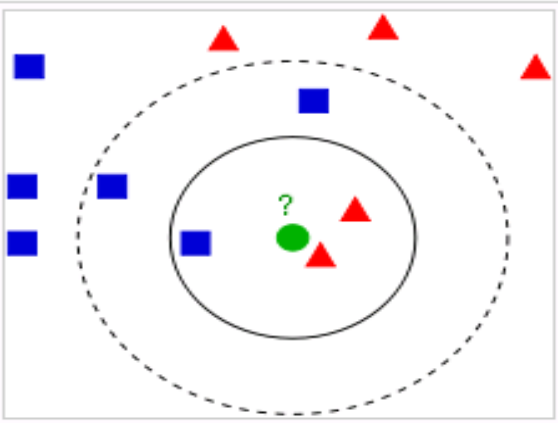
In pattern recognition, the k-**nearest neighbors algorithm** (k-NN) is a method for classifying objects based on closest training examples in the feature space. k-NN is a type of instance based learning, or lazy learning where the function is only approximated locally and all computation is deferred until classification. The k-nearest neighbor algorithm is amongst the simplest of all machine learning algorithms: an object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its ***k*** nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of its nearest neighbor.

The same method can be used for regression, by simply assigning the property value for the object to be the average of the values of its k nearest neighbors. It can be useful to weight the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. (A common weighting scheme is to give each neighbor a weight of 1/*d*, where *d* is the distance to the neighbor. This scheme is a generalization of linear interpolation.)

The neighbors are taken from a set of objects for which the correct classification (or, in the case of regression, the value of the property) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required. The K-nearest neighbor algorithm is sensitive to the local structure of the data.

Nearest neighbor rules in effect compute the decision boundary in an implicit manner. It is also possible to compute the decision boundary itself explicitly, and to do so in an efficient manner so that the computational complexity is a function of the boundary complexity

3.3) Basic Algorithm



Example of *k*-NN classification. The test sample (green circle) should be classified either to the first class of blue squares or to the second class of red triangles. If *k* = 3 it is classified to the second class because there are 2 triangles and only 1 square inside the inner circle. If *k* = 5 it is classified to first class (3 squares vs. 2 triangles inside the outer circle).

The training examples are vectors in a multidimensional feature space, each with a class label. The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples.

In the classification phase, k is a user-defined constant, and an unlabelled vector (a query or test point) is classified by assigning the label which is most frequent among the k training samples nearest to that query point.

Usually Euclidean distance is used as the distance metric; however this is only applicable to continous variables. In cases such as text classification, another metric such as the **overlap metric** (orHamming distance) can be used. Often, the classification accuracy of "k"-NN can be improved significantly if the distance metric is learned with specialized algorithms such as e.g. Large Margin Nearest Neighbor or Neighbourhood components analysis.

A drawback to the basic "majority voting" classification is that the classes with the more frequent examples tend to dominate the prediction of the new vector, as they tend to come up in the *k* nearest neighbors when the neighbors are computed due to their large number.One way to overcome this problem is to weight the classification taking into account the distance from the test point to each of its k nearest neighbors.

3.4) Parameter selection

The best choice of *k* depends upon the data; generally, larger values of k reduce the effect of noise on the classification, but make boundaries between classes less distinct. A good *k* can be selected by various heuristic techniques, for example, cross - validation. The special case where the class is predicted to be the class of the closest training sample (i.e. when k = 1) is called the nearest neighbor algorithm.

The accuracy of the k-NN algorithm can be severely degraded by the presence of noisy or irrelevant features, or if the feature scales are not consistent with their importance. Much research effort has been put intoselecting or scaling features to improve classification. A particularly popular approach is the use of evolutionary algorithms to optimize feature scaling. Another popular approach is to scale features by the mutual information of the training data with the training classes.

In binary (two class) classification problems, it is helpful to choose K to be an odd number as this avoids tied votes. One popular way of choosing the empirically optimal K in this setting is via bootstrap method

4. Conclusions

This study has given a brief view of different algorithms that can be used for data mining. we have discussed one algorithms each on clustering and classification. Partitional clustering Algorithm K-means algorithm and classification algorithm(K-nearest neighbor algorithm)has been briefly discussed in this paper.

References

1. ^ Bremner D, Demaine E, Erickson J, Iacono J, Langerman S, Morin P, Toussaint G (2005). "Output-sensitive algorithms for computing nearest-neighbor decision boundaries". *Discrete and Computational Geometry* **33** (4): 593–604. doi:10.1007/s00454-004-1152-0.

2. ^ D. G. Terrell; D. W. Scott (1992). "Variable kernel density estimation". *Annals of Statistics* **20**: 1236–1265. doi:10.1214/aos/1176348768.

3. ^ Mills, Peter. "Efficient statistical classification of satellite measurements". *International Journal of Remote Sensing*.

4. ^ Nigsch, F.; A. Bender, B. van Buuren, J. Tissen, E. Nigsch & J.B.O. Mitchell (2006). "Melting Point Prediction Employing k-nearest Neighbor Algorithms and Genetic Parameter Optimization". *Journal of Chemical Information and Modeling* **46** (6): 2412–2422. doi:10.1021/ci060149f. PMID 17125183.

^ P. Hall; B. U. Park; R. J. Samworth (2008). "Choice of neighbor order in nearest-neighbor classification". *Annals of Statistics* **36**: 2135–2152. doi:10.1214/07-AOS537.

5. ^ Cover TM, Hart PE (1967). "Nearest neighbor pattern classification". *IEEE Transactions on Information Theory* **13** (1): 21–27. doi:10.1109/TIT.1967.1053964.

6. ^ Toussaint GT (April 2005). "Geometric proximity graphs for improving nearest neighbor methods in instance-based learning and data mining". *International Journal of Computational Geometry and Applications* **15** (2): 101–150. doi:10.1142/S0218195905001622.

7. ^ MacQueen, J. B. (1967). Some Methods for classification and Analysis of Multivariate Observations, Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, 1:281-297

8. ^ Bezdek, James C. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. ISBN 0306406713

9. ^ Google News personalization: scalable online collaborative filtering

^ [1], Cluster Analysis: Basic Concepts and Algorithms.

10. ^ Google News personalization: scalable online collaborative filtering

11. ^ [1], Cluster Analysis: Basic Concepts and Algorithms

12. ^ a b Finch, H. (2005). Comparison of distance measures in cluster analysis with dichotomous data. Journal of Data Science, 3, 85-100

13. ^ a b c d e Huberty, C. J., Jordan, E. M., & Brandt, W. C. (2005). Cluster analysis in higher education research. In J. C. Smart (Ed.), Higher Education: Handbook of Theory and Research (Vol. 20, pp. 437-457). Great Britain: Springer.

14. ^ a b c d e f g h i Hattie (2002). Schools Like Mine: Cluster Analysis of New Zealand Schools. *Technical Report 14, Project asTTle*. University of Auckland.

^ a b Bennett, S. N. (1975). Cluster analysis in educational research: A non-statistical introduction. Research Intelligence, 1, 64-70

Dr A.N.Pathak , Professor and head Applied Scinece Department NITTTR,Chandigarh, member of professional Body : Fellow of Institution of Engineers India. Life member of Indian Society of Biotechnology. Gold medalist Institution of Engineers India.
Educational Qualification**:**M.Sc,B.Tech,M.Tech, Phd (chem-Engineering IIT Delhi) FIE Post Doctorate stuttgart University(Germany)
Gold medalist at Birbal Savitree Shahani Foundation area of Specilaization :Applied Chemistry,Biotechnoligy,Nanotechnology,IPR, Chemical engineering,Fluid Mechanics, Applied Scineces

Manu Sehgal Asssitant professor of Computer Science in Information Technology Department of GGDSD College,Chandigarh.She has done her bachelors in Computer Application and Masters in Information Technology with distinction from India. Aera of specialzation : Database Management.

Divya Christopher Senoir Lecturer in department of Biotechnology Management.She did her Bachlors in commerce and Masters in Management in India. Aera of specialzation: operations