Mining Clusters and Sparse Regions through Decision Tree Construction

Bing Liu
School of Computing
National University of Singapore
http://www.comp.nus.edu.sg/~liub

(The paper will appear in ACM CIKM-2000, by B. Liu, Y. Xia and P. S. Yu)
A longer version of the paper is available as an IBM Research Report, RC 21695 (97737)
by B. Liu, Y. Xia and P. S. Yu

Clustering

- Clustering is an exploratory data analysis task.
- Its objective is to find the intrinsic structure of data by organizing objects in the data into similarity groups or clusters.
- Clustering is often called unsupervised learning

Clusters and Sparse Regions

- Although Cluster (or dense) regions are useful, sparse regions (or holes) are also important.
- Limited work has been done on finding holes.

E.g., in a marketing application, clusters may represent different segments of customers, while the holes represent non-customers.

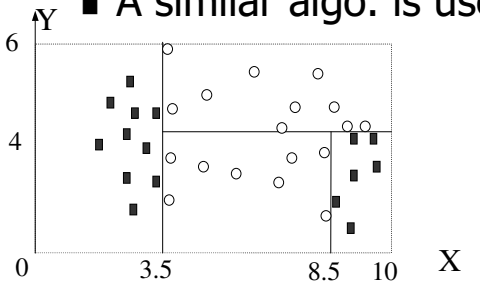
- Knowing the profiles of non-customers allows the company to modify the services or products and/or of doing more promotion in order to attract these potential customers.

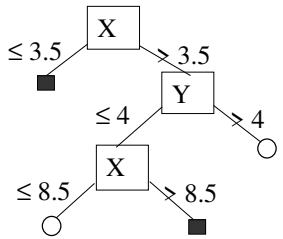
Decision Tree Construction

- Decision tree construction is a classic technique for classifying data of various classes (at least two) - supervised learning
- The dataset for classification consists of a set of records with pre-assigned class labels.
- A decision tree algorithm uses a *purity function* (or criterion) to partition the data space into regions in such a way that each region is occupied by only one class of data points (records) as much as possible.

Decision Tree Construction

- Recursively partition the data space.
- Each partition is determined by a *purity criteria*, e.g. information gain.
- A similar algo. is used in our method.





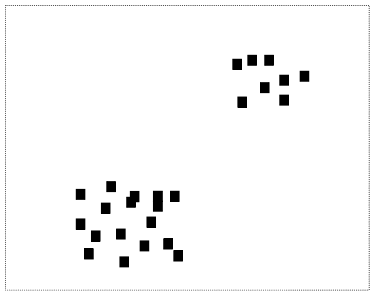
Can We Use Decision Trees for Clustering?

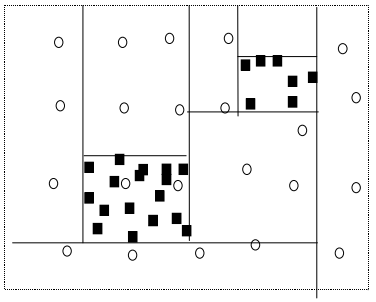
- Clearly, the decision tree technique is not directly applicable to clustering
  - As datasets for clustering have no pre-assigned class labels.
- We present a technique to solve this problem, called CLTree.

The Basic Idea

- Y class data: We regard each record (or data point) in the dataset to have a class label, *Y*.
- N class data: We then assume that the data space is uniformly distributed with another type of points called *non-existing points*. We give them the class label, *N*.
- With the *N* points added to the data space, our problem of partitioning the data space into dense regions and sparse regions becomes a classification problem.
- A decision tree algorithm can now be used.

An Example



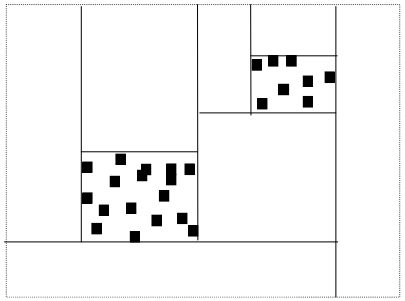


Do not Physically Add N points

Physically adding N points to data is not desirable.

- Too many data points
- A subtle but important issue is that it is unlikely that we can have points truly uniformly distributed in a very high dimensional space.

We aim to achieve the following



The Proposed Technique

- Assume the existence of N points
  - How many N points to introduce?
    - It depends

The proposed technique has two steps:

- Construct the cluster tree.
- Prune the clustering tree.

Advantages of CLTree

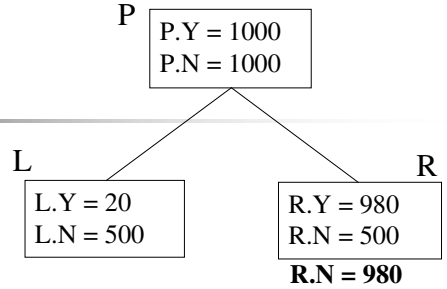
- It is able to find "natural" clusters.
- It is able to find clusters in subspaces as well as in the full dimensional space.
- It provides descriptions of the resulting clusters in terms of hyper-rectangles.
- It can handle large datasets of high dimensionality efficiently.
- It also finds sparse regions or holes.

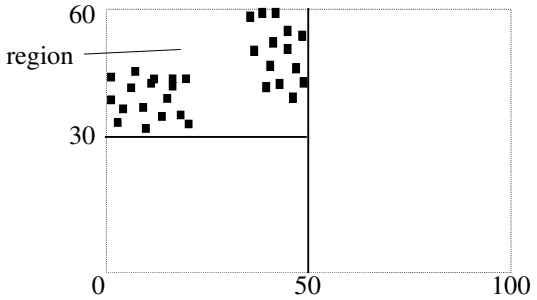
Introduce N points (how many?)

- **If** # of *N* points inherited from the parent node of a node *E* is less than # of *Y* points in *E* **then**

  # of *N* points for *E* is increased to the # of *Y* points in *E*
- **else** # of inherited *N* points from its parent is used for *E*

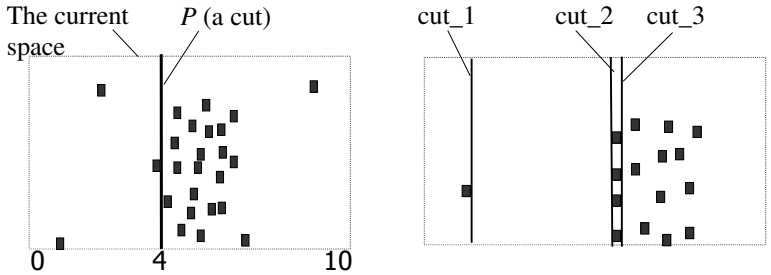
An Example





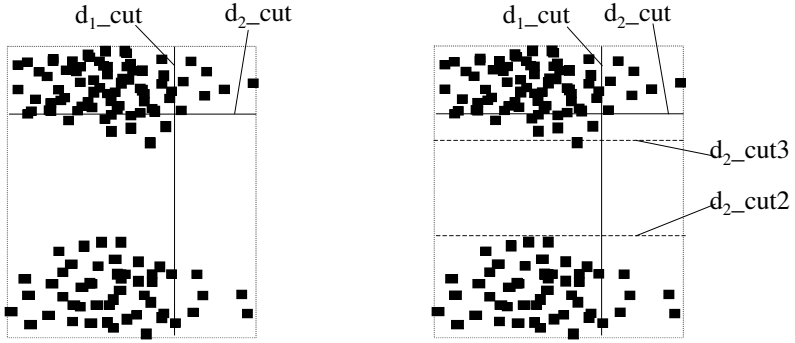
Modifications to the Decision Tree Algorithm

- Info. gain needs # of Y and N points. Since we do not physically add N points
  - We need to compute # of N points (easy).
- Evaluate on both sides of data points



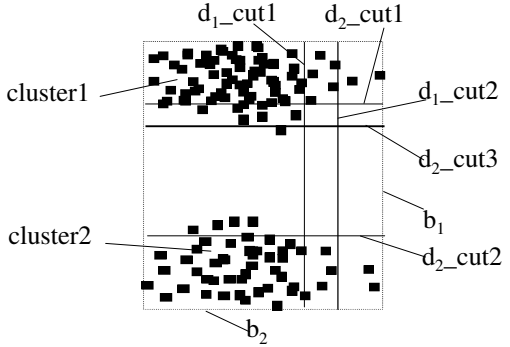
Info. gain is not Sufficient

- The cut with the best information gain tends to cut into clusters.



Lookahead Gain Criterion

- We need to lookhead to find better cuts.
- We do more evaluations.



User-Oriented Pruning of Cluster Trees

- The recursive partitioning method of building cluster trees will divide the data space until each partition contains only points of a single class, or until no test (or cut) offers any improvement.
- The result is often a very complex tree that partitions the space more than necessary.

Two Possible Solutions

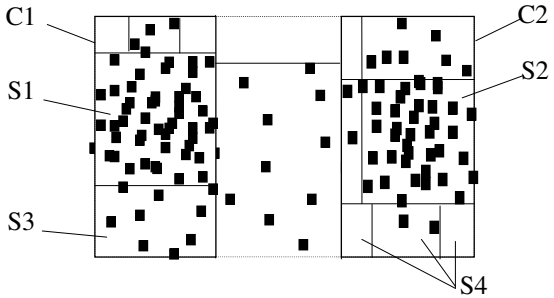
Stopping: deciding not to divide a region any further, or

Pruning: removing some of the tree parts after the tree has been built.

- Stopping is hard.
- Pruning is more realistic (still hard).

Clustering is a Subjective Task

- A tree partitions the space into dense & sparse regions at various levels of details.
- How many clusters? What are they?

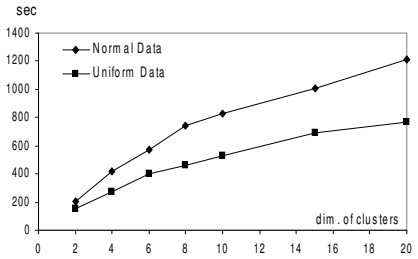


Browsing and Pruning

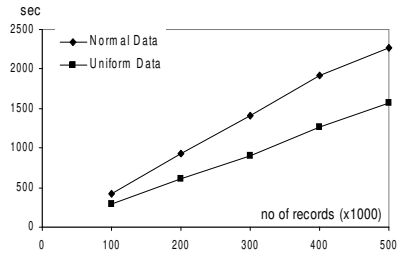
- Browsing: freely browsing the tree.
- User-oriented pruning: The tree is pruned using two user-specify parameters.
  - *min_y*: min # of Y points in a region.
  - *min_rd*: It specifies whether a *N* region (node) *E* should join an adjacent cluster (*Y*) region *F* to form a bigger cluster. If the relative density of *E*, *E.Y/E.N,* is greater than *min_rd*, then join.

Experiments

- We generate data using normal and uniform distributions.

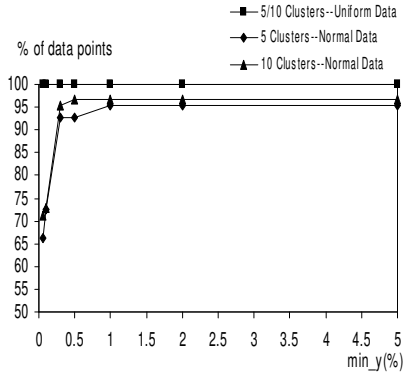


Scalability with the dimensionality of hidden clusters

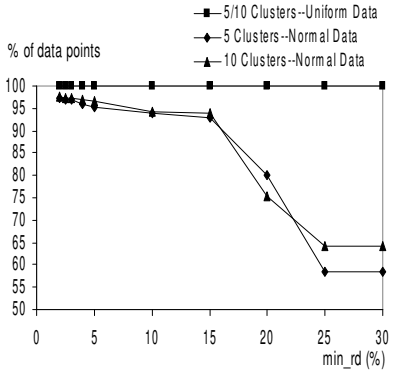


Scalability with the dataset size of hidden clusters

Effects of min_y and min_rd



Percentage of cluster data points found with *min_y*



Percentage of cluster data points found with *min_rd*

An Application

Clusters	Courses												% of Students
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	
1	B-A	C-A	C-A	C-A	C-A	C-A	C-A	C-A	D-A	B-A	C-A	B-A	14.2%
2		C-A								F-C			47.7%
3	F-C	D-D	F-C	F-D	D-D	F-D	D-D	F-C		D-D	F-D	F-D	6.5%
4	F-D	F-D	F-D	F-D	F-D	F-C	C-A			F-C			8.1%

Summary

- We proposed a new clustering technique, called CLTree.
- It is based on decision tree building, a supervised learning technique.
- It has advantages:
  - It finds "natural" clusters (no assumptions made).
  - It is suitable for subspace and full space clustering.
  - It provides descriptions of the clusters.
  - It comes with an important by-product, the empty (sparse) regions.