Clustering High-Dimensional Data

BY

T.R.Lekhaa

AP-IT

SNSCE

Clustering High-Dimensional Data

- Clustering high-dimensional data
  - Many applications: text documents, DNA micro-array data
  - Major challenges:
    - Many irrelevant dimensions may mask clusters
    - Distance measure becomes meaningless—due to equi-distance
    - Clusters may exist only in some subspaces
- Methods
  - Feature transformation: only effective if most dimensions are relevant
    - PCA & SVD useful only when features are highly correlated/redundant
  - Feature selection: wrapper or filter approaches
    - useful to find a subspace where the data have nice clusters
  - Subspace-clustering: find clusters in all the possible subspaces
    - CLIQUE, ProClus, and frequent pattern-based clustering

The Curse of Dimensionality

- Data in only one dimension is relatively packed
- Adding a dimension “stretch” the points across that dimension, making them further apart
- Adding more dimensions will make the points further apart—high dimensional data is extremely sparse
- Distance measure becomes meaningless—due to equi-distance

CLIQUE (Clustering In QUEst)

- Agrawal, Gehrke, Gunopulos, Raghavan (SIGMOD'98)
- Automatically identifying subspaces of a high dimensional data space that allow better clustering than original space
- CLIQUE can be considered as both density-based and grid-based
  - It partitions each dimension into the same number of equal length interval
  - It partitions an m-dimensional data space into non-overlapping rectangular units
  - A unit is dense if the fraction of total data points contained in the unit exceeds the input model parameter
  - A cluster is a maximal set of connected dense units within a subspace

CLIQUE: The Major Steps

- Partition the data space and find the number of points that lie inside each cell of the partition.
- Identify the subspaces that contain clusters using the Apriori principle
- Identify clusters
  - Determine dense units in all subspaces of interests
  - Determine connected dense units in all subspaces of interests.
- Generate minimal description for the clusters
  - Determine maximal regions that cover a cluster of connected dense units for each cluster
  - Determination of minimal cover for each cluster

Strength and Weakness of *CLIQUE*

- Strength
  - *automatically* finds subspaces of the highest dimensionality such that high density clusters exist in those subspaces
  - *insensitive* to the order of records in input and does not presume some canonical data distribution
  - scales *linearly* with the size of input and has good scalability as the number of dimensions in the data increases
- Weakness
  - The accuracy of the clustering result may be degraded at the expense of simplicity of the method

Frequent Pattern-Based Approach

- Clustering high-dimensional space (e.g., clustering text documents, microarray data)
  - Projected subspace-clustering: which dimensions to be projected on?
    - CLIQUE, ProClus
  - Feature extraction: costly and may not be effective?
  - Using frequent patterns as "features"
    - "Frequent" are inherent features
    - Mining freq. patterns may not be so expensive
- Typical methods
  - Frequent-term-based document clustering
  - Clustering by pattern similarity in micro-array data (pClustering)

Clustering by Pattern Similarity (*p*-Clustering)

- Right: The micro-array “raw” data shows 3 genes and their values in a multi-dimensional space
  - Difficult to find their patterns
- Bottom: Some subsets of dimensions form nice shift and scaling patterns

- Right: The micro-array “raw” data shows 3 genes and their values in a multi-dimensional space
  - Difficult to find their patterns
- Bottom: Some subsets of dimensions form nice shift and scaling patterns

Why *p*-Clustering?

- Microarray data analysis may need to
  - Clustering on thousands of dimensions (attributes)
  - Discovery of both shift and scaling patterns
- Clustering with Euclidean distance measure? — cannot find shift patterns
- Clustering on derived attribute $A_{ij} = a_i - a_j$? — introduces N(N-1) dimensions
- Bi-cluster using transformed mean-squared residue score matrix (I, J)

$$H(IJ) = \frac{1}{|I||J|}\Sigma_{i \in I, j \in J}(d_{ij} - d_{iJ} - d_{Ij} + d_{IJ})^2$$

  - A submatrix is a δ-cluster if $H(I, J) \leq \delta$ for some $\delta > 0$
- Problems with bi-cluster
  - No downward closure property,
  - Due to averaging, it may contain outliers but still within δ-threshold

p-Clustering: Clustering by Pattern Similarity

- Given object x, y in O and features a, b in T, pCluster is a 2 by 2 matrix

- A pair (O, T) is in δ-pCluster if for any 2 by 2 matrix X in (O, T), pScore(X) ≤ δ for some δ > 0
- Properties of δ-pCluster
  - Downward closure
  - Clusters are more homogeneous than bi-cluster (thus the name: pair-wise Cluster)
- Pattern-growth algorithm has been developed for efficient mining
- For scaling patterns, one can observe, taking logarithmic on will lead to the pScore form