Clustering Large Document Collections

- A critical literature review

Zheyuan Yu

zyu@cs.dal.ca

Jun 27, 2003

Outline

- Introduction
- Challenge of High Dimensional Data
- Techniques for High Dimension problem
- Conclusion

Introduction
Document Clustering

Document clustering is the process of organizing documents into clusters so that

- Documents within a cluster have high similarity in comparison to one another.
- But are very dissimilar to documents in other clusters.

1. Introduction

How to represent the document's information – Vector Space Model

- Also known as bag-of-words model
- Documents are represented as vectors
- Each direction of the vector space corresponds to a unique term in the document collection

$$D_i =< w_{i1}, w_{i2}, \ldots, w_{it} >$$

1. Introduction
Vector Space Model - Weight

tf-idf term weight

- Term frequency - tf.
- Inverse document frequency - idf.

$tf_{ij} = \frac{f_{ij}}{\max_l f_{ij}}$, $\max_l f_{ij}$ is the maximal term frequecy in the i th document

$idf_{ij} = \log \frac{N}{n_i}$, N is the total number of documents,

n_j is the number of documents that contain the j th term

$Weight: w_{ij} = tf_{ij} * idf_{ij}$ $w_{ij} = 0$ if a term is absent

w_{ij} is the term weight of the j th term of the i th document in the document set

1. Introduction

How to calculate similarity

$$D_i = < w_{i1}, w_{i2}, \ldots, w_{it} >$$

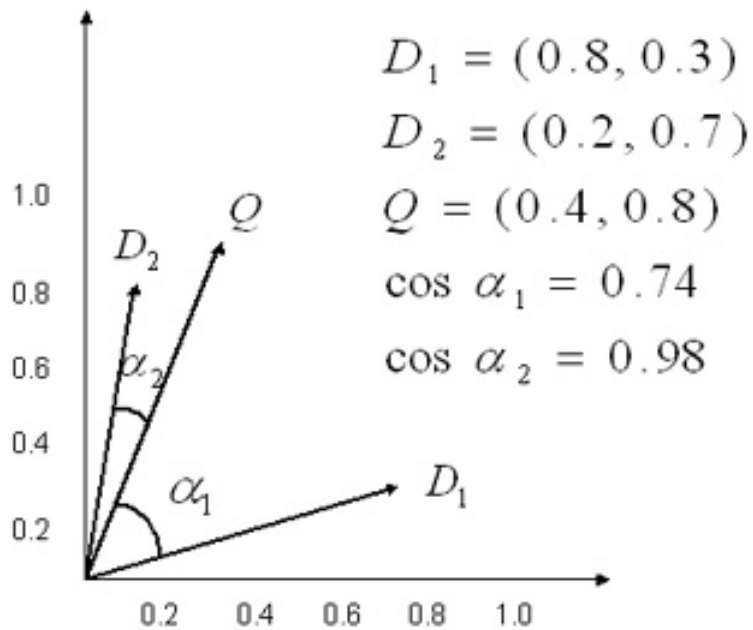
$$D_j = < w_{j1}, w_{j2}, \ldots, w_{jt} >$$

First normalize D_i and D_j

$$\overrightarrow{D_i} = \frac{\overrightarrow{D_i}}{\left|\overrightarrow{D_i}\right|} \qquad \overrightarrow{D_j} = \frac{\overrightarrow{D_j}}{\left|\overrightarrow{D_j}\right|}$$

Then $sim(D_i, D_j) = \overrightarrow{D_i} \cdot \overrightarrow{D_j}$

$$= \sum_{k=1}^{t} w_{ik} * w_{jk}$$

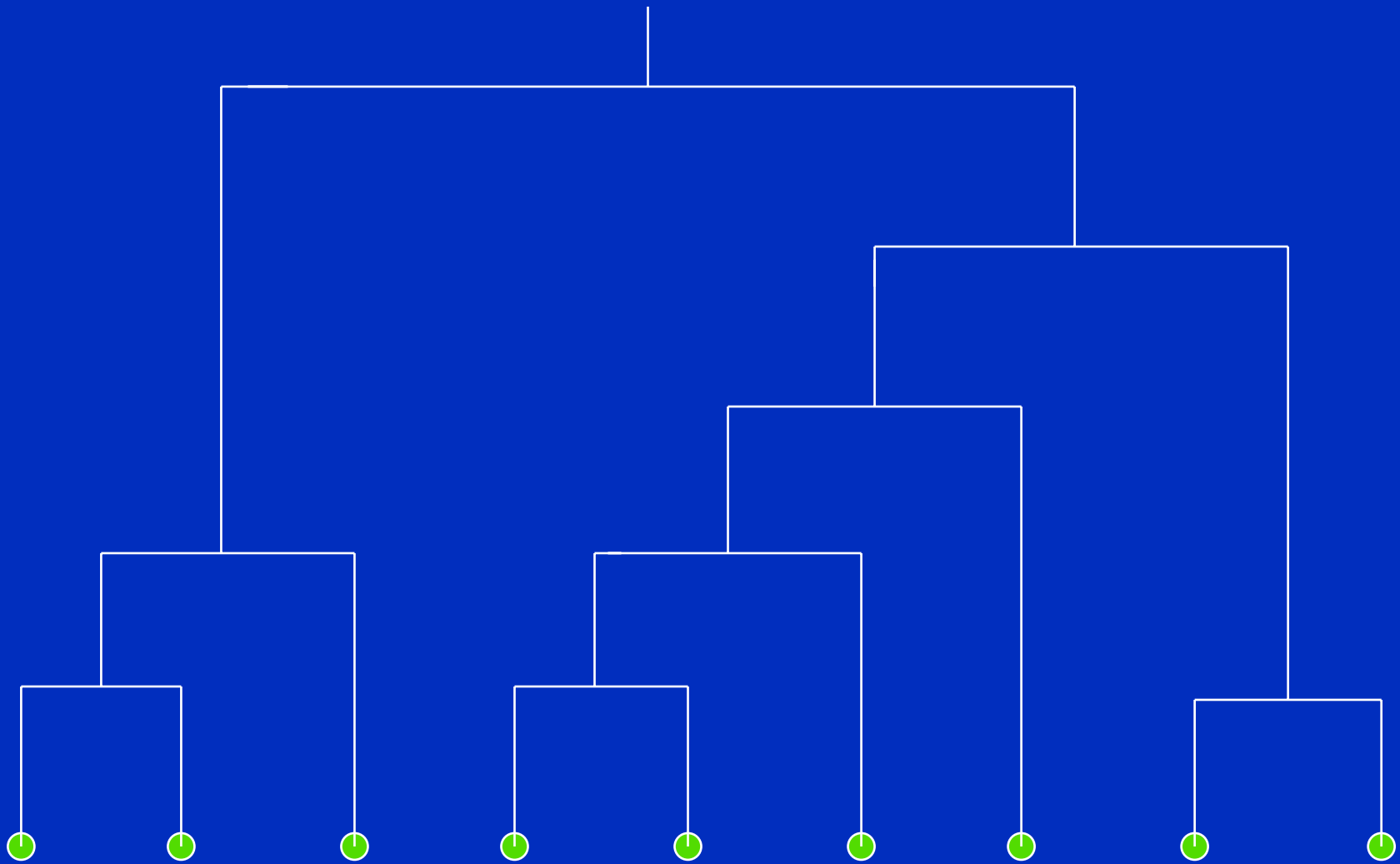


1. Introduction
Document Clustering Algorithms

- **Hierarchical methods**
  - Agglomerative
  - Divisive
- **Partition methods**

1. Introduction

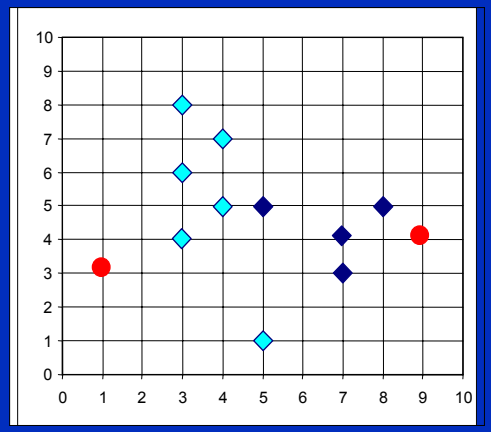
Document Clustering Algorithms- Hierarchical methods



1. Introduction

Document Clustering Algorithms- Partition Methods

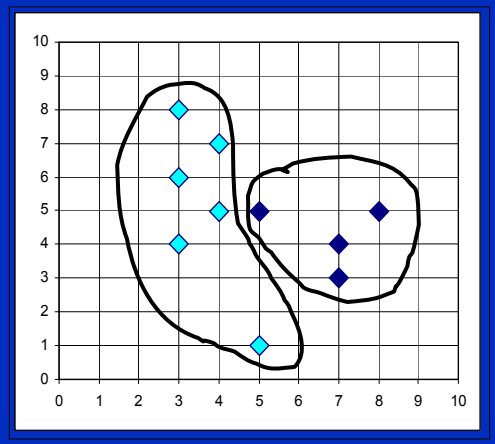
- **Example of K-means**



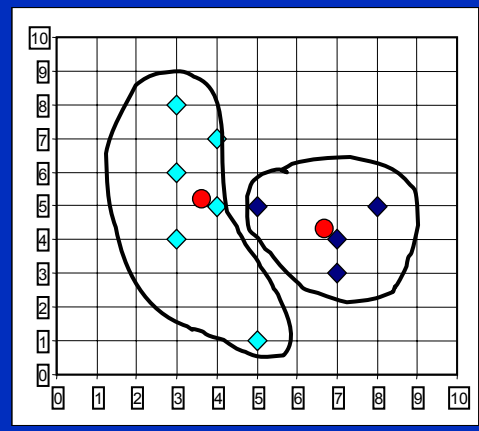
K=2

Arbitrarily choose K object as initial cluster center

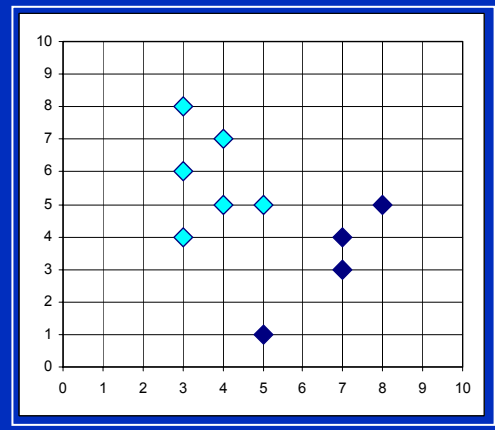
Assign each objects to most similar center



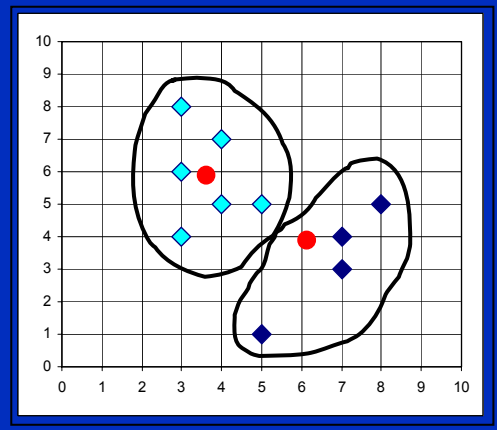
Update the cluster means



reassign



Update the cluster means



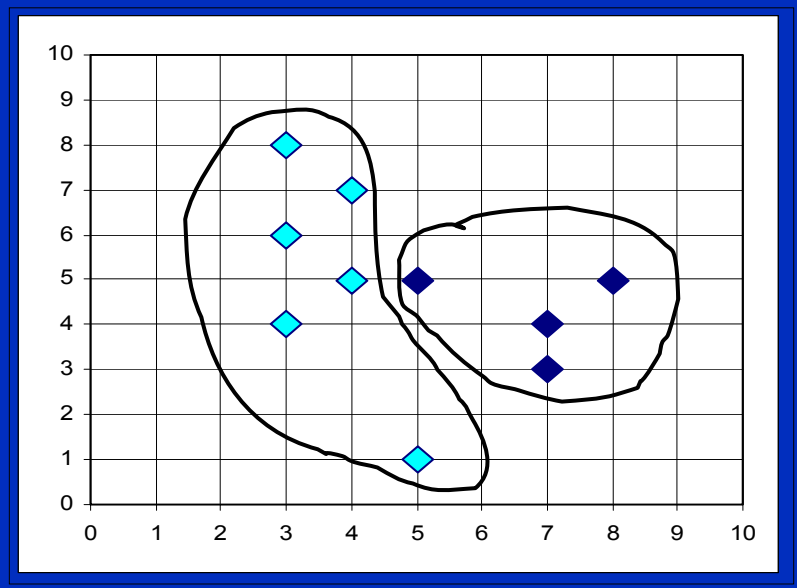
reassign

2. Challenge of High Dimension

- The document vectors are very high dimensional because even a small document collection may have thousands unique terms
- K. Beyer et. al. [1] have shown that in high dimensional space, the distance to the nearest data point approaches the distance to the farthest data point.
- The similarity measure of the clustering algorithms do not work effectively, hence the meaningfulness of clustering may be doubtful.
- This problem was traditionally referred to as dimensionality curse [2]

2. Challenge of High Dimension

- In high dimensional space, the mean values may not differ significantly to each other, hence the cluster means may not provide good separation.



2. Challenge of High Dimension
Techniques

- **Dimensionality Reduction**
- **Subspace Clustering**

3. Dimensionality Reduction

- **Attributes transformations**
  - Principal components analysis (PCA) [3]
  - Singular value decomposition [4]
- **Domain decomposition**
  - Divide dataset into subsets, *canopies* [5]

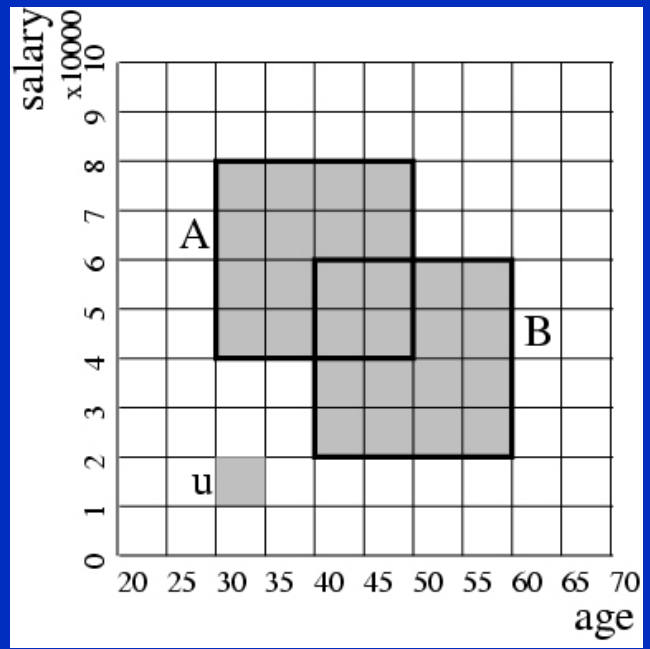
4. Subspace Clustering
CIQUE

- A cluster is a region that has a higher density of points than its surrounding region.
- Partition each dimension into the same number of equal length intervals.
- Identify high-density clusters in the subspaces of a high dimensional data space (the space of a subset of the original attributes)

4. Subspace Clustering

Background

- A = {*A1*, *A2*, ..., *Ad*} is a set of bounded, totally ordered domain.
- $S = A1 \times A2 \times \ldots \times Ad$ is a *d-dimensional space*.
- Each dimension is discretized into ζ equal length intervals.
- Space S is partitioned into rectangular units.
  - unit *u:* {*u1*, *u2*, ..., *ud*} where *ui* = [*li*, *hi*)
- The fraction of total data points contained in the unit is referred as *selectivity* of the unit

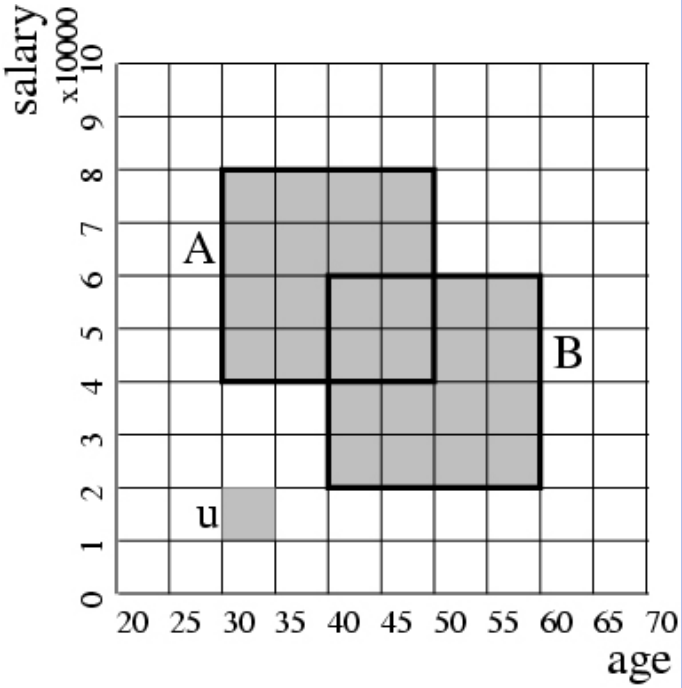


4. Subspace Clustering

Background

- Cluster: a maximal set of connected dense units
- Dense unit: selectivity(u) > τ
  - where τ is a user specified parameter

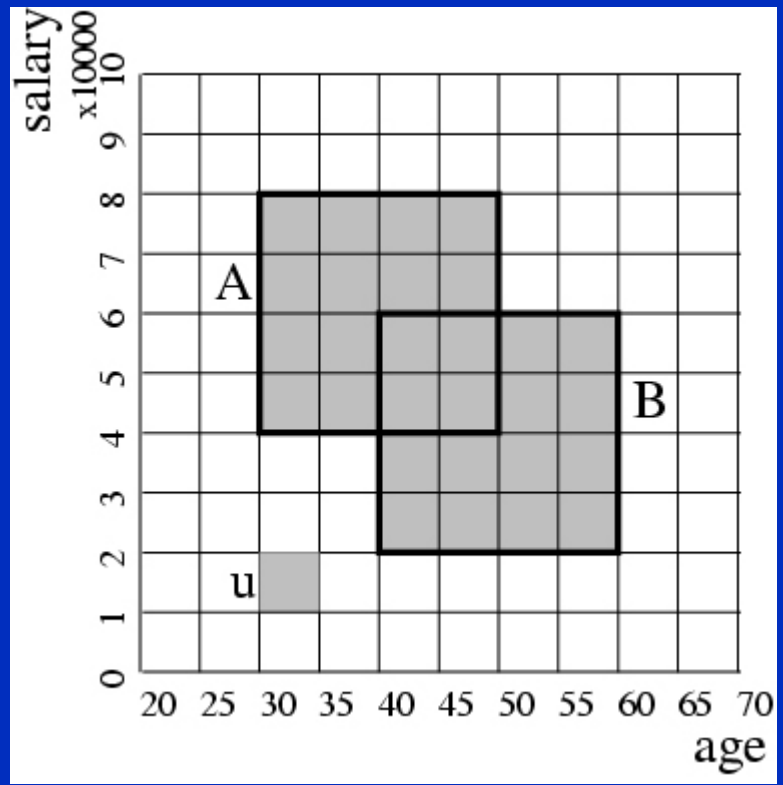
Units $u_1 = \{r_{t_1}, ..., r_{t_k}\}$ and $u_2 = \{r'_{t_1}, ..., r'_{t_k}\}$ are connected if they have a common face : there are k - 1 dimensions, assume dimensions $A_{t_1}, ..., A_{t_{k-1}}$, such that $r_{t_j} = r'_{t_j}$ and either $h_{t_k} = l'_{t_k}$ or $h'_{t_k} = l_{t_k}$.



4. Subspace Clustering

Background

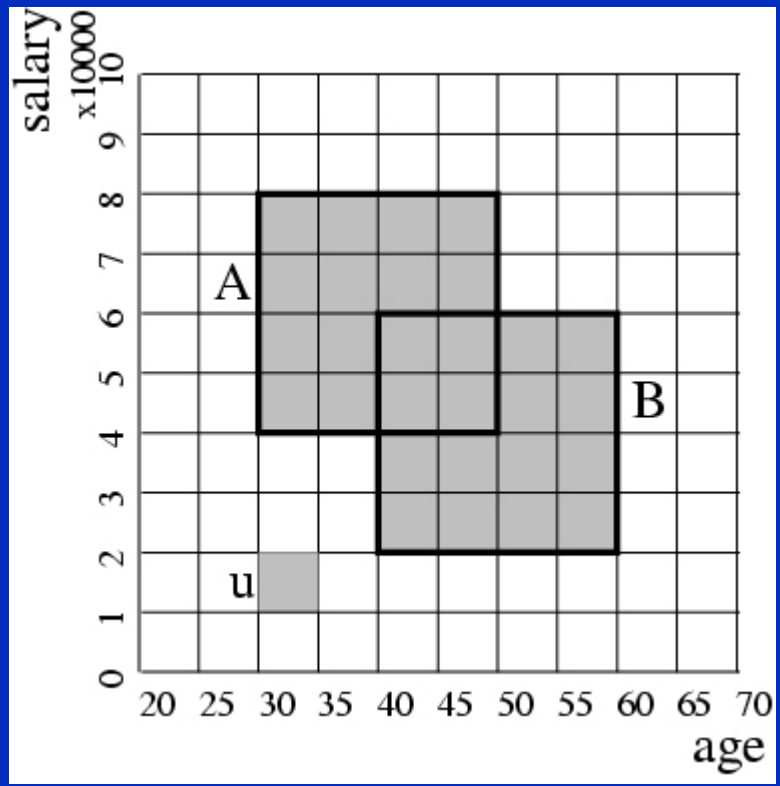
- Region in k dimensions: a axis-parallel rectangular k-dimensional set
- Region R is contained in cluster C if R∩C=R.
- maximal region R in C: no proper superset of R is contained in C
- minimal description: a non-redundant covering of the cluster with maximal regions



$((30 \leq age < 50) \wedge (40K \leq salary < 80K)) \vee ((40 \leq age < 60) \wedge (20K \leq salary < 60K))$

4. Subspace Clustering

Problem Statement



The Problem: Given a set of data points and the input parameters, τ and ξ, find clusters in all subspaces of the original data space and present a minimal description of each cluster in the form of a DNF expression.

$((30 \leq age < 50) \wedge (40K \leq salary < 80K)) \wedge ((40 \leq age < 60) \vee (20K \leq salary < 60K))$

4. Subspace Clustering

Example

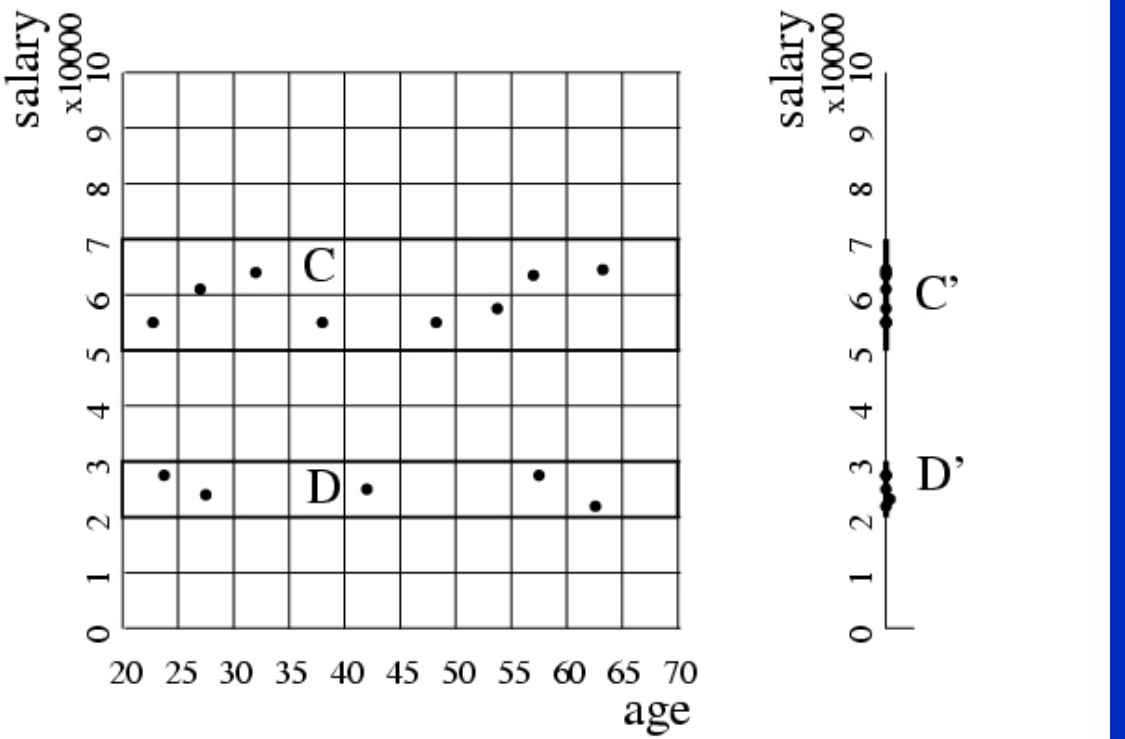


Figure 2: Identification of clusters in subspaces (projections) of the original data space.

If threshold is set to 20%, no unit is dense in 2-dimension. But there are 3 1-dimensional dense units. They form two clusters in the 1-dimensional salary space: C' = $5 \leq \text{salary} < 7$ and D' = $2 \leq \text{salary} < 3$

4. Subspace Clustering
Overview CLIQUE Algorithm

- Identification of subspaces that contains clusters
- Identification of clusters
- Generation of minimal description for the clusters

4. Subspace Clustering

Identification of subspaces that contain clusters

- **Lemma 1 (Monotonicity) : If a collection of points S is a cluster in a k-dimensional space, then S is also part of a cluster in any (k-1)-dimensional projection of this space.**

4. Subspace Clustering

Identification of subspaces that contain clusters - Algorithm

1. determine dense units in one-dimensional space
2. k =2
3. generate candidate (self-joining) of k dimensional units from (k-1)-dimensional dense units.
4. if candidates are not empty

   find dense units

   k = k + 1

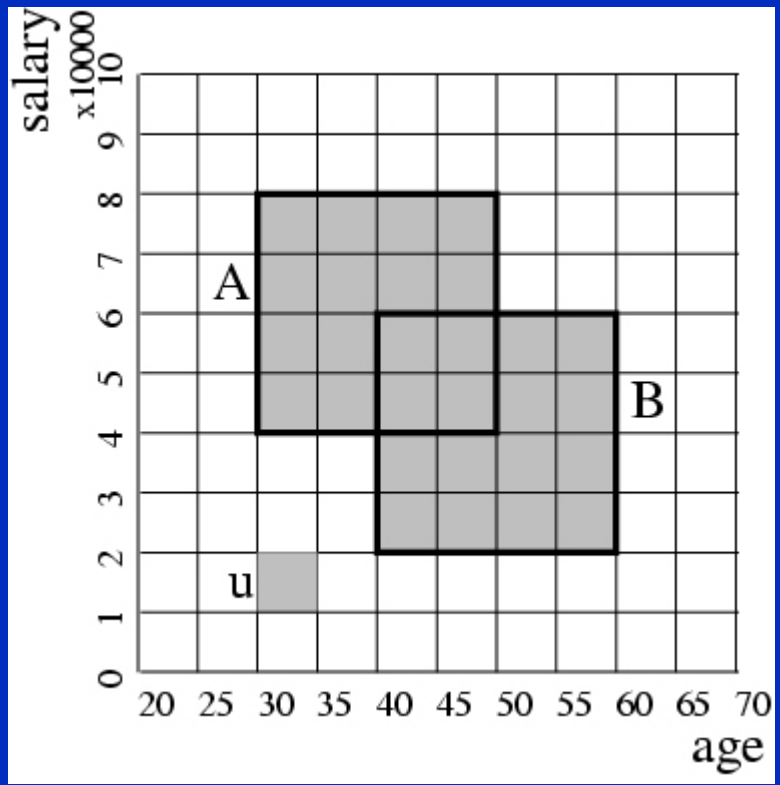
   go to step 3

Example: Generating 3 dimensional candidates

<A1, A2>
<A1, A4>
<A1, A5>
<A2, A3>

<A1, A2, A4>
<A1, A2, A5>
<A2, A4, A5>
<A1, A2, A3>

4. Subspace Clustering
Finding clusters



This problem is equivalent to finding connected components in a graph defined as follows: Graph vertices correspond to dense units, and there is an edge between two vertices if and only if the corresponding dense units have a common face

4. Subspace Clustering

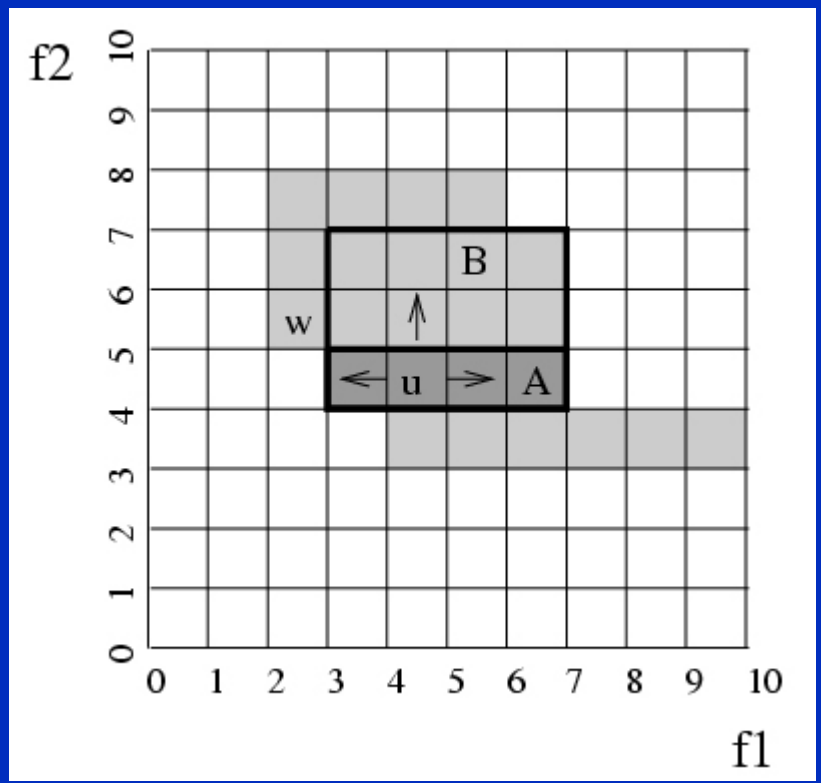
Finding clusters – Algorithm

Use depth-first search algorithm

- Start with one unit u in D, assign it the first cluster number
- Find all the units it connected to
- If there still are units in D that have not been visited, repeat the procedure.

4. Subspace Clustering

Generating minimal cluster descriptions

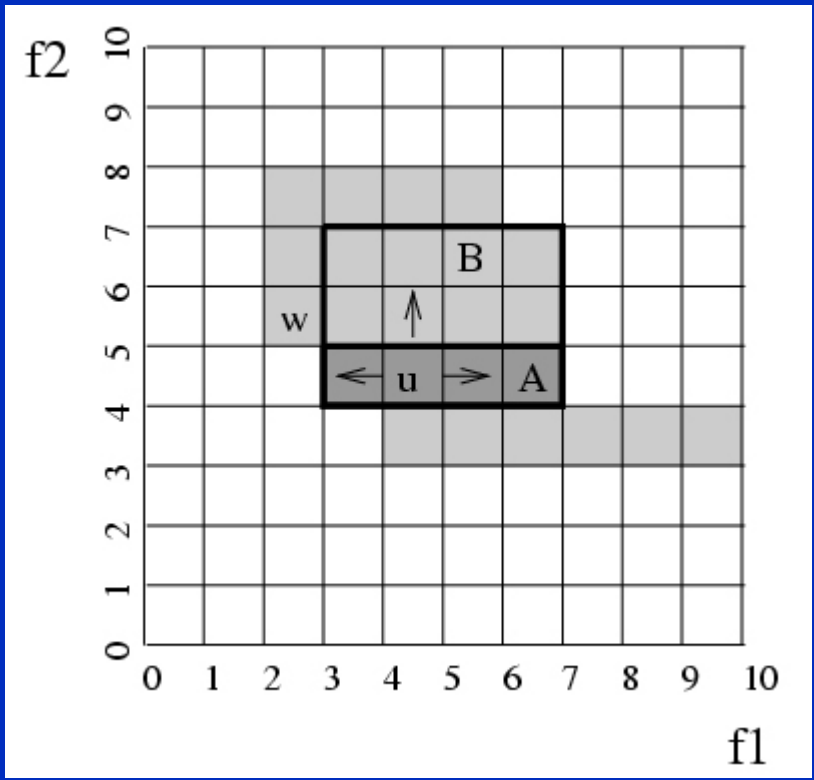


- The optimal cover is the cover with the minimal number of rectangles.
- Computing the optimal cover is know to be NP-hard
- solution to the problem – approximating the smallest set cover：
  - greedily cover the cluster by a number of maximal regions
  - discard the redundant regions to generate a minimal cover

$$((3 \le f1 < 7) \wedge (4 \le f2 < 7)) \vee ((2 \le f1 < 6) \wedge (5 \le f2 < 8)) \vee ((4 \le f1 < 10) \wedge (3 \le f2 < 4))$$

4. Subspace Clustering

Generating minimal cluster descriptions - Greedy Growth for Minimal Cover



1) begin with an arbitrary dense unit $u \in C$

2) Greedily grow to a maximal region covering u, add to R

3) Repeat 2) until all $u \in C$ are covered by some maximal regions in R

4) Remove from the cover the smallest maximal region which is redundant.

4. Subspace Clustering

Performance Evaluation

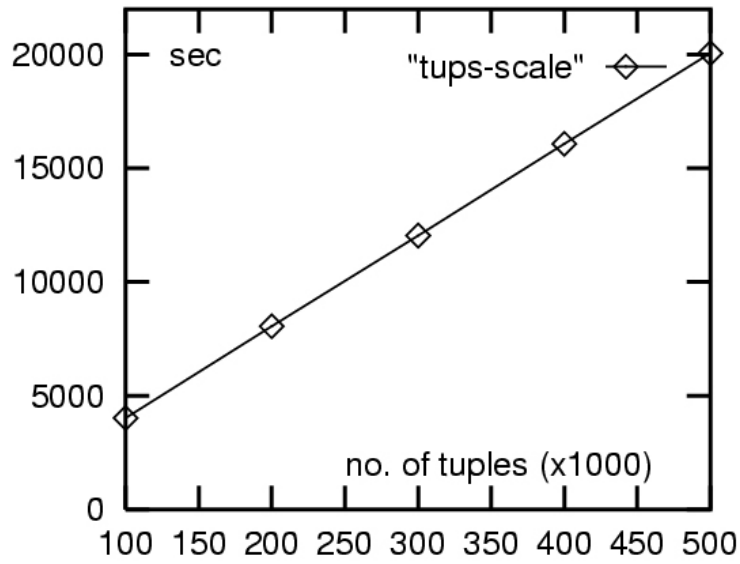


Figure 6: Scalability with the number of data records.

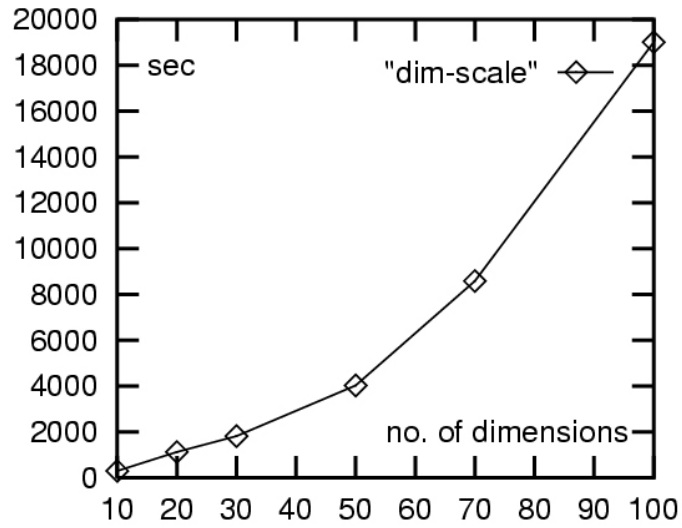


Figure 7: Scalability with the dimensionality of the data space.

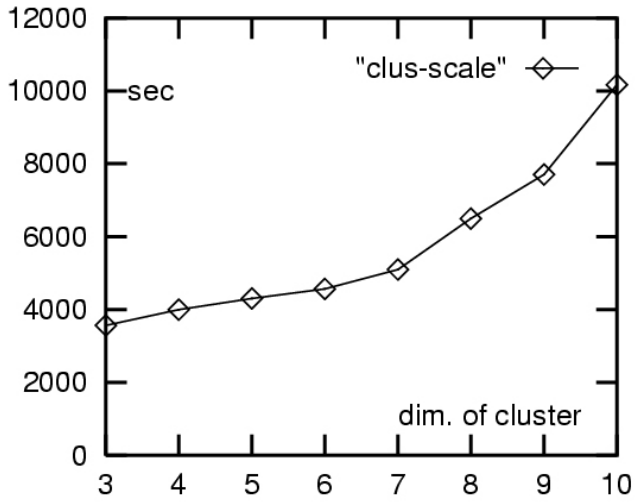


Figure 9: Scalability with the dimensionality of the hidden cluster.

4. Subspace Clustering

Comparison with BIRCH, DBSCAN and SVD

Use clusters embedded in 5-dimensional subspaces while varying the dimensionality of the space from 5 to 50.

CLIQUE was able to recover all clusters in every case

CLIQUE has no advantage

Table 1: BIRCH experimental results.

Dim. of data	Dim. of clusters	No. of clusters	Clusters found	True clusters identified
5	5	5	5	5
10	5	5	5	5
20	5	5	3,4,5	0
30	5	5	3,4	0
40	5	5	3,4	0
50	5	5	3	0

Table 2: DBSCAN experimental results.

Dim. of data	Dim. of clusters	No. of clusters	Clusters found	True clusters identified
5	5	5	5	5
7	5	5	5	5
8	5	5	3	1
10	5	5	1	0

Table 3: SVD decomposition experimental results.

Dim. of data (d)	Dim. of clusters	No. of clusters	$r_{d/2}$	$r_{(d-5)}$	$r_{(d-1)}$
10	5	5	0.647	0.647	0.937
20	5	5	0.606	0.827	0.969
30	5	5	0.563	0.858	0.972
40	5	5	0.557	0.897	0.981
50	5	5	0.552	0.919	0.984

4. Subspace Clustering

Real Dataset

Two Insurance Companies:

Insure1 and 2.

One Department Store

One Bank

Table 4: Real data experimental results.

Dataset	Dim. of data	Dim. of clusters	No. of clusters
Insur1	9	7	2
Insur2	7	4	5
Store	24	10	4
Bank	52	9	1

Meaningful clusters were detected from all those cases.

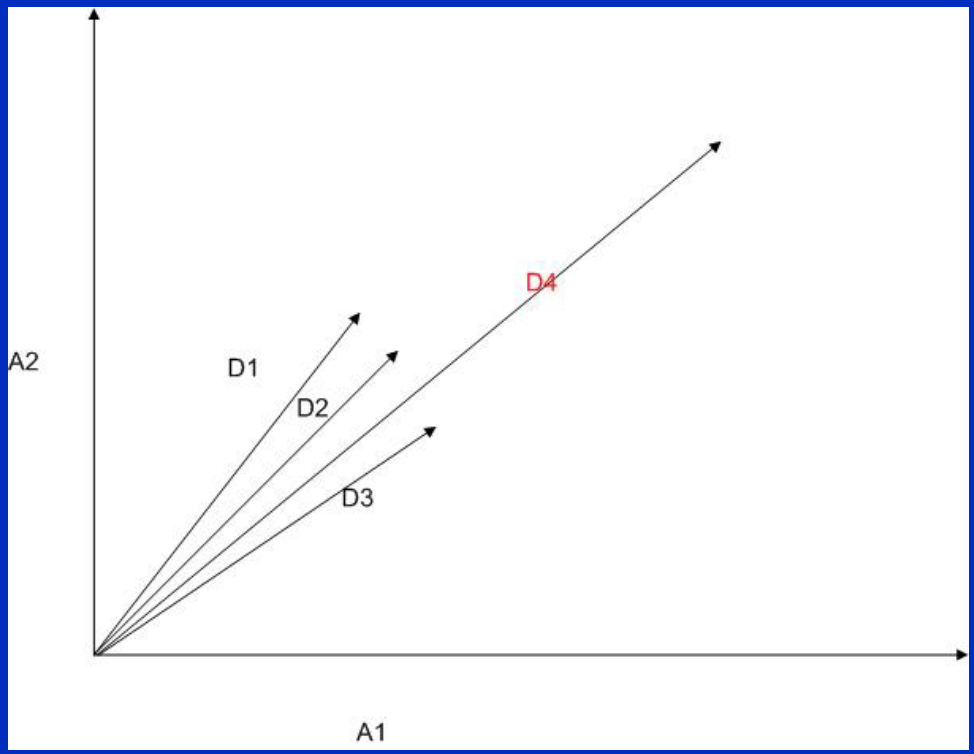
4. Subspace Clustering

Strength and Weakness of CLIQUE

- **Strength**
  - It scales *linearly* with the size of input
  - Tolerating missing values/noise
  - It is *insensitive* to the order of records in input
- **Weakness**
  - User specified parameter: ζ, # of intervals along one dimension and τ the threshold may affect the result.
  - Linear to the size of input, but not linear to the number of dimension

4. Subspace Clustering
Question

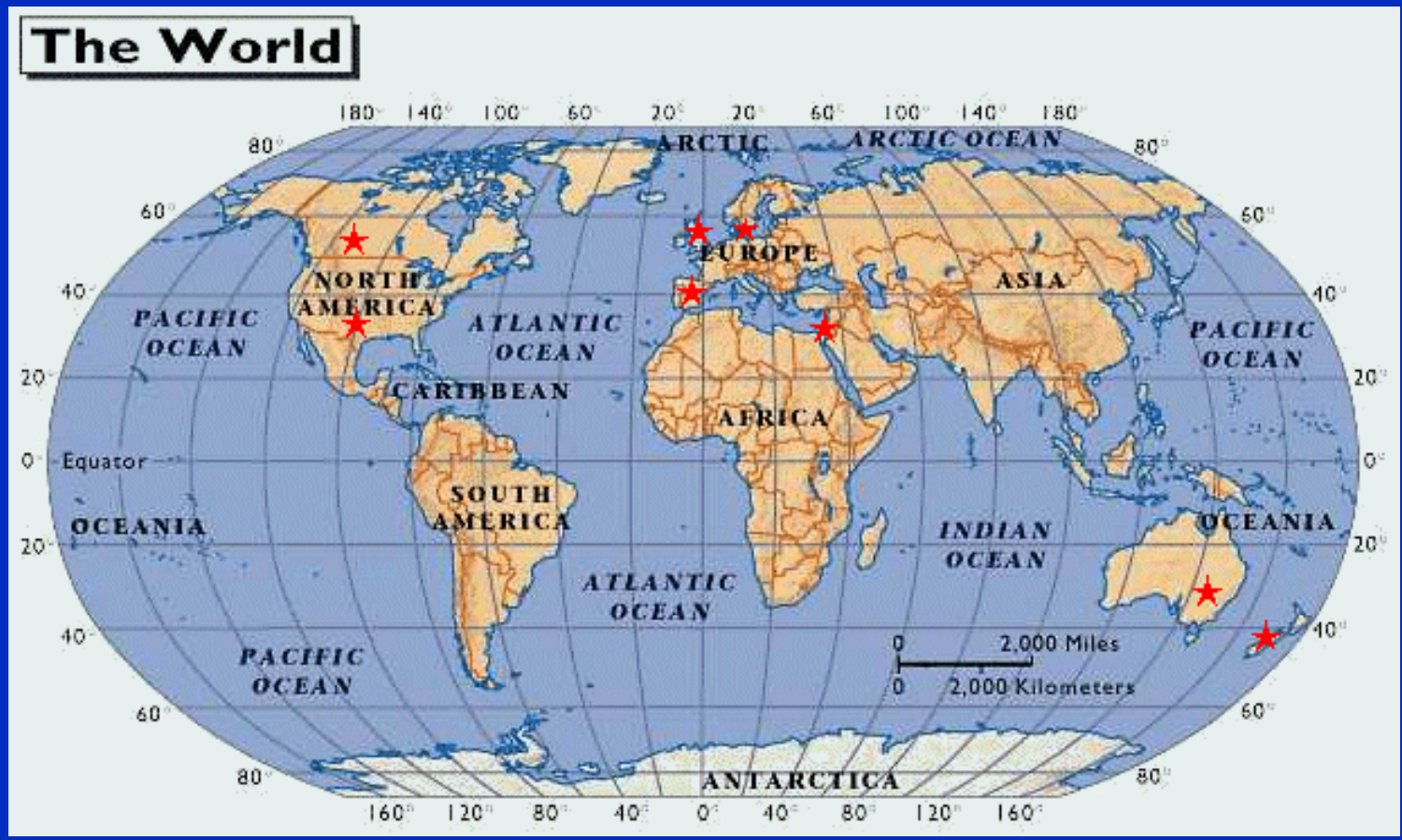
- How to use CLIQU in Document Clustering?



- Normalize the document vectors

4. Subspace Clustering
Question

- How to partition the vector space?



5. Conclusion

- Introduction of the document clustering algorithm
- The challenge of High Dimensional data
- Techniques for High Dimensionality

References

[1] Beyer, K., Goldstein, J., Ramakrishnan, R., and Shaft, U. *When is nearest neighbor meaningful*? In proceeding of the 7th ICDT, Jerusalem, Israel. 1999

[2] R. Bellman. *Adaptive Control Processes: A Guided Tour*. Princeton University, Princeton, 1961

[3] JOLIFFE, I. *Principal Component Analysis*. Springer-Verlag, New York, 1986

[4] M. Berry et. al. *Using linear algebra for intelligent information retrieval*. SIAM Review, 37, 4, 573-595. 1995

[5] A. McCallum et. al. *Efficient clustering of high dimensional data sets with application to reference matching*. In proceedings of the 6th ACM SIGKDD, 169-178, Boston, MA

[6] R. Agrawal et. Al. Automatic subspace clustering of high dimensional data for data mining applications. In Proc. SIGMOD'98. Seattle, Washington, June 1998

Happy Canada Day!

Thanks, any questions?