
Information Theoretic Clustering, Co-clustering and Matrix Approximations

Inderjit S. Dhillon

University of Texas, Austin

**IMA Workshop on Data
Analysis & Optimization
May 7, 2003**

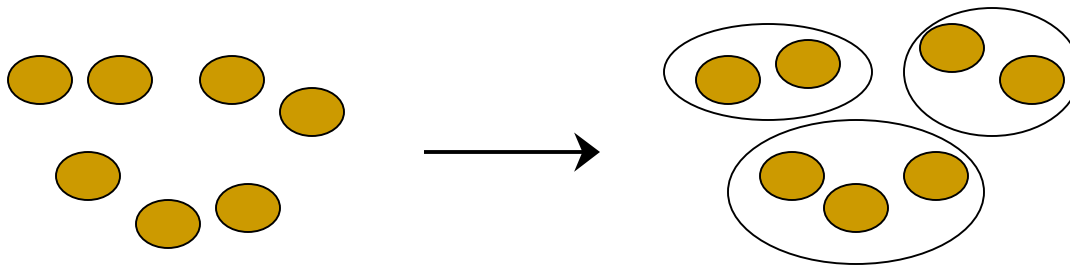
Joint work with Y. Guan, S. Mallela & Dharmendra Modha

Introduction

- Important Tasks in Data Mining
 - Clustering
 - Grouping together of “similar” objects
 - Classification
 - Labelling new objects given an existing grouping
 - Matrix Approximations
 - Reducing dimensionality (SVD, PCA, NNMF,.....)
 - Obstacles
 - High Dimensionality
 - Sparsity
 - Noise
 - Need for robust and scalable algorithms
-

Clustering

- Clustering along “One-Dimension”
 - Grouping together of “similar” objects
 - Hard Clustering -- Each object belongs to a single cluster
 - Soft Clustering -- Each object is probabilistically assigned to clusters



Co-clustering

- Given a multi-dimensional data matrix, co-clustering refers to **simultaneous** clustering along multiple dimensions
 - In a two-dimensional case it is simultaneous clustering of rows and columns
 - Most traditional clustering algorithms cluster along a single dimension
 - Co-clustering is more robust to sparsity
-

Matrix Approximations

- Co-occurrence matrices frequently arise in applications, for example, word-document matrices
 - Matrix characteristics
 - Large
 - Sparse
 - Non-negative
 - Traditional matrix approximations, such as SVD(PCA) do not preserve non-negativity or sparsity
-

$$\begin{bmatrix}
 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
 0 & 1 & 1 & 2 & 0 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1
 \end{bmatrix}$$

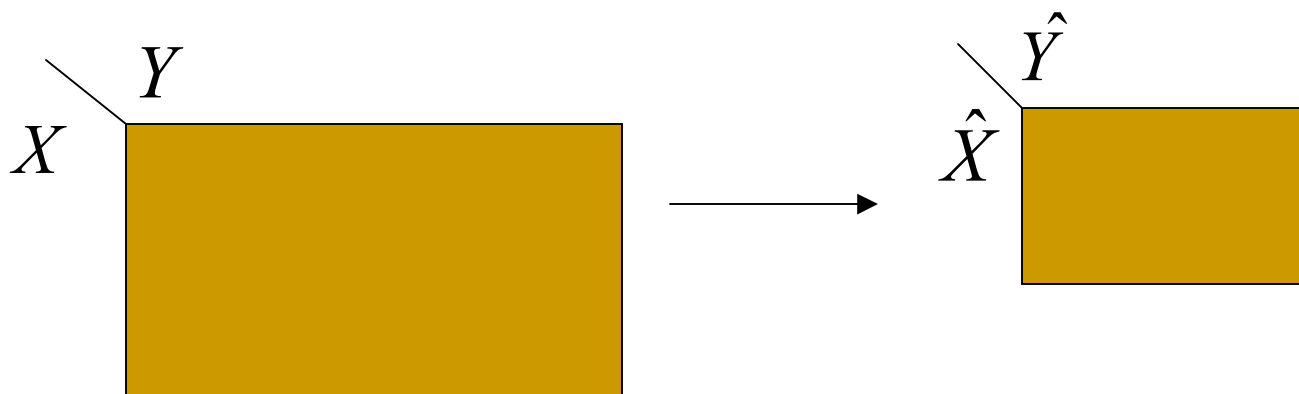
\approx

$$\begin{bmatrix}
 .36 & .460 & .380 & .470 & .183 & 0.050 & -.120 & -.160 & -.100 \\
 .34 & .350 & .340 & .440 & .163 & 0.030 & -.070 & -.100 & -.040 \\
 .35 & .560 & .340 & .440 & .243 & 0.020 & .06 & 0.09 & 0.12 & 0 \\
 .46 & .890 & .660 & .760 & .394 & .503 & 0.08 & 0.12 & 0.19 & 0 \\
 .45 & .122 & .180 & .138 & .566 & 0.070 & -.150 & -.210 & .19 & 0 \\
 .36 & .580 & .380 & .440 & .283 & 0.06 & 0.13 & 0.19 & 0.22 & 0 \\
 .36 & .580 & .380 & .440 & .283 & 0.06 & 0.13 & 0.19 & 0.22 & 0 \\
 .32 & .550 & .540 & .640 & .243 & 0.070 & -.140 & -.200 & -.110 & 0 \\
 .15 & .530 & .220 & .220 & .271 & .514 & .113 & .122 & .443 & .342 & .33 \\
 -.06 & .23 & .34 & -.27 & .140 & .243 & .335 & .567 & .771 & 1.066 & 1.0 \\
 -.06 & .34 & -.15 & -.30 & .200 & .313 & .369 & .679 & .810 & .851 & 1.0 \\
 -.04 & .25 & -.10 & -.20 & .150 & .222 & .250 & .447 & 1.676 & 1.67 & 1.67
 \end{bmatrix}$$

$$\begin{bmatrix}
 .05 & .05 & .05 & 0 & 0 & 0 \\
 .05 & .05 & .05 & 0 & 0 & 0 \\
 0 & 0 & 0 & .05 & .05 & .05 \\
 0 & 0 & 0 & .05 & .05 & .05 \\
 .04 & .04 & 0 & .04 & .04 & .04 \\
 .04 & .04 & .04 & 0 & .04 & .04
 \end{bmatrix}
 \approx
 \begin{bmatrix}
 .0554 & .0554 & .0472 & -.007 & .003 & .003 \\
 .0554 & .0554 & .0472 & -.007 & .003 & .003 \\
 .003 & .003 & -.007 & .0427 & .0551 & .0551 \\
 .003 & .003 & -.007 & .0427 & .0551 & .0551 \\
 .0336 & .0336 & .0228 & .0385 & .0344 & .0344 \\
 .0336 & .0336 & .0228 & .0280 & .0350 & .0350
 \end{bmatrix}$$

Co-clustering and Information Theory

- View (scaled) co-occurrence matrix as a joint probability distribution between row & column random variables



- We seek a hard-clustering of both dimensions such that loss in “Mutual Information”

$$I(X, Y) - I(\hat{X}, \hat{Y})$$

is minimized given a fixed no. of row & col. clusters (similar framework as in Tishby, Pereira & Bialek(1999), Berkhin & Becher(2002))

Information Theory Concepts

- Entropy of a random variable X with probability distribution $p(x)$:

$$H(p) = -\sum_x p(x) \log p(x)$$

- The Kullback-Leibler(KL) Divergence or “Relative Entropy” between two probability distributions p and q :

$$KL(p, q) = \sum_x p(x) \log(p(x)/q(x))$$

- Mutual Information between random variables X and Y :

$$I(X, Y) = \sum_x \sum_y p(x, y) \log\left(\frac{p(x, y)}{p(x)p(y)}\right)$$

Jensen-Shannon Divergence

- Jensen-Shannon(JS) divergence between two probability distributions:

$$\begin{aligned} JS_{\Pi}(p_1, p_2) &= \pi_1 KL(p_1, \pi_1 p_1 + \pi_2 p_2) + \pi_2 KL(p_2, \pi_1 p_1 + \pi_2 p_2) \\ &= H(\pi_1 p_1 + \pi_2 p_2) - \pi_1 H(p_1) - \pi_2 H(p_2) \end{aligned}$$

where $\pi_1, \pi_2 \geq 0, \pi_1 + \pi_2 = 1$

- Jensen-Shannon(JS) divergence between a finite number of probability distributions:

$$\begin{aligned} JS_{\Pi}(\{p_1, \dots, p_n\}) &= \sum_i \pi_i KL(p_i, \pi_1 p_1 + \dots + \pi_n p_n) \\ &= H\left(\sum_i \pi_i p_i\right) - \sum_i \pi_i H(p_i) \end{aligned}$$

Information-Theoretic Clustering: (preserving mutual information)

- (Lemma) The loss in mutual information equals:

$$I(X, Y) - I(X, \hat{Y}) = \sum_{j=1}^k \pi(\hat{y}_j) JS_{\pi^*}(\{p(x | y_t) : y_t \in \hat{y}_j\})$$

- Interpretation: Quality of each cluster is measured by the Jensen-Shannon Divergence between the individual distributions in the cluster.
- Can rewrite the above as:

$$I(X, Y) - I(X, \hat{Y}) = \sum_{j=1}^k \sum_{y_t \in \hat{y}_j} \pi_t KL(p(x | y_t), p(x | \hat{y}_j))$$

- Goal: Find a clustering that minimizes the above loss

Information Theoretic Co-clustering (preserving mutual information)

- (Lemma) Loss in mutual information equals

$$\begin{aligned} I(X, Y) - I(\hat{X}, \hat{Y}) &= KL(p(x, y) \parallel q(x, y)) \\ &= H(\hat{X}, \hat{Y}) + H(X | \hat{X}) + H(Y | \hat{Y}) - H(X, Y) \end{aligned}$$

where

$$q(x, y) = p(\hat{x}, \hat{y})p(x | \hat{x})p(y | \hat{y}), \quad \text{where } x \in \hat{x}, y \in \hat{y}$$

- Can be shown that $q(x, y)$ is a “maximum entropy” *approximation* to $p(x, y)$.
- $q(x, y)$ preserves marginals : $q(x) = p(x)$ & $q(y) = p(y)$

$$p(x, y)$$

$$\begin{bmatrix} .05 & .05 & .05 & 0 & 0 & 0 \\ .05 & .05 & .05 & 0 & 0 & 0 \\ 0 & 0 & 0 & .05 & .05 & .05 \\ 0 & 0 & 0 & .05 & .05 & .05 \\ .04 & .04 & 0 & .04 & .04 & .04 \\ .04 & .04 & .04 & 0 & .04 & .04 \end{bmatrix}$$

$$\begin{bmatrix} .5 & 0 & 0 \\ .5 & 0 & 0 \\ 0 & .5 & 0 \\ 0 & .5 & 0 \\ 0 & 0 & .5 \\ 0 & 0 & .5 \end{bmatrix} \begin{bmatrix} .3 & 0 \\ 0 & .3 \\ .2 & .2 \end{bmatrix} \begin{bmatrix} .36 & .36 & .28 & 0 & 0 & 0 \\ 0 & 0 & 0 & .28 & .36 & .36 \end{bmatrix} = \begin{bmatrix} .054 & .054 & .042 & 0 & 0 & 0 \\ .054 & .054 & .042 & 0 & 0 & 0 \\ 0 & 0 & 0 & .042 & .054 & .054 \\ 0 & 0 & 0 & .042 & .054 & .054 \\ .036 & .036 & .028 & .028 & .036 & .036 \\ .036 & .036 & .028 & .028 & .036 & .036 \end{bmatrix}$$

$$p(x|\hat{x})$$

$$p(\hat{x}, \hat{y})$$

$$p(y|\hat{y})$$

$$q(x, y)$$

#parameters that determine q are: $(m-k)+(k-1)+(n-1)$

Preserving Mutual Information

- **Lemma:**

$$KL(p(x, y) \parallel q(x, y)) = \sum_{\hat{x}} \sum_{x \in \hat{x}} p(x) KL(p(y | x) \parallel q(y | \hat{x}))$$

$$\text{where } q(y | \hat{x}) = p(y | \hat{y}) p(\hat{y} | \hat{x}) = p(y | \hat{y}) \sum_{x \in \hat{x}} p(\hat{y} | x) p(x | \hat{x})$$

Note that $q(y | \hat{x})$ may be thought of as the “prototype” of row cluster \hat{x} (the usual “centroid” of the cluster is $\sum_{x \in \hat{x}} p(y | x) p(x | \hat{x})$)

$$\text{Similarly, } KL(p(x, y) \parallel q(x, y)) = \sum_{\hat{y}} \sum_{y \in \hat{y}} p(y) KL(p(x | y) \parallel q(x | \hat{y}))$$

Example – Continued

$q(y | \hat{x})$

.36	.36	.28	0	0	0
0	0	0	.28	.36	.36
.18	.18	.14	.14	.18	.18

.30	0
0	.30
.20	.20

$p(\hat{x}, \hat{y})$

.30	0
.30	0
0	.30
0	.30
.16	.24
.24	.16

$q(x | \hat{y})$

Co-Clustering Algorithm

Algorithm Co-Clustering(p, k, ℓ, C_X^t, C_Y^t)

Input: The joint probability distribution $p(X, Y)$, k the desired number of row clusters, and ℓ the desired number of column clusters.

Output: The partition functions C_X^t and C_Y^t .

1. Initialization: Set $t = 0$. Start with some initial partition functions $C_X^{(0)}$ and $C_Y^{(0)}$. Compute

$$q^{(0)}(\hat{X}, \hat{Y}), q^{(0)}(X|\hat{X}), q^{(0)}(Y|\hat{Y})$$

and the distributions $q^{(0)}(Y|\hat{b}), 1 \leq \hat{b} \leq k$ using (18).

2. Compute row clusters: For each row x , find its new cluster index as

$$C_X^{(t+1)}(x) = \operatorname{argmin}_{\hat{a}} D(p(Y|x) || q^{(t)}(Y|\hat{a})),$$

resolving ties arbitrarily. Let $C_Y^{(t+1)} = C_Y^{(t)}$.

3. Compute

$$q^{(t+1)}(\hat{X}, \hat{Y}), q^{(t+1)}(X|\hat{X}), q^{(t+1)}(Y|\hat{Y})$$

and the distributions $q^{(t+1)}(X|\hat{b}), 1 \leq \hat{b} \leq \ell$ using (19).

4. Compute column clusters: For each column y , find its new cluster index as

$$C_Y^{(t+2)}(y) = \operatorname{argmin}_{\hat{b}} D(p(X|y) || q^{(t+1)}(X|\hat{b})),$$

resolving ties arbitrarily. Let $C_X^{(t+2)} = C_X^{(t+1)}$.

5. Compute

$$q^{(t+2)}(\hat{X}, \hat{Y}), q^{(t+2)}(X|\hat{X}), q^{(t+2)}(Y|\hat{Y})$$

and distributions $q^{(t+2)}(Y|\hat{b}), 1 \leq \hat{b} \leq k$ using (18).

6. Stop and return $C_X^t = C_X^{(t+2)}$ and $C_Y^t = C_Y^{(t+2)}$, if the change in objective function value, that is, $D(p(X, Y) || q^{(t)}(X, Y)) - D(p(X, Y) || q^{(t+2)}(X, Y))$, is "small" (say 10^{-5}); Else set $t = t + 2$ and go to step 2.

Figure 1: Information theoretic co-clustering algorithm that simultaneously clusters both the rows and columns

Properties of Co-clustering Algorithm

- **Theorem:** The co-clustering algorithm monotonically decreases loss in mutual information (objective function value)
 - Marginals $p(x)$ and $p(y)$ are preserved at every step ($q(x)=p(x)$ and $q(y)=p(y)$)
 - Can be generalized to higher dimensions
-

	$g^{(0)}(X, Y)$						$p^{(0)}(\bar{X}, \bar{Y})$	
	\hat{y}_1	\hat{y}_1	\hat{y}_2	\hat{y}_1	\hat{y}_2	\hat{y}_2		
\hat{z}_3	.029	.029	.019	.022	.024	.024	0.10	0.05
\hat{z}_1	.036	.036	.014	.028	.018	.018	0.10	0.20
\hat{z}_2	.018	.018	.028	.014	.036	.036	0.30	0.25
\hat{z}_2	.018	.018	.028	.014	.036	.036		
\hat{z}_3	.039	.039	.025	.030	.032	.032		
\hat{z}_3	.039	.039	.025	.030	.032	.032		

↓ steps 2 & 3 of Figure 1

	\hat{y}_1	\hat{y}_1	\hat{y}_2	\hat{y}_1	\hat{y}_2	\hat{y}_2		
\hat{z}_1	.036	.036	.014	.028	.018	.018	0.20	0.10
\hat{z}_1	.036	.036	.014	.028	.018	.018	0.18	0.32
\hat{z}_2	.019	.019	.026	.015	.034	.034	0.12	0.08
\hat{z}_2	.019	.019	.026	.015	.034	.034		
\hat{z}_3	.043	.043	.022	.033	.028	.028		
\hat{z}_2	.025	.025	.035	.020	.046	.046		

↓ steps 4 & 5 of Figure 1

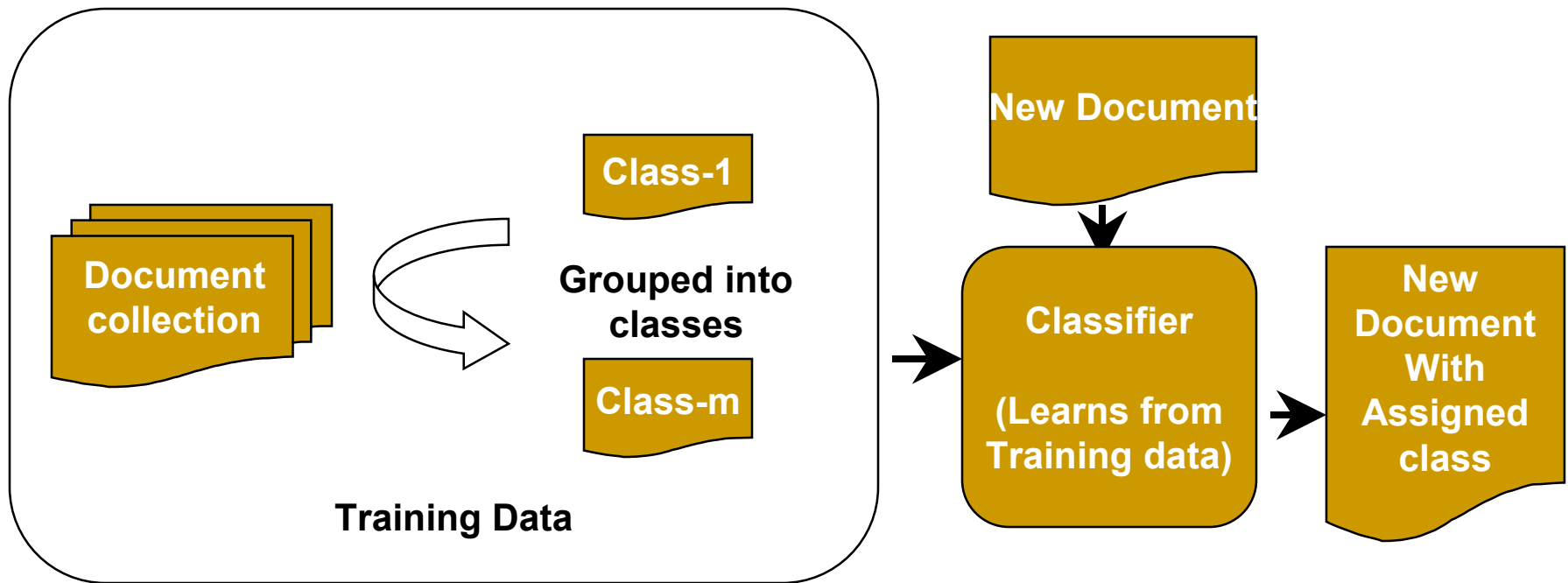
	\hat{y}_1	\hat{y}_1	\hat{y}_1	\hat{y}_2	\hat{y}_2	\hat{y}_2		
\hat{z}_1	.054	.054	.042	0	0	0	0.30	0
\hat{z}_1	.054	.054	.042	0	0	0	0.12	0.38
\hat{z}_2	.013	.013	.010	.031	.041	.041	0.08	0.12
\hat{z}_2	.013	.013	.010	.031	.041	.041		
\hat{z}_3	.028	.028	.022	.033	.043	.043		
\hat{z}_2	.017	.017	.013	.042	.054	.054		

↓ steps 2 & 3 of Figure 1

	\hat{y}_1	\hat{y}_1	\hat{y}_1	\hat{y}_2	\hat{y}_2	\hat{y}_2		
\hat{z}_1	.054	.054	.042	0	0	0	0.30	0
\hat{z}_1	.054	.054	.042	0	0	0	0	0.30
\hat{z}_2	0	0	0	.042	.054	.054	0.20	0.20
\hat{z}_2	0	0	0	.042	.054	.054		
\hat{z}_3	.036	.036	.028	.028	.036	.036		
\hat{z}_3	.036	.036	.028	.028	.036	.036		

Applications -- Text Classification

- Assigning class labels to text documents
- Training and Testing Phases

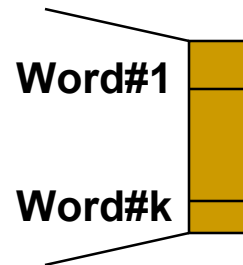
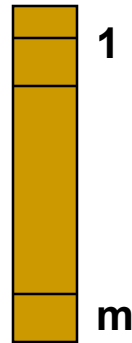


Dimensionality Reduction

■ Feature Selection

Document
Bag-of-words

Vector
Of
words

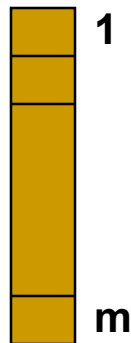


- Select the “best” words
- Throw away rest
- Frequency based pruning
- Information criterion based pruning

■ Feature Clustering

Document
Bag-of-words

Vector
Of
words



- Do *not* throw away words
- Cluster words instead
- Use clusters as features

Experiments

- Data sets
 - 20 Newsgroups data
 - 20 classes, 20000 documents
 - Classic3 data set
 - 3 classes (cisi, med and cran), 3893 documents
 - Dmoz Science HTML data
 - 49 leaves in the hierarchy
 - 5000 documents with 14538 words
 - Available at <http://www.cs.utexas.edu/users/manyam/dmoz.txt>
 - Implementation Details
 - Bow – for indexing, co-clustering, clustering and classifying
-

Naïve Bayes with word clusters

- Naïve Bayes classifier

- Assign document d to the class with the highest score

$$c^*(d) = \operatorname{argmax}_i (\log(p(c_i)) + \sum_{t=1}^v p(w_t | d) \log(p(w_t | c_i)))$$

- Relation to KL Divergence

$$c^*(d) = \operatorname{arg min}_i (KL(p(W | d), p(W | c_i)) - \log p(c_i))$$

- Using word clusters instead of words

$$c^*(d) = \operatorname{argmax}_i (\log(p(c_i)) + \sum_{s=1}^k p(\hat{x}_s | d) \log(p(\hat{x}_s | c_i)))$$

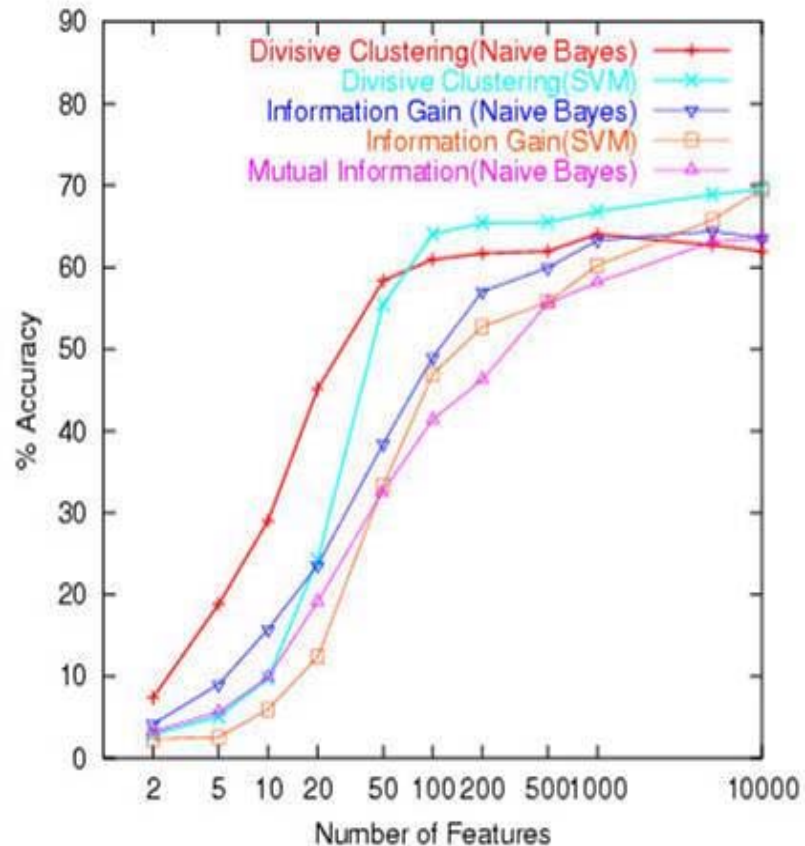
where parameters for clusters are estimated according to joint statistics

Results (20Ng)

- Classification Accuracy on 20 Newsgroups data with 1/3-2/3 test-train split
 - Clustering beats feature selection algorithms by a large margin
 - The effect is more significant at lower number of features
-

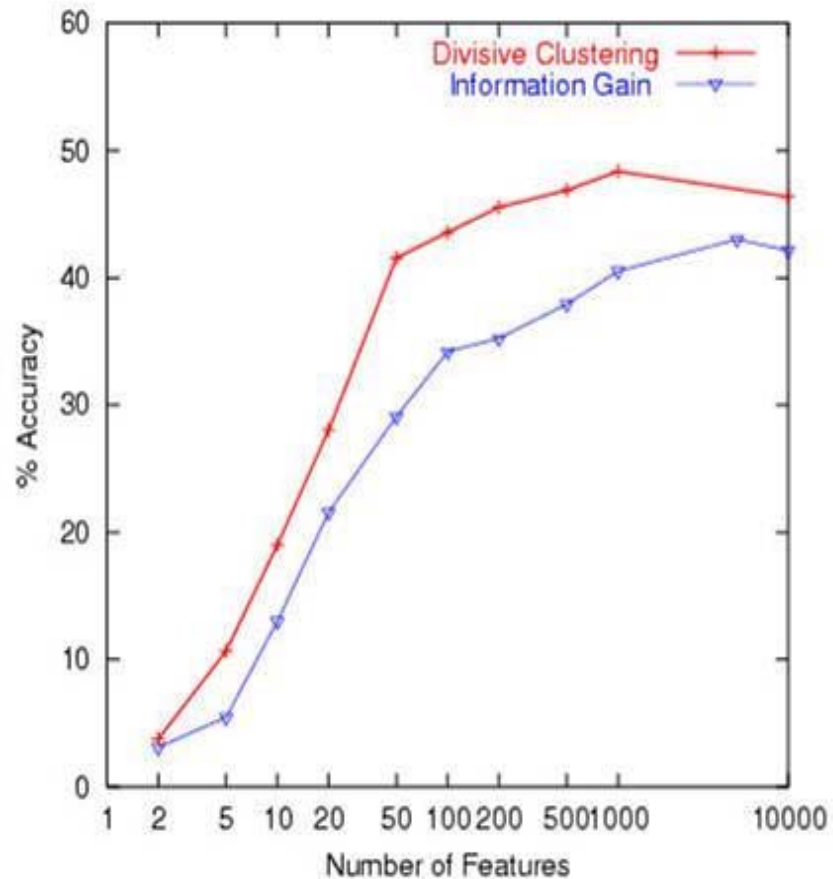
Results (Dmoz)

- Classification Accuracy on Dmoz data with 1/3-2/3 test train split
- Divisive Clustering is better at lower number of features
- Note contrasting behavior of Naïve Bayes and SVMs

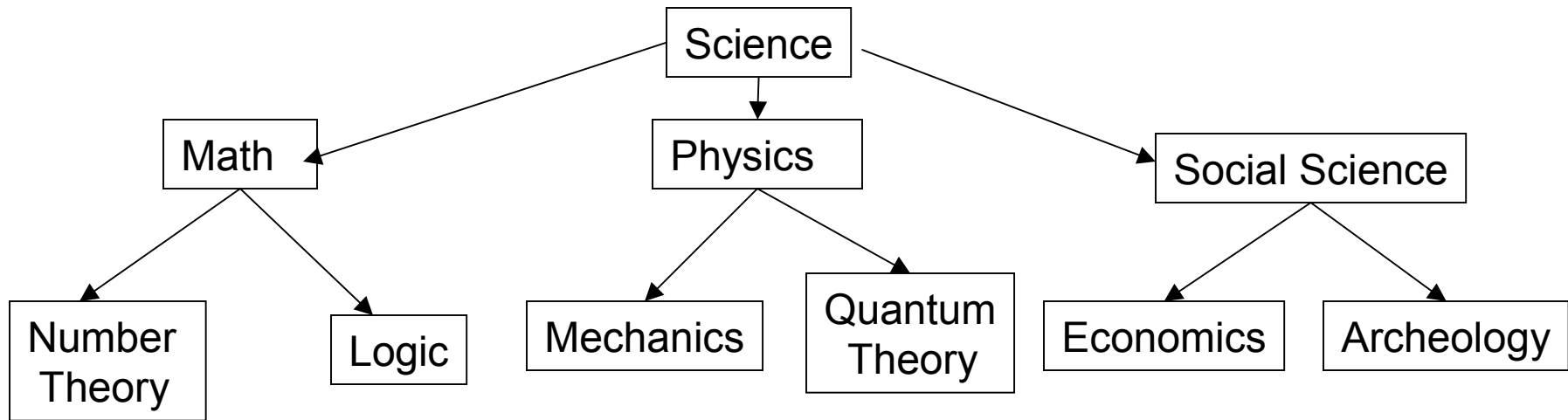


Results (Dmoz)

- Naïve Bayes on Dmoz data with only 2% Training data
- Note that Divisive Clustering achieves higher maximum than IG with a significant 13% increase
- Divisive Clustering performs better than IG at lower training data



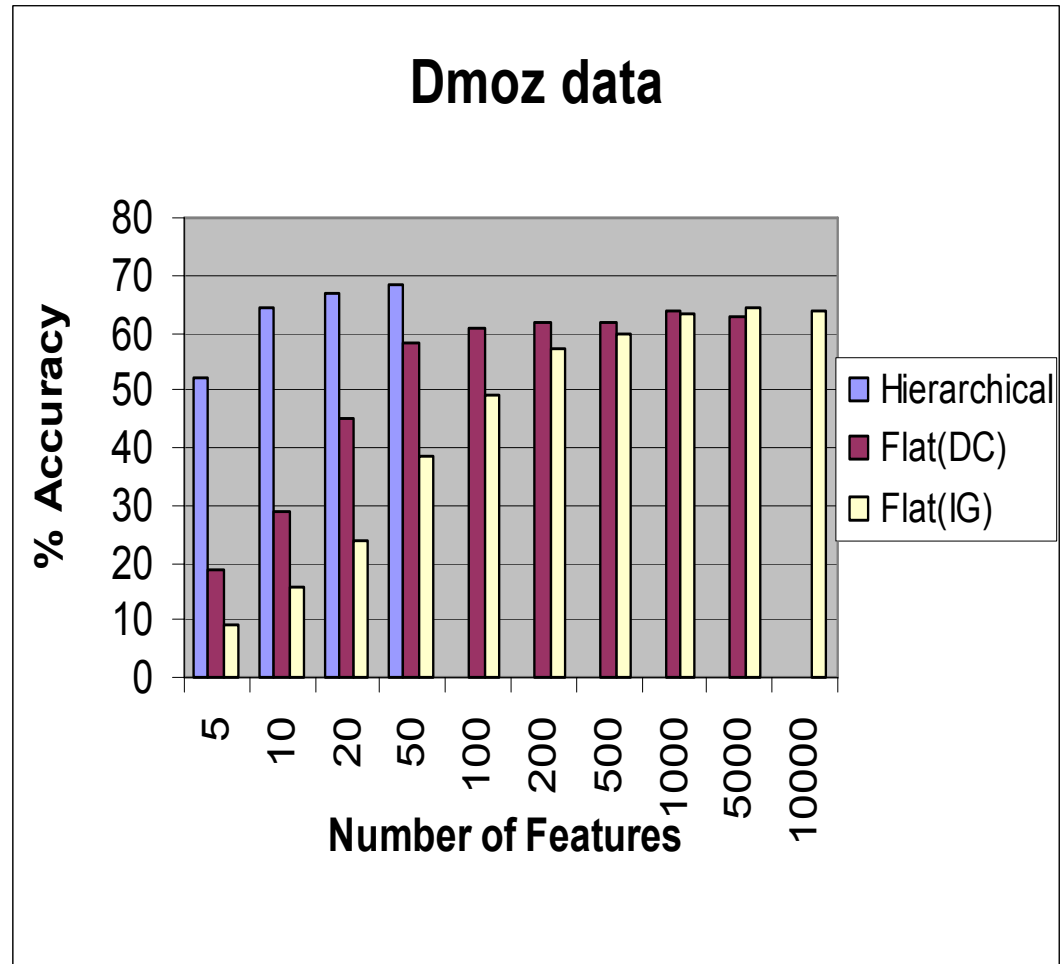
Hierarchical Classification



- Flat classifier builds a classifier over the leaf classes in the above hierarchy
 - Hierarchical Classifier builds a classifier at each internal node of the hierarchy
-

Results (Dmoz)

- Hierarchical Classifier (Naïve Bayes at each node)
- Hierarchical Classifier: 64.54% accuracy at just 10 features (Flat achieves 64.04% accuracy at 1000 features)
- Hierarchical Classifier improves accuracy to 68.42% from 64.42%(maximum) achieved by flat classifiers



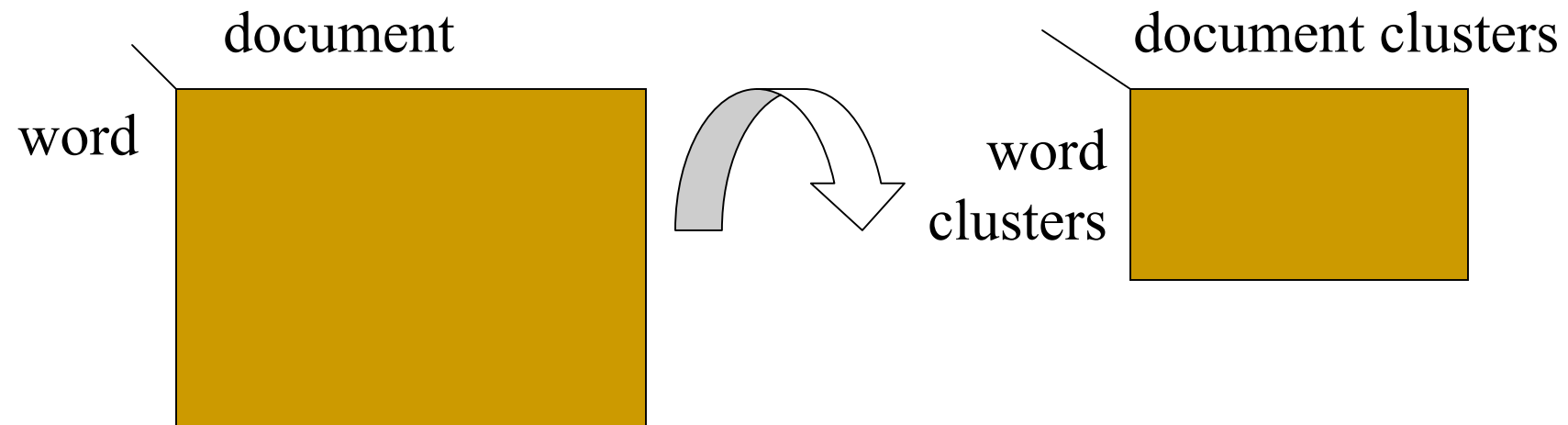
Example

Cluster 10 Divisive Clustering <i>(rec.sport.hockey)</i>	Cluster 9 Divisive Clustering <i>(rec.sport.baseball)</i>	Cluster 12 Agglomerative Clustering <i>(rec.sport.hockey and rec.sport.baseball)</i>
team game play hockey Season boston chicago pit van nhl	hit runs Baseball base Ball greg morris Ted Pitcher Hitting	team detroit hockey pitching Games hitter Players rangers baseball nyi league morris player blues nhl shots Pit Vancouver buffalo ens

Top few words sorted in Clusters obtained by Divisive and Agglomerative approaches on 20 Newsgroups data

Co-clustering Example for Text Data

- Co-clustering clusters both words and documents *simultaneously* using the underlying co-occurrence frequency matrix



Results– CLASSIC3

Co-Clustering (0.9835)			1-D Clustering (0.821)		
992	4	8	847	142	44
40	1452	7	41	954	405
1	4	1387	275	86	1099

Results – Sparsity

Results – continued

Results (Monotonicity)

Related Work

- **Distributional Clustering**
 - Pereira, Tishby & Lee (1993)
 - Baker & McCallum (1998)
 - **Information Bottleneck**
 - Tishby, Pereira & Bialek (1999)
 - Berkhin & Becher (2002)
 - **Probabilistic Latent Semantic Indexing**
 - Hofmann (1999)
 - **Non-Negative Matrix Approximation**
 - Lee & Seung (2000)
-

Conclusions

- Information theoretic approaches to clustering, co-clustering
 - Co-clustering problem is tied to a non-negative matrix approximation
 - Requires estimation of fewer number of parameters
 - Can be extended to the more general class of Bregman divergences
 - KL-divergence, squared Euclidean distances are special cases
 - Theoretical approach has the potential of extending to other problems:
 - incorporating unlabelled data
 - multi-dimensional co-clustering
 - MDL to choose number of clusters
-

Contact Information

- Email: inderjit@cs.utexas.edu
 - Papers are available at:
<http://www.cs.utexas.edu/users/inderjit>
 - “Divisive Information-Theoretic Feature Clustering for Text Classification”, Dhillon, Mallela & Kumar, *Journal of Machine Learning Research (JMLR)*, March, 2003 (also see *KDD*, 2002)
 - “Information-Theoretic Co-clustering”, Dhillon, Mallela & Modha, To appear in *KDD*, 2003 (also *UTCS Technical Report*).
 - “Clustering with Bregman Divergences”, Banerjee, Merugu, Dhillon & Ghosh, *UTCS Technical Report*, 2003
-