MDL-Based Unsupervised Attribute Ranking

Zdravko Markov
Computer Science Department
Central Connecticut State University
New Britain, CT 06050, USA
http://www.cs.ccsu.edu/~markov/
markovz@ccsu.edu

MDL-Based Unsupervised Attribute Ranking

- Introduction (Attribute Selection)
- MDL-based Clustering Model Evaluation
- Illustrative Example (“play tennis” data)
- Attribute Ranking Algorithm
- Hierarchical Clustering Algortihm
- Experimental Evaluation
- Conclusion

Attribute Selection

- Supervised / Unsupervised. Find the smallest set of attributes that
  - maximizes predictive accuracy
  - best uncovers interesting natural groupings (clusters) in data according to the chosen criterion
- Subset Selection / Ranking (Weighting)
  - Computationally expensive: 2^m attribute sets for m attributes
  - Assumes that attributes are independent

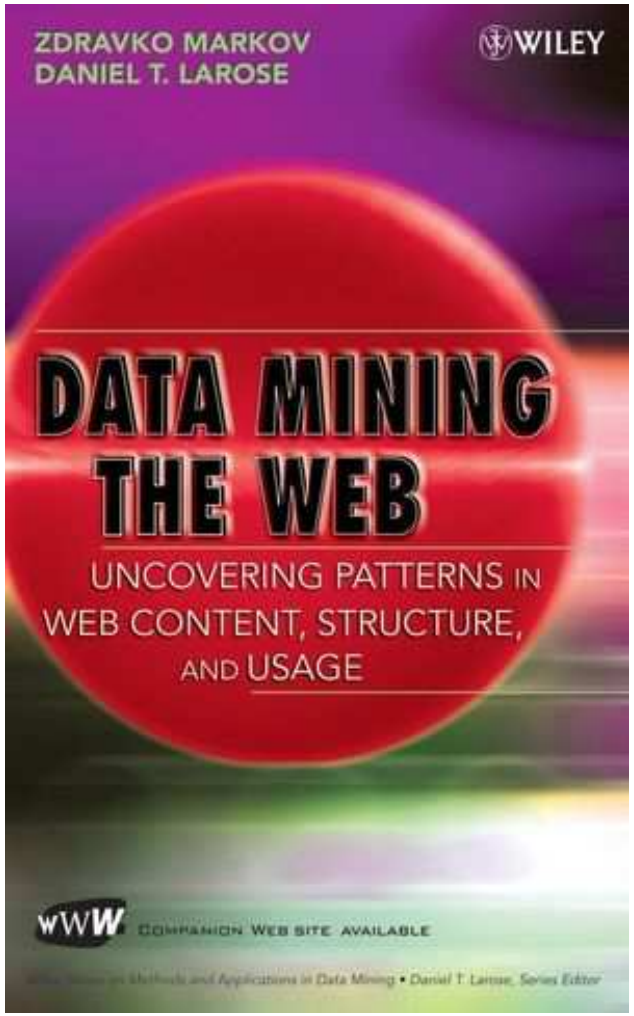
Supervised Attribute Selection

- **Wrapper methods** create prediction models and use the predictive accuracy of these models to measure the attribute relevance to the classification task.
- **Filter methods** directly measure the ability of the attributes to determine the class labels using statistical correlation, information metrics, probabilistic or other methods.
- **There exist numerous methods** in this setting due to the wide availability of model evaluation criteria in supervised learning.

Unsupervised Attribute Selection

- **Wrapper methods** evaluate a subset of attributes by the quality of clustering obtained by using these attributes.
- **Filter methods** explore classical statistical methods for dimensionality reduction, like PCA and maximum variance, information-based or entropy measures.
- There exist **very few methods** in this setting generally because of the difficulty to evaluate clustering models.

Clustering Model Evaluation



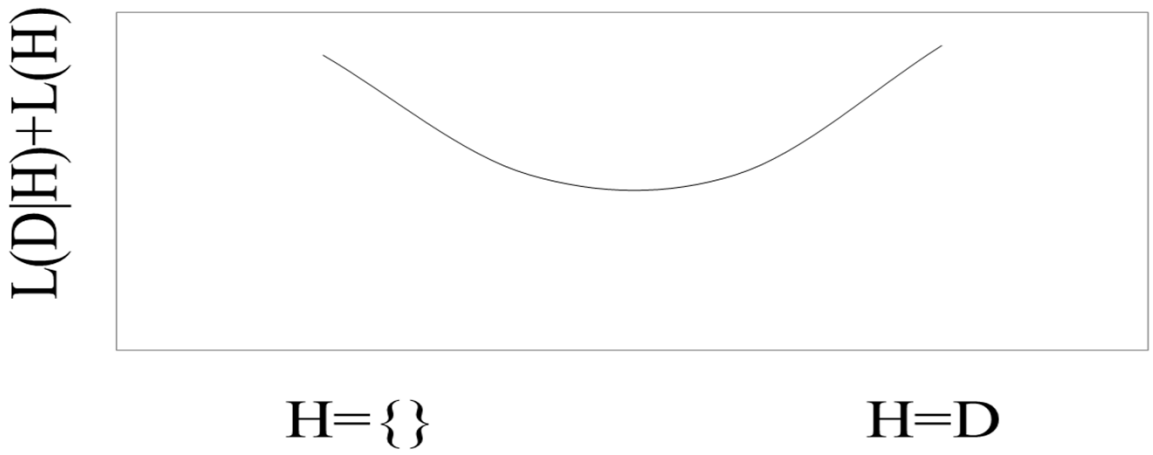
Chapter 4: Evaluating Clustering
- MDL-Based Model and Feature Evaluation

http://www.cs.ccsu.edu/~markov/
http://www.cs.ccsu.edu/~markov/dmw4.pdf
http://www.cs.ccsu.edu/~markov/dmwdata.zip
http://www.cs.ccsu.edu/~markov/DMWsoftware.zip

Clustering Model Evaluation

- Consider each possible clustering as a *hypothesis H* that *describes* (*explains*) data *D* in terms of frequent patterns (regularities).
- Compute the *description length* of the data $L(D)$, the hypothesis $L(H)$, and data given the hypothesis $L(D|H)$.
- $L(H)$ and $L(D)$ are the minimum number of bits needed to encode (or communicate) *H* and *D* respectively.
- $L(D|H)$ represents the number of bits needed to encode *D* if we know *H*.
- If we know the pattern of H, no need to encode all its occurrences in D, rather we may encode only the pattern itself and the differences that identify each individual instance in *D*.

Minimum Description Length (MDL) and Information Compression



- The more regularity in D the shorter description length $L(D|H)$.
- Need to balance $L(D|H)$ with $L(H)$, because the latter depends on the complexity of the pattern. Thus **the best hypothesis** should
  - minimize the sum $L(H)+L(D|H)$ (***MDL principle***)
  - or maximize $L(D) - L(H) - L(D|H)$ (***Information Compression***)

Encoding MDL

- Hypotheses and data are uniformly distributed and the probability of occurrence of an item out of n alternatives is $1/n$.
- Minimum code length of the message that a particular item has occurred is $-log_2\ 1/n = log_2\ n$ bits.
- The number of bits needed to encode the choice of k items out of n possible items is

$$-\log_2 \frac{1}{\binom{n}{k}} = \log_2 \binom{n}{k}$$

Encoding MDL (attribute-value)

- Data D, instance $X \in D$, X is a set of m attribute values, $|X| = m$
- $T = \bigcup_{X \in D} X$ - set of all attribute values in D, $k = |T|$
- Cluster C_i is defined by the set of all attribute values $T_i \subseteq T$ that occur in its members, $C_i = \{X \in C_i, X \subseteq T_i\}$
- Clustering $H = \{C_1, C_2, \ldots, C_n\}$ is defined by $\{T_1, T_2, \ldots, T_n\}$, $k_i = |T_i|$

$$L(C_i) = \log_2 \binom{k}{k_i} + \log_2 n \qquad L(H) = \sum_{i=1}^{n} L(C_i)$$

$$L(D_i \mid C_i) = |C_i| \times \log_2 \binom{k_i}{m} \qquad L(D \mid H) = \sum_{i=1}^{n} L(D_i \mid C_i)$$

$$MDL(C_i) = \log_2 \binom{k}{k_i} + \log_2 n + |C_i| \times \log_2 \binom{k_i}{m} \qquad MDL(H) = \sum_{i=1}^{n} MDL(C_i)$$

Play Tennis Data

ID	outlook	temp	humidity	windy	play
1	sunny	hot	high	false	no
2	sunny	hot	high	true	no
3	overcast	hot	high	false	yes
4	rainy	mild	high	false	yes
5	rainy	cool	normal	false	yes
6	rainy	cool	normal	true	no
7	overcast	cool	normal	true	yes
8	sunny	mild	high	false	no
9	sunny	cool	normal	false	yes
10	rainy	mild	normal	false	yes
11	sunny	mild	normal	true	yes
12	overcast	mild	high	true	yes
13	overcast	hot	normal	false	yes
14	rainy	mild	high	true	no

C_1 = {1, 2, 3, 4, 8, 12, 14} (humidity=high)

C_2 = {5, 6, 7, 9, 10, 11, 13} (humidity=normal)

T_1 = {outlook=sunny, outlook=overcast, outlook=rainy, temp=hot, temp=mild, humidity=high, windy=false, windy=true}

T_2 = {outlook=sunny, outlook=overcast, outlook=rainy, temp=hot, temp=mild, temp=cool, humidity=normal, windy=false, windy=true}.

Clustering Play Tennis Data

$$MDL(C_i) = \log_2 \binom{k}{k_i} + \log_2 n + |C_i| \times \log_2 \binom{k_i}{m}$$

$k_1 = |T_1| = 8$, $k_2 = |T_2| = 9$, $k = 10$, $m = 4$, $n = 2$

$$MDL(C_1) = \log_2 \binom{10}{8} + \log_2 2 + 7 \times \log_2 \binom{8}{4} = 49.39$$

$$MDL(C_2) = \log_2 \binom{10}{9} + \log_2 2 + 7 \times \log_2 \binom{9}{4} = 53.16$$

$MDL(\{C_1, C_2\})$ = MDL(humidity) = 102.55 bits

1. MDL(temp) = 101.87
2. MDL(humidity) = 102.56
3. MDL(outlook) = 103.46
4. MDL(windy) = 106.33

- Best attribute is **temp**

MDL Ranker

- Let A have values $v_1, v_2, \ldots, v_p$
- Clustering $\{C_1, C_2, \ldots, C_p\}$, where $C_i = \{X \mid x_i \in X\}$
- Let $V_i^A = \varnothing$
- For each data instance $X = \{x_1, x_2, \ldots, x_m\}$
- For each attribute A
- For each value x_i
- $V_i^A = V_i^A \cup \{x_i\}$
- $k_i = \sum_{j=1}^{m} \left| V_j^A \right|$
- Compute $MDL(\{C_1, C_2, \ldots, C_p\})$

- Incremental (no need to store instances)
- Time $O(nm^2)$, n is the number of data instances
- Space $O(pm^2)$, p is the max number of attribute values
- Evaluates 3204 instances with 13195 attributes (trec data) in 3 minutes.

Experimental Evaluation Data

Data Set	Instances	Attributes	Classes
reuters	1504	2887	13
reuters-3class	1146	2887	3
reuters-2class	927	2887	2
trec	3204	13195	6
soybean	683	36	19
soybean-small	47	36	4
iris	150	5	3
ionosphere	351	35	2

Java implementations of MDL ranking and clustering available from http://www.cs.ccsu.edu/~markov/DMWsoftware.zip

Experimental Evaluation Metrics

- $$\text{Average Precision} = \frac{1}{|D_q|}\sum_{k=1}^{|D|} r_k \times \text{PrecisionAtRank(k)}$$

$$\text{PrecisionAtRank(k)} = \frac{1}{k}\sum_{i=1}^{k} r_i \qquad r_i = \begin{cases} 1 & \text{if } a_i \in D_q \\ 0 & \text{otherwise} \end{cases}$$

- Classes-to-clusters accuracy (“true” cluster membership)

```
root [5, 9]
  temperature=hot [2, 2]
    outlook=sunny [2] no
    outlook=overcast [2] yes
  temperature=mild [4, 2]
    windy=FALSE [2, 1] yes
    windy=TRUE [2, 1] yes
  temperature=cool [3, 1]
    windy=FALSE [2] yes
    windy=TRUE [1, 1] no
------------------------------------------
Clusters (leaves): 6
Correctly classified instances: 11 (78%)
```

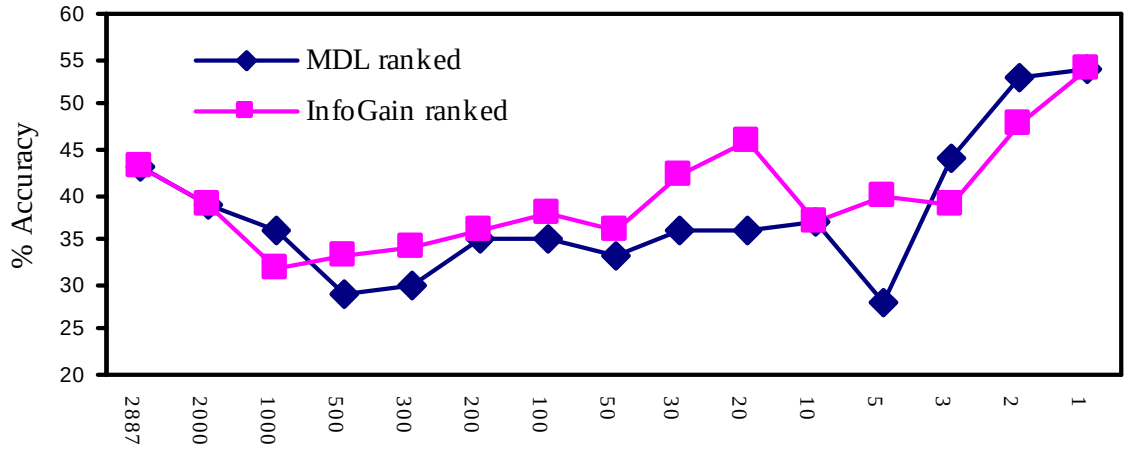
Average Precision of Attribute Ranking

Data set	$\mid D_q \mid$	**InfoGain**	**MDL**	**Error**	**Entropy**
reuters	15	0.3183	0.1435	0.0642	0.0030
reuters-3class	10	0.3948	0.1852	0.1257	0.0027
reuters-2class	7	0.5016	0.2438	0.1788	0.3073
trec	14	0.4890	0.2144	0.0637	0.0010
soybean	16	0.6265	0.5606	0.3871	0.4152
soybean-small	2	0.6428	0.3500	0.0913	0.1213
iris	1	1.0000	1.0000	1.0000	0.3333
ionosphere	9	0.6596	0.5041	0.2575	0.4252

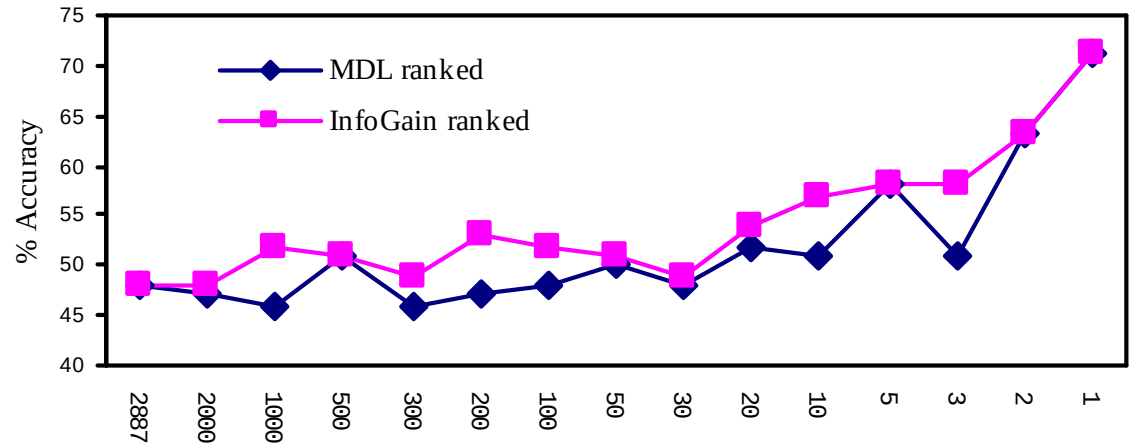
$\boldsymbol{D_q}$ – set of attributes selected by Wrapper Subset Evaluator with Naïve Bayes classifier.
InfoGain – supervised attribute ranking using Information Gain Evaluator.
Error – unsupervised ranking based on evaluating the quality of clustering by the *sum of squared errors*.
Entropy – unsupervised ranking based on the reduction of the entropy in data when the attribute is removed (Dash and Liu 2000).

Classes-To-Clusters Accuracy With Reuters Data

EM

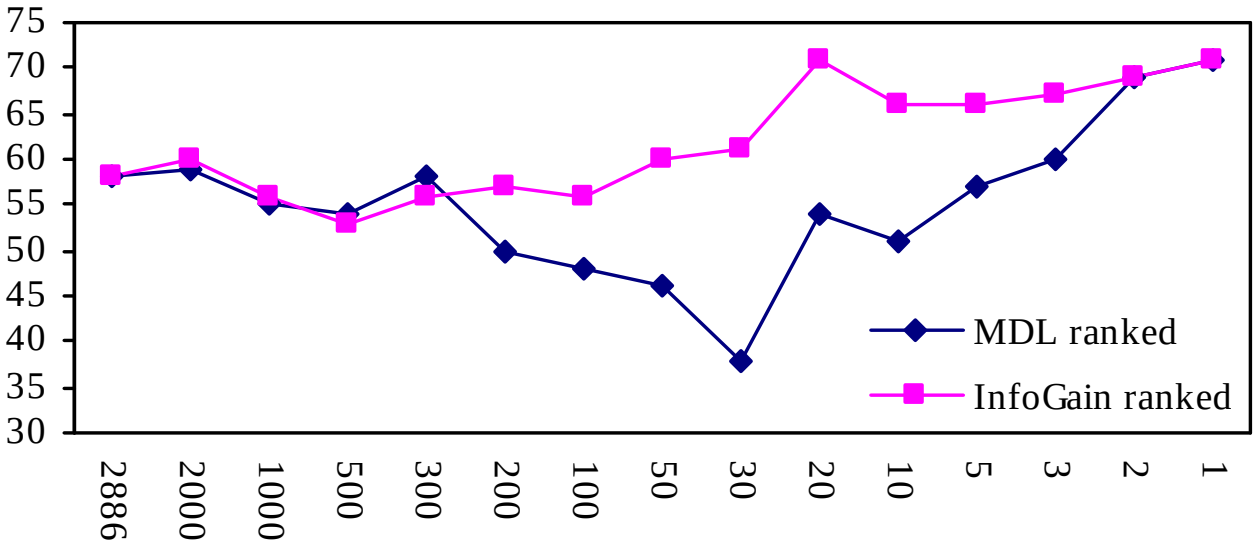


k-means

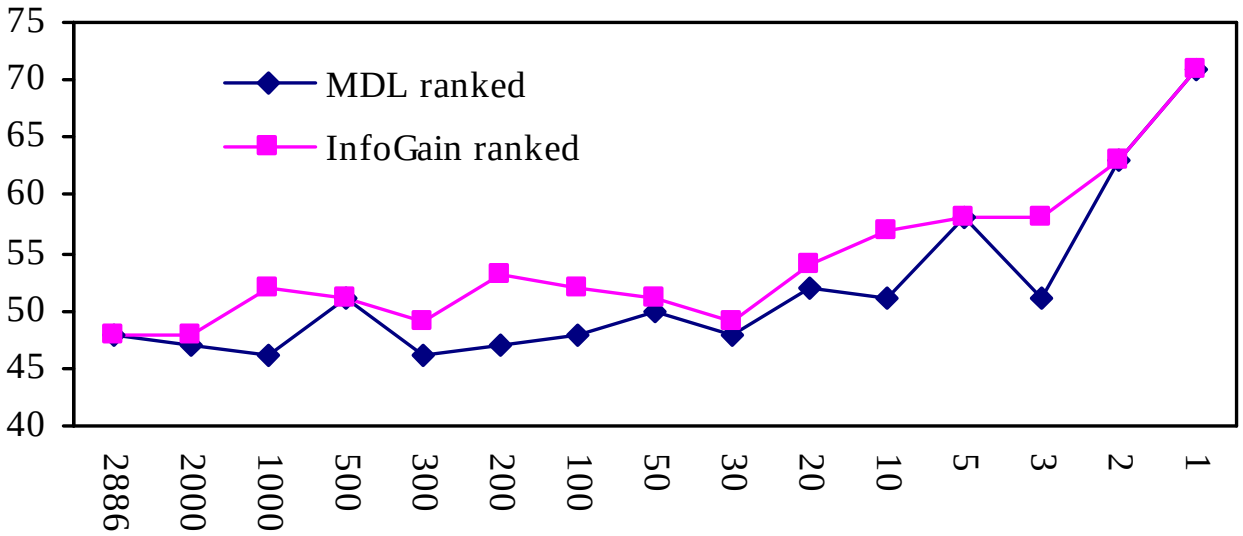


Classes-To-Clusters Accuracy With Reuters-3class Data

EM

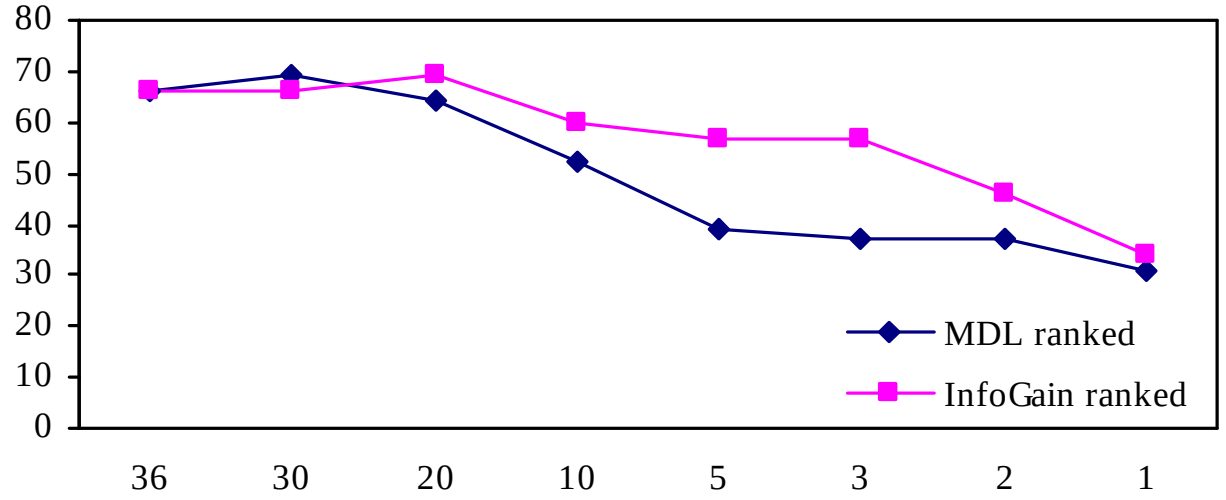


K-means

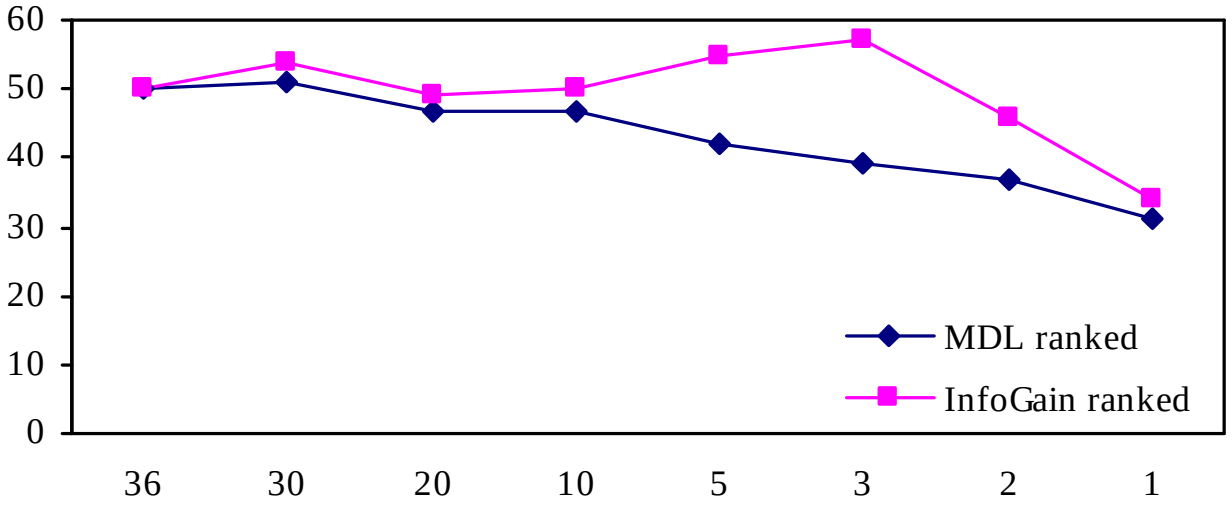


Classes-To-Clusters Accuracy With Soybean Data

EM



k-means



MDL-Based Clustering

Function *MDL-Cluster(D)*

1. Choose attribute $A = \arg\min_i MDL(A_i)$
2. Let A take values $v_1, v_2, \ldots, v_p$
3. Split data $D = \bigcup_{i=1}^{n} C_i$, $C_i = \{X \mid x_i \in X\}$
4. If $Comp(A) > \sum_{i=1}^{p} Comp(C_i)$ then stop. Return D.
5. For each $i = 1,...,n$ *Call MDL-Cluster(C_i)*

Clustering Reuters-2class Data

root (516550.58) [608, 319]
  trade=0 (**434956.39**) [507, 18]
    *rate=0 (**266126.68**) [339, 18]*
      monei=1 (122154.60) [148] money
      monei=0 (161236.34) [191, 18] money
    *rate=1 (**125589.70**) [168]*
      currenc=0 (70870.68) [100] money
      currenc=1 (50491.67) [68] money
  trade=1 (204850.37) [301, 101]
    market=0 (157978.80) [186, 39]
      countri=1 (64418.90) [67, 20] trade
      countri=0 (106457.20) [119, 19] trade
    market=1 (106422.43) [115, 62]
      bank=0 (73572.74) [94, 11] trade
      bank=1 (48489.70) [21, 51] money

Clusters (leaves): 8
Correctly classified instances: 838 (90%)

MDL-Cluster Tree:

root (516550.58) [608, 319]
  trade=0 (434956.39) [507, 18] money
  trade=1 (204850.37) [301, 101]
    market=0 (157978.80) [186, 39]
      countri=1 (64418.90) [67, 20] trade
      countri=0 (106457.20) [119, 19] trade
    market=1 (106422.43) [115, 62]
      bank=0 (73572.74) [94, 11] trade
      bank=1 (48489.70) [21, 51] money

Clusters (leaves): 5
Correctly classified instances: 838 (90%)

Comparing MDL, EM and k-Means

Data set	EM		k-Means		MDL-Cluster	
	Acc. %	**No. of Clusters**	**Acc. %**	**No. of Clusters**	**Acc. %**	**No. of Clusters**
reuters	43	6	31	13	59	12
reuters-3class	58	3	48	3	73	7
reuters-2class	71	2	61	2	90	7
trec	26	6	29	6	44	11
soybean	60	19	51	19	51	7
soybean-small	100	4	91	4	83	4
iris	95	3	69	3	96	3
ionosphere	89	2	81	2	80	3

Conclusion

- MDL-ranker without class information performs closely to the InfoGain method, which essentially uses class information.
- Thus, our approach can improve the performance of clustering algorithms in purely unsupervised setting.
- MDL-cluster outperforms EM and k-means on most benchmark data sets.
- Numeric attributes ?
- Subset evaluation ?
- Non-hierarchical clustering ?

Thank You!