Clustering Categorical Data by CD Vectors

Xiaogang(Steven) Wang

Dept. of Math. and Stat.
York University

Joint work with Peng Zhang and
Peter X. –K.Song, University of Waterloo

Presentation Outline

- Review of current literature
- Hamming Distance and CD vector
- Modified Chi-square test
- Description of our Algorithm
- Numerical Results
- Conclusion and Discussions

Review of existing algorithms

K-modes

1. This algorithm is built on the idea of K-means algorithm.
2. It demands the number of clusters.
3. Partition is sensitive to the input order.
4. Computational Complexity O(n)

AutoClass Algorithm

This algorithm can cluster both categorical and numeric data types.

1. It utilizes the EM algorithm.
2. It searches the optimal number of clusters
3. EM algorithm is known to have slow convergence.
4. The computational complexity is O(n).

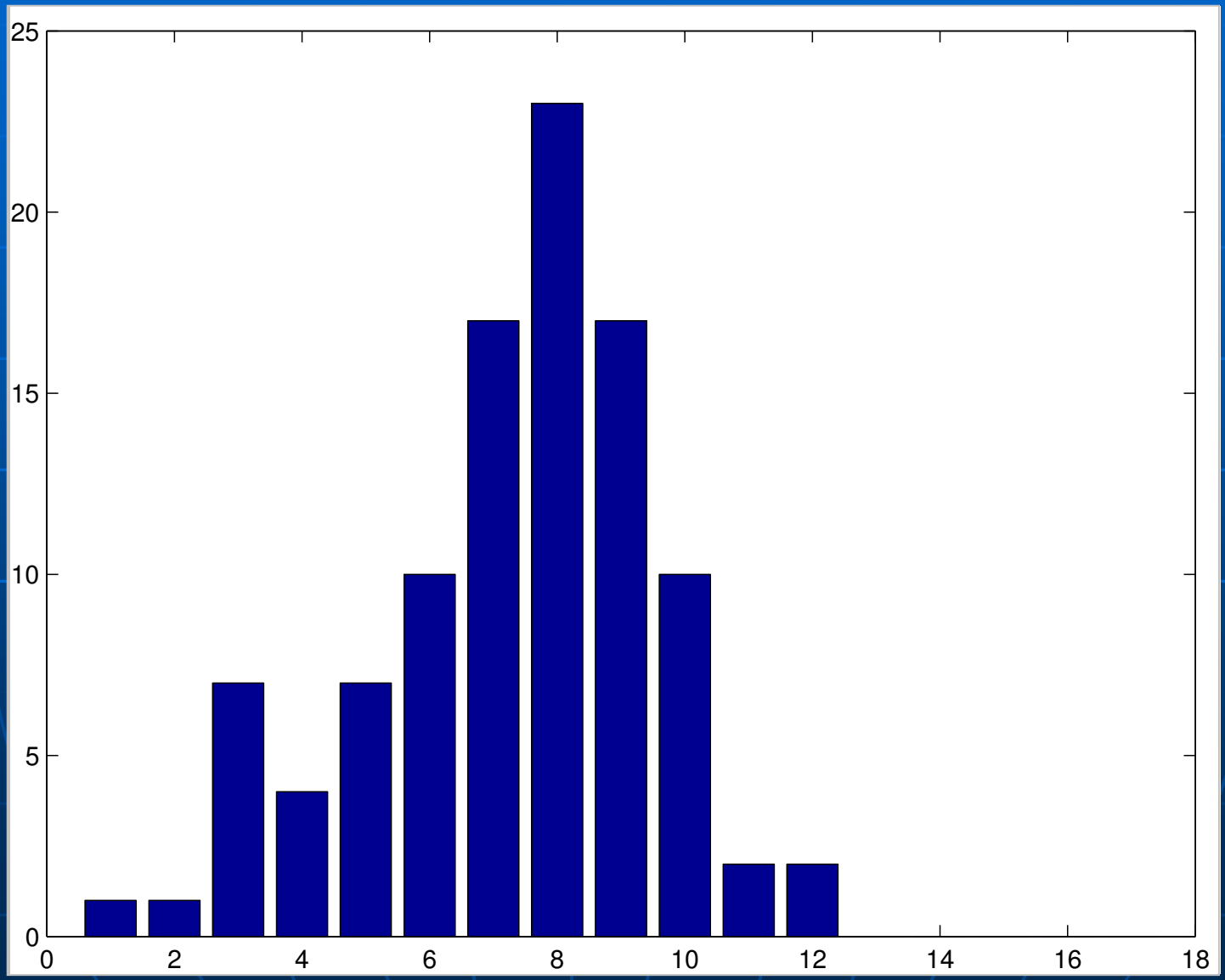
Categorical Sample Space

- Assume that the data set is stored in a ***n*p*** matrix, where ***n*** is the number of observations and ***p*** the number of categorical variables.
- The sample space consists of all possible combinations generated by ***p*** variables.
- The sample space is discrete and has no natural origin.

Hamming Distance and CD vector

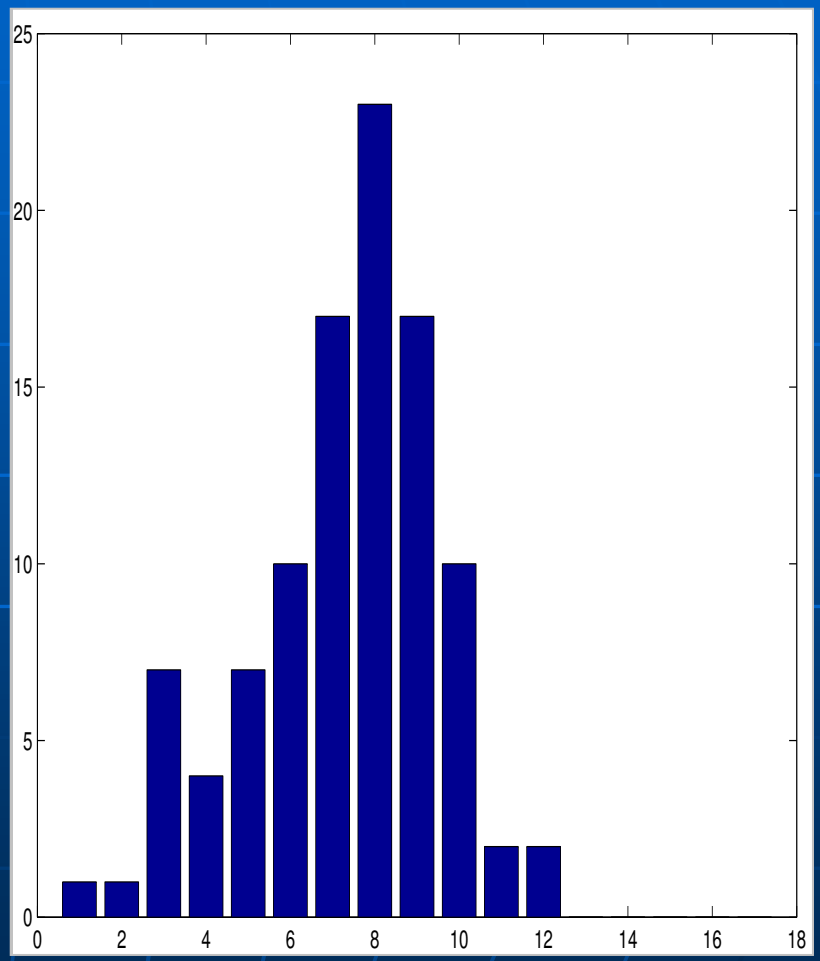
- Hamming distance measures the number of different attributes between two categorical variables.
- Hamming Distance has been used in clustering categorical data in algorithms similar to K-modes.
- We construct Categorical Distance (CD) vector to project the sample space into 1-dimesional space.

Example of a CD vector



More on CD vector

- The dense region of the CD vector is necessarily a cluster!
- The length of the CD vector is ***p***.
- *We can construct many CD vectors on one data set by choosing different "origin".*



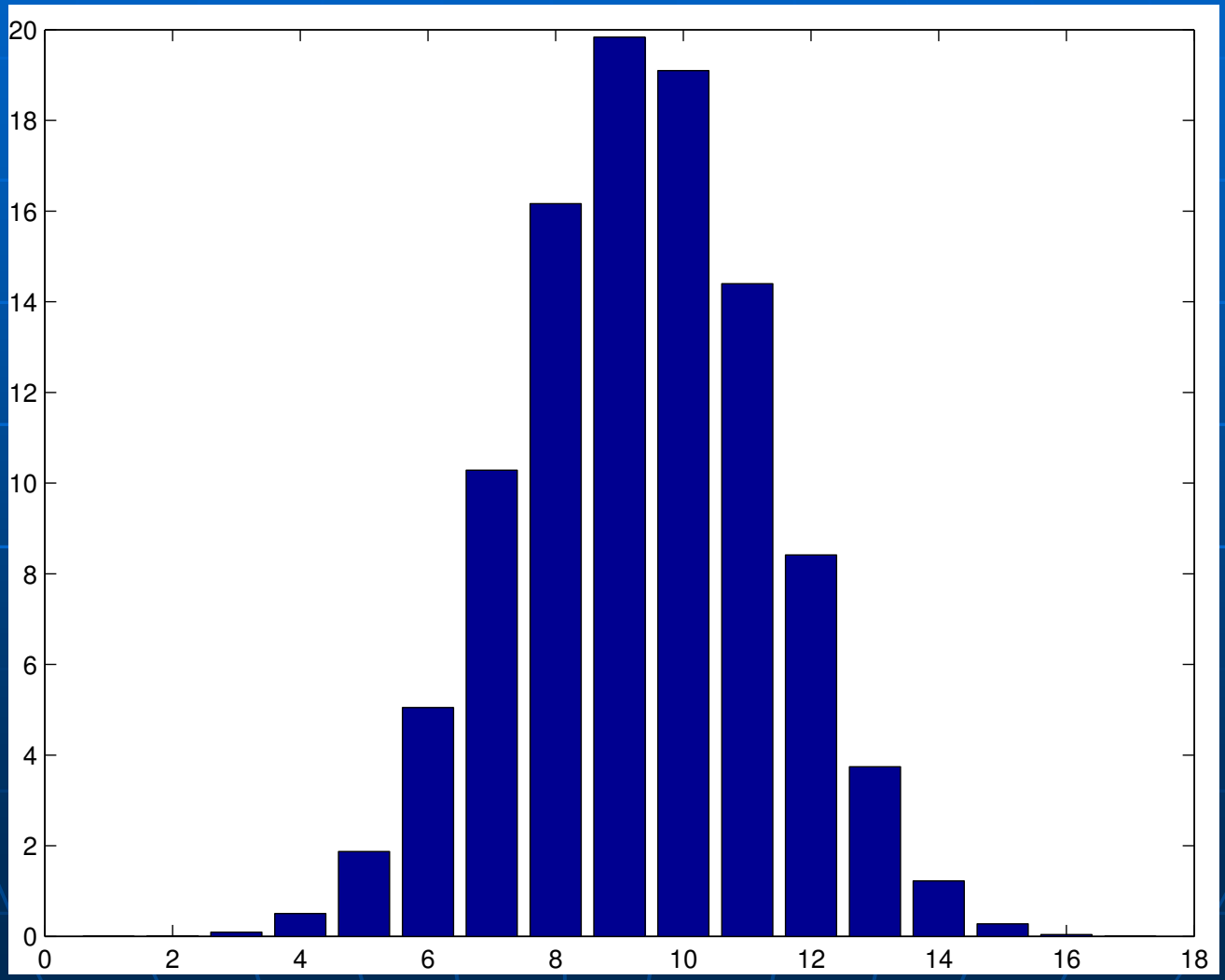
How to detect a cluster ?

- The CD vector shows some clustering pattern. But are they statistically significant?
- Statistical Hypothesis Testing:

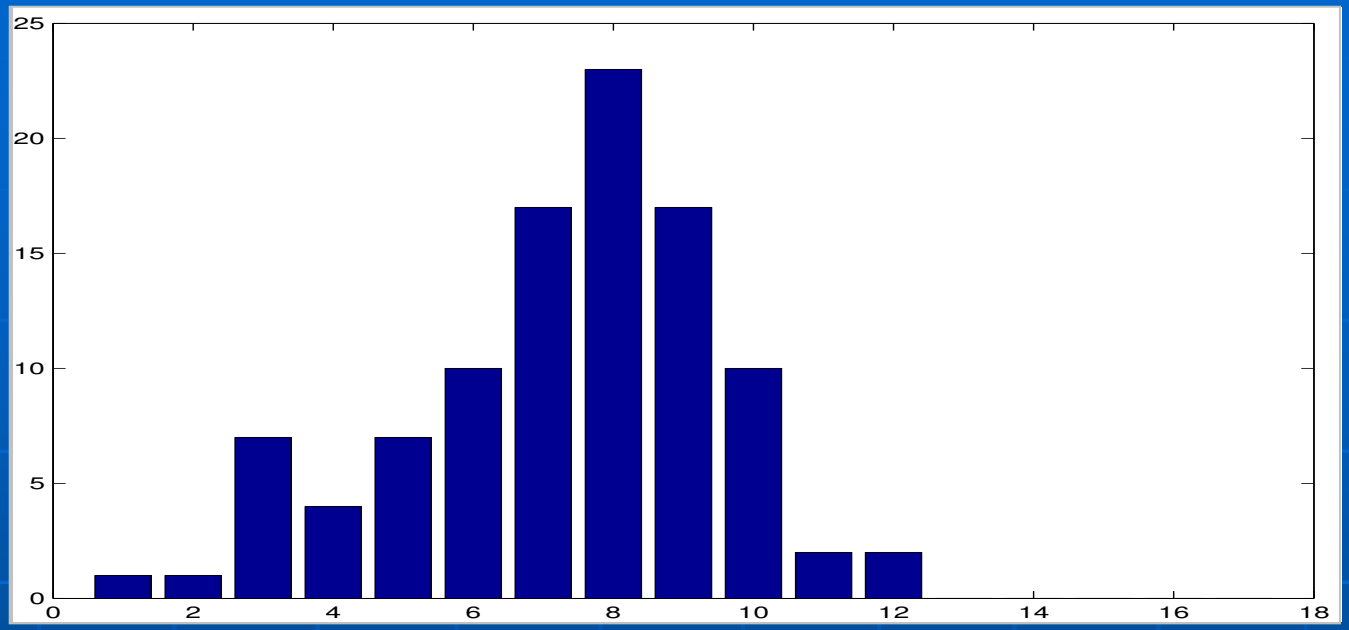
  Null Hypothesis: Uniformly distributed.

  Alternative: Not uniformly distributed.
- We call the expected CD vector under the null Uniform CD vector (UCD).

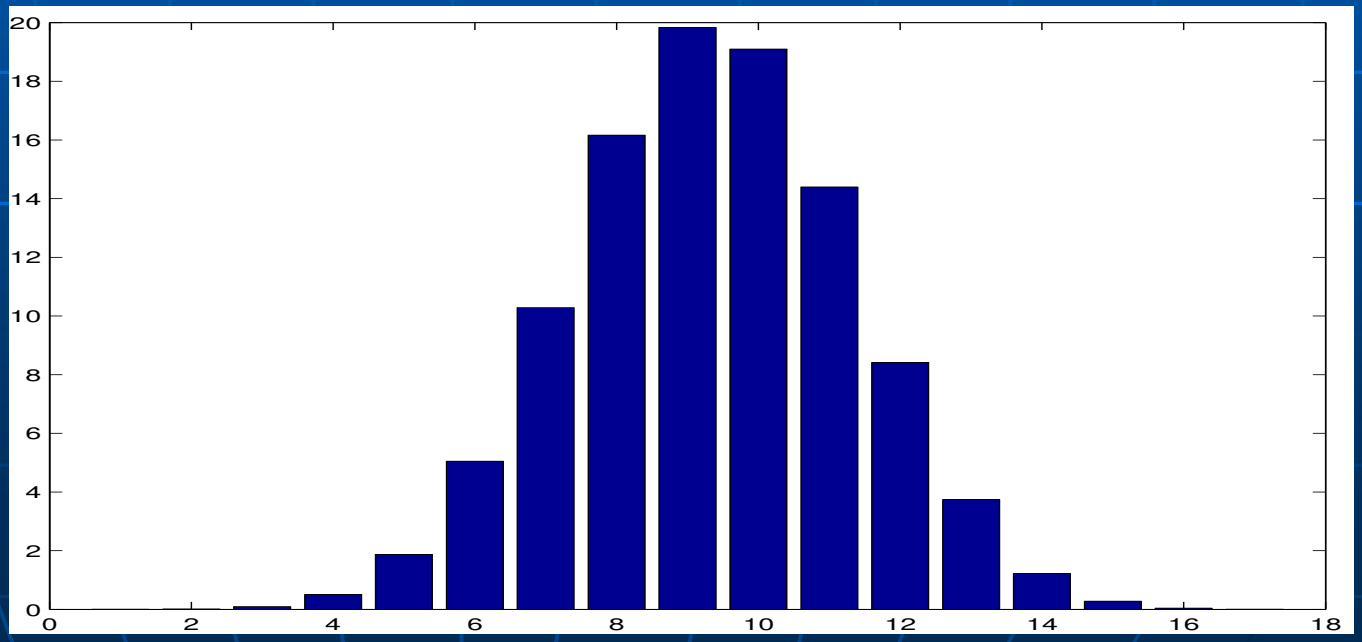
UCD: Expected CD vector under Null.



CD Vector



UCD Vector



How to compare these 2 vectors?

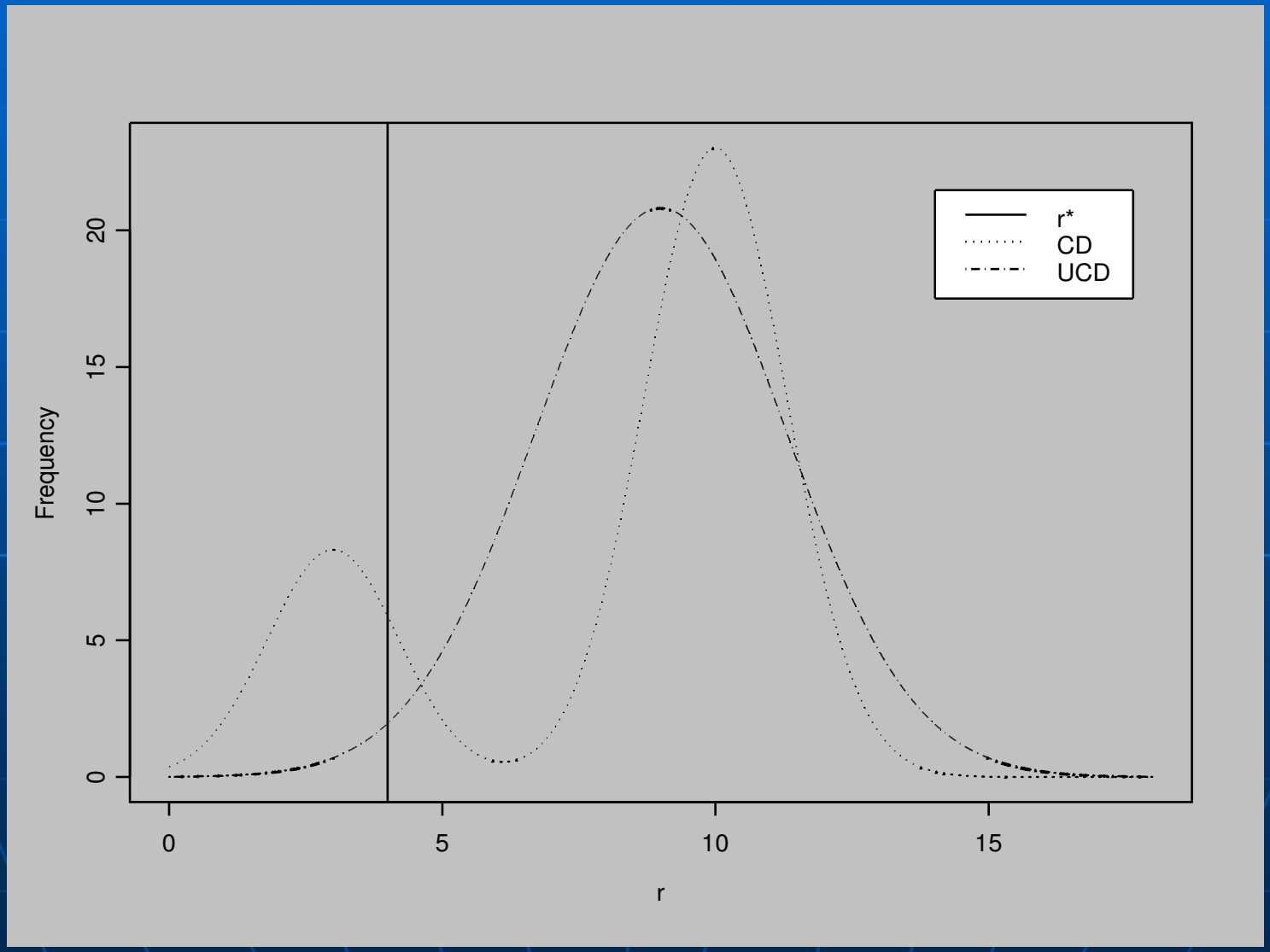
- One is the observed CD vector.
- The other is the expected CD vector under null hypothesis.
- Chi-square is the most natural tool to test the null hypothesis based on these two vectors.
- However clustering patterns are all local features. Thus we are not interested in a comparison at a global level.

Modified Chi-square Test

The modified Chi-square is defined as:

$$\chi^2_M = \sum_{i=1}^{C} \frac{(U_i - E_i)^2}{E_i} + \frac{\left((n - \sum_{k=1}^{C} U_i) - (n - \sum_{k=1}^{C} E_i) \right)^2}{E_i}$$

Choice of C and Radius of a Cluster



CD Algorithm

- Find a cluster center;
- Construct the CD vector given the current center ;
- Perform modified Chi-square test;
- If we reject the null, then determine the radius of the current cluster;
- Extract the cluster
- Repeat until we do not reject the null.

Numerical Comparison with K-mode and AutoClass

	CD	AutoClass	K-mode		
No. of Clusters	4	4	[3]	[4]	[5]
Classi. Rates	100%	*100%*	*75%*	*84%*	*82%*
"Variations"	0%	*0%*	*6%*	*15%*	*10%*
Inform. Gain	100%	*100%*	*67%*	*84%*	*93%*
"Variations"	0%	*0%*	*10%*	*15%*	*11%*

Soybean Data: n=47 and p=35. No of clusters=4.

Numerical Comparison with K-mode and AutoClass

	CD	AutoClass	K-mode		
No. of Clusters	7	3	[6]	[7]	[8]
Classi. Rates	95%	73%	74%	72%	71%
"Variations"	0%	0%	6%	15%	10%
Inform. Gain	92%	60%	75%	79%	81%
"Variations"	0%	0%	7%	6%	6%

Zoo Data: n=101 and p=16. No of clusters=7.

Run Times Comparison

	K-modes	CD
Soybean		
Average	0.0653	0.0496
S.D	0.0029	0.0010
Zoo Data		
Average	0.0139	0.0022
S.D	0.0018	0.0001

Note that AutoClass requires human intervention.

Computational Complexity

- The upper bound of the computational complexity of our algorithm is O(kpn)
- Note that the sample size shrinks if the CD algorithm detects a cluster
- It is less computational intensive than K-modes and AutoClass since both have complexity of O(akpn) where a>1.

Conclusion

- Our algorithm requires no convergence criterion.
- It automatically estimate the number of clusters.

  It does not demand or search for the true number of clusters.
- The sample size is reduced after one detected cluster is extracted.
- The computational complexity of our algorithm is bounded by O(n).

Future Work

- Scale the algorithm to large data sets by using the idea of Bradley et al.
- Generalize the idea to mixed data types
- Improve the distance function to handle correlated data
- Implement a parallel algorithm

Reference:

Zhang, P, Wang, X. and Song, P. Clustering Categorical Data Based on Distance Vectors. Revised for JASA.