Applications of Soft Clustering in Fraud and Credit Risk Modeling

Vijay S. Desai
CSCC X, Edinburgh
August 29, 2007

Need for Segmentation In Risk Modeling

- Can build better models for homogeneous segments
- Accounts exhibit different behavior
  - Transactors use credits cards for convenience and rewards, Revolvers use them for unsecured credit
  - Business travelers use cards differently from infrequent travelers
  - For the same individual, behavior for card-of-choice different from other cards owned by the individual
- Data availability
  - Short history for Young accounts versus Mature accounts

Use of Clustering to Segment Accounts

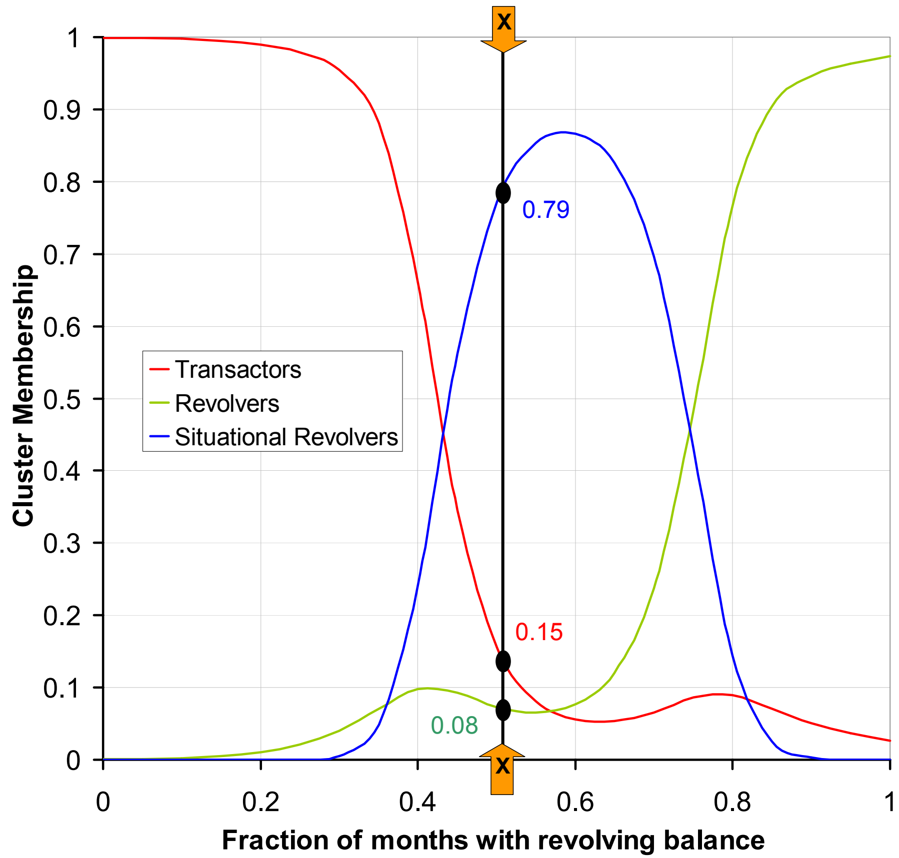
- Manual segmentation can be replaced by data driven methods like clustering
- Use relevant data to cluster accounts
  - Use transaction volume and revolving balance data to cluster revolvers, transactors
  - Use delinquency information to cluster clean, dirty accounts
- K-means clustering
  - Starts with k randomly selected seeds and assigns data to these clusters using some similarity measure
  - Centroids of the k clusters are computed and data reassigned to clusters based upon some similarity measure
  - Computationally efficient, hence the most popular method in practice

Issues with Segmentation of Accounts

- Behavior is not permanent
  - Situational revolvers
  - Business travelers use cards at home
  - Card-of-choice changes with utilization, incentives
- Data availability
  - Hard segmentation exacerbates data availability problems
    - Especially true for rare events such as fraud
- Censoring creates large gaps in data
  - Issuers don't offer high credit limits to low credit individuals
- K-means clustering
  - Susceptible to local optima, outliers
  - Only applicable for data with defined means

Ways to Alleviate Some Problems With Clustering

- Use domain expertise to assign initial cluster seeds
- Use multiple starting points to alleviate local optima problems
- Soft clustering: each data point belongs to all clusters in graded degree
  - Cluster membership determined by distance from center.
  - Data-driven: Cluster centers and shape updated intelligently
- Soft-clustering allows same account to be used in multiple models



Soft Clustering Methods

- Fuzzy clustering
  - Fuzzy c-means clustering
  - Fuzzy c-means clustering with extragrades
- Possibilistic clustering
  - Possibilistic c-means clustering
- Kernel based clustering
  - Kernel K-means
  - Kernel fuzzy c-means clustering
  - Kernel possibilistic c-means clustering

Fuzzy Clustering

- Similar to k-means clustering
- Extends the notion of membership, data point might belong to more than one cluster
- Examples:
  - Situational revolvers should not be forced into revolvers or transactors
  - Accounts with a single late payment should not be forced into current or delinquent segments
- Bezdek (1981)

Fuzzy c-means Algorithm

$$\text{Minimize } J(\mathbf{U}, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} u_{ik}^{m} d_{i,k}^{2} \qquad \text{Eq. (1)}$$

$$\text{s.t. } \sum_{i=1}^{c} u_{i,k} = 1, \forall \text{ i} = 1,\ldots, \text{n} \qquad \text{Eq. (2)}$$

where

$$0 \le u_{\mathrm{r,c}} \le 1$$

μ_{ik} is the cluster center of fuzzy group i

$d_{i,k} = \| x_k - v_i \| =$ the Euclidean distance from cluster center v_i to data point x_k

$m \in (1, \infty)$ is a weighting exponent

Fuzzy c-means Algorithm: Solution

- Picard iterations to update the centroids and membership until J(U,V) changes are below threshold

Picard Iteration scheme :

Update centroids as

$$v_i = \frac{\sum_{j=1}^{c} u_{ik}^{m} \mathbf{x}_k}{\sum_{k=1}^{n} u_{ik}^{m}} \qquad \text{Eq. (3)}$$

Update membership as

$$u_{ik} = \frac{1}{\sum_{j=1}^{c} \left(\frac{d_{ik}}{d_{ik}} \right)^{2/(m-1)}} \qquad \text{Eq. (4)}$$

Fuzzy c-means clustering with Extragrades

- McBratney and De Gruijter (1992)
- Alleviates the influence of outliers by adding a penalty term to the objective function
- Potential outliers are put in "extragrades" classes u_{k*}^{m}
- Solution involves Picard iterations updating the centroids and membership matrices

$$\text{Minimize } J(\mathbf{U}, V) = \alpha \sum_{k=1}^{n} \sum_{i=1}^{c} u_{ik}^{m} d_{i,k}^{2} + (1-\alpha) \sum_{k=1}^{n} u_{k*}^{m} \sum_{i=1}^{c} d_{i,k}^{-2}$$

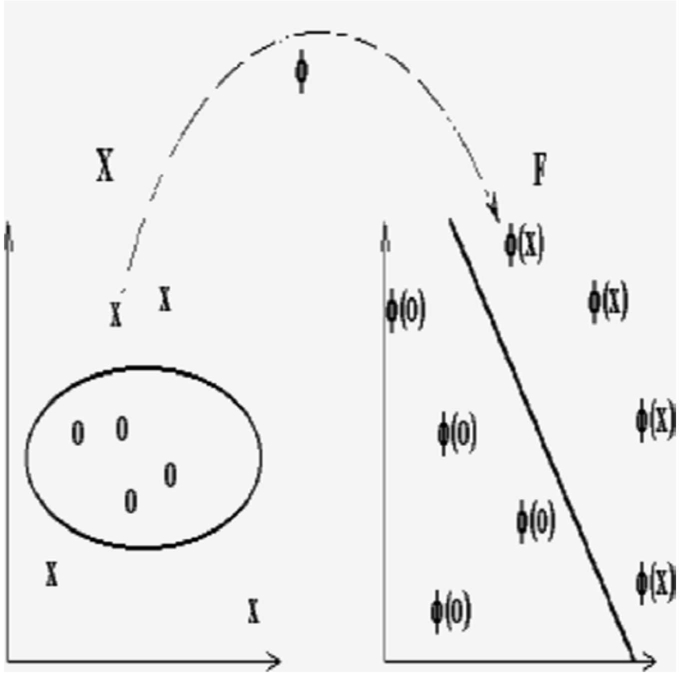
Possibilistic Clustering

- Relaxes the probabilistic constraint on membership (Eq. (2))
- Allows an observation to have low membership in all clusters, e.g., outliers
- Allows an observation to have high membership in multiple clusters, e.g., overlapping clusters
- Krishnapuram and Keller (1993), (1996)

$$\text{Minimize} \quad J(\mathbf{U}, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} u_{ik}^{m} d_{i,k}^{2} + \sum_{i=1}^{c} \eta_i \sum_{k=1}^{n} (1 - u_{i,k})^{m} \quad (1993)$$

$$\text{Minimize} \quad J(\mathbf{U}, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} u_{ik} d_{i,k}^{2} + \sum_{i=1}^{c} \eta_i \sum_{k=1}^{n} (u_{i,k} \ln(u_{i,k}) - u_{i,k}) \quad (1996)$$

Kernel Basics



- Kernel definition K(x,y)
  - A similarity measure
  - Implicit mapping ϕ, from input space to feature space
  - Condition: $K(x,y)=\phi(x)\bullet\phi(y)$
- Theorem: $K(x,y)$ is a valid kernel if K is positive definite and symmetric (Mercer Kernel)
  - A function is P.D. if $\int K(\mathbf{x},\mathbf{y})f(\mathbf{x})f(\mathbf{y})d\mathbf{x}d\mathbf{y} \geq 0 \quad \forall f \in L_2$
  - In other words, the Gram matrix **K** (whose elements are $K(x_i,x_j)$) must be positive definite for all x_i, x_j of the input space
- Kernel Examples: $\mathrm{K(x, y)} = \langle \mathrm{x, y} \rangle^d$

$$\mathrm{K(x, y)} = e^{-\|x-y\|^2/2\sigma}$$

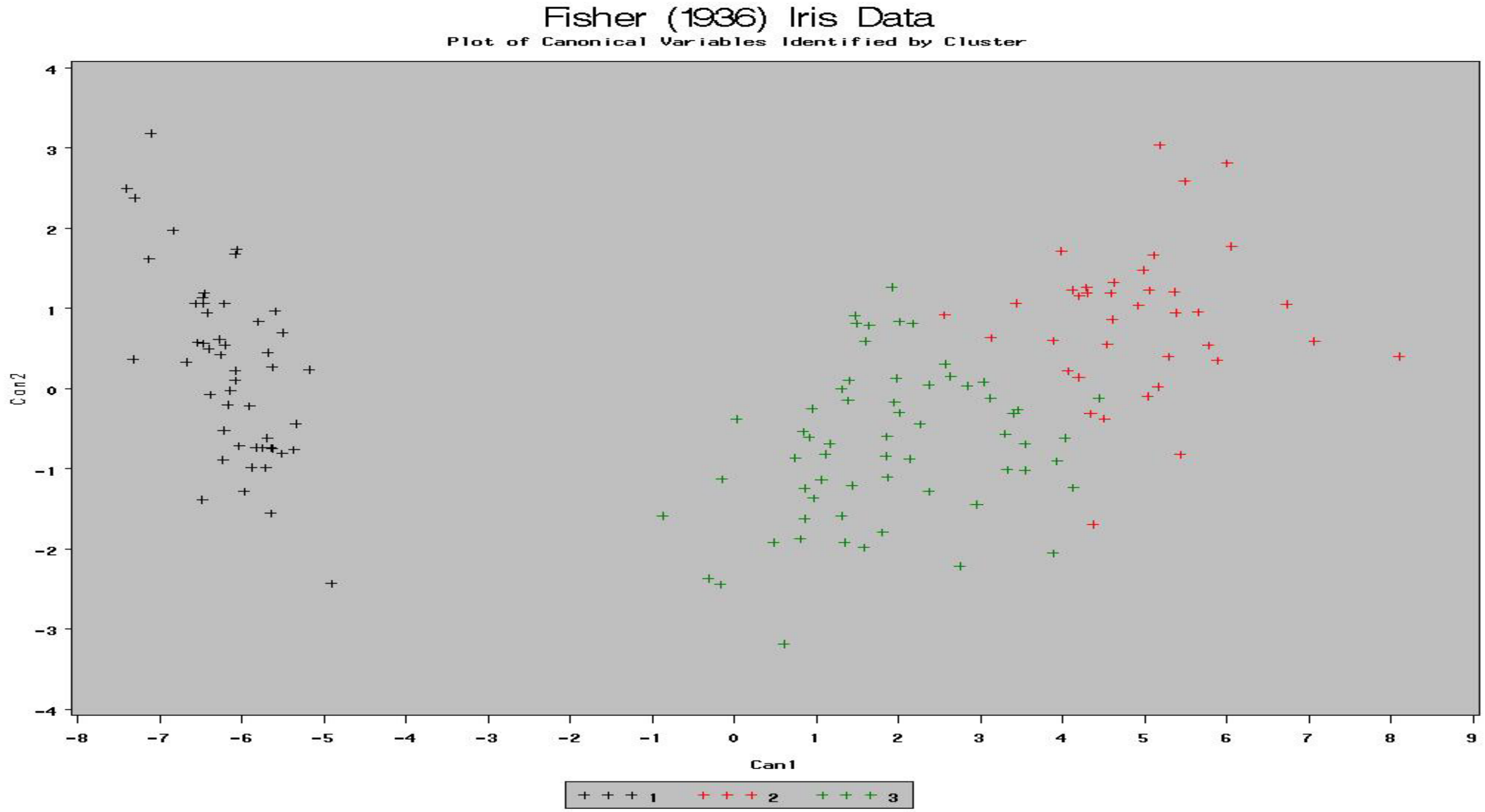
Kernel Based Clustering

- Kernel based clustering increases the separability of the input data by mapping them to a new high dimensional space
- Kernel functions allows the exploration of data patterns in new space, in contrast to K-means with Euclidean distance which expects the data to distribute into elliptical regions
- Zhong, Xie and Yu (2003)

$$\text{Minimize } J^{\phi}(\mathbf{U}, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} u_{ik}^{m} \left\| \Phi(x_k) - \Phi(v_i) \right\|^2$$

Iris Data Set Description

- 4 variables, 3 classes, 50 observations per class
- Class 1 easily separable, Class 2-3 overlapping



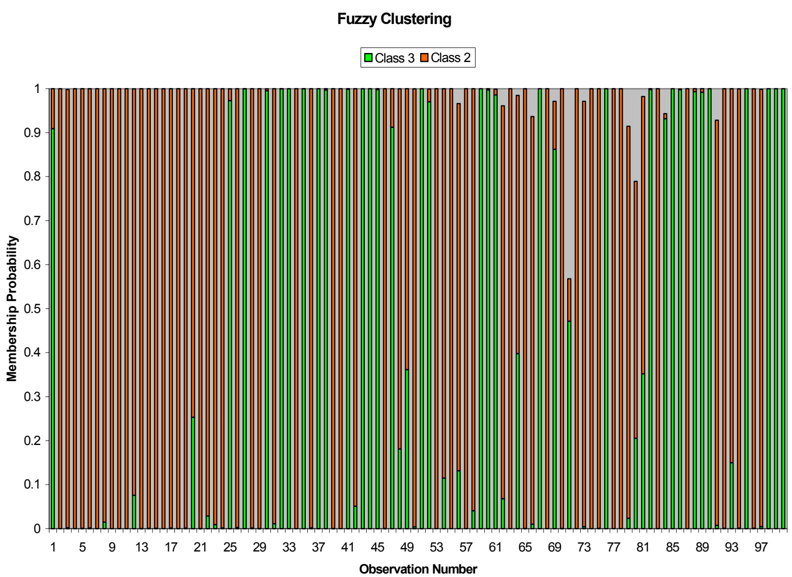
Iris Data Set: K-Means Clustering Results

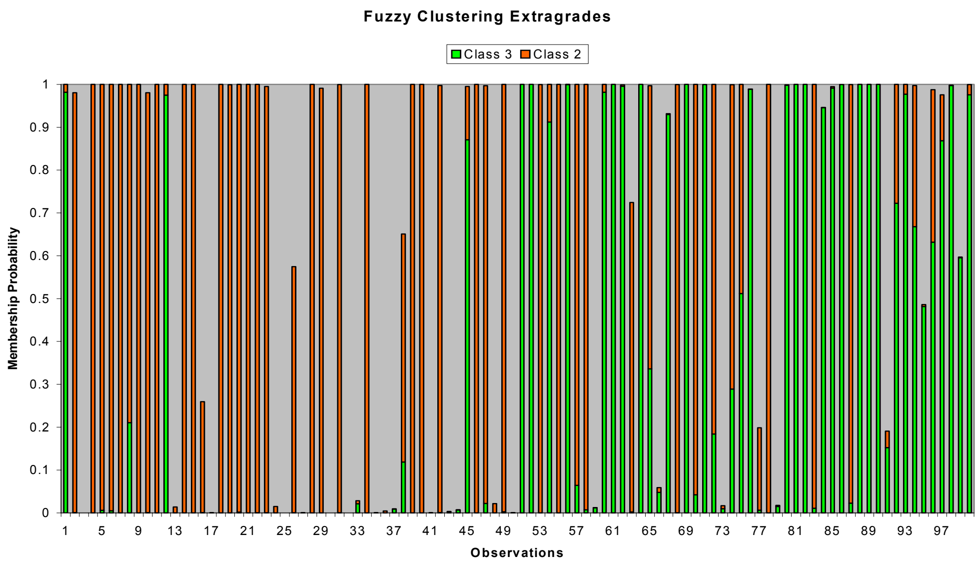
- Classes 1 and 2 identified
- Class 3 (Virginica) misclassified

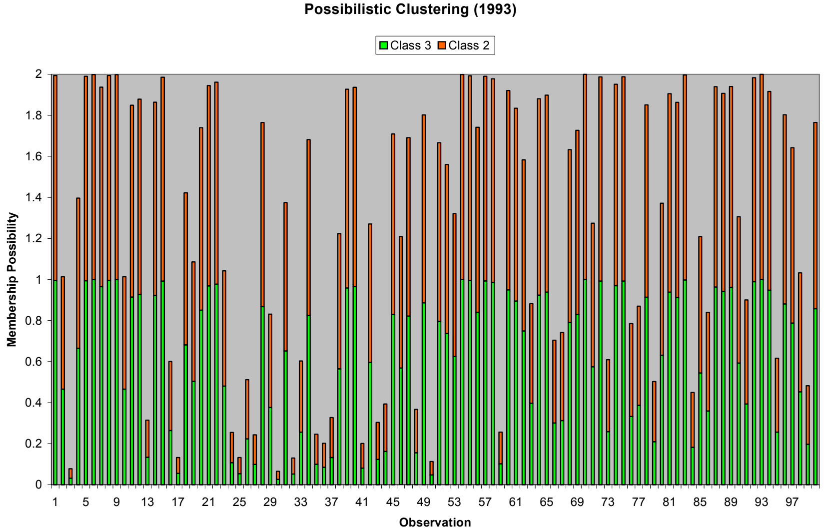
CLUSTER(Cluster) Species

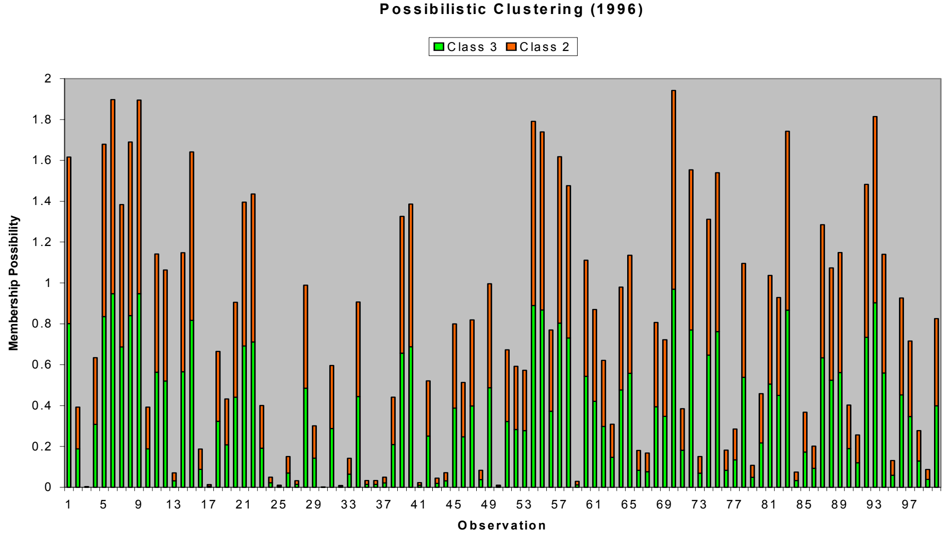
Frequency Percent Row Pct Col Pct	Setosa	Versicolor	Virginica	Total
1	50 33.33 100.00 100.00	0 0.00 0.00 0.00	0 0.00 0.00 0.00	50 33.33
2	0 0.00 0.00 0.00	0 0.00 0.00 0.00	33 22.00 100.00 66.00	33 22.00
3	0 0.00 0.00 0.00	50 33.33 74.63 100.00	17 11.33 25.37 34.00	67 44.67
Total	50 33.33	50 33.33	50 33.33	150 100.00

Iris Data: Soft Clustering Results









Suggestions on When to Use Each Method

- Very large data sets with many clusters
  - K-means clustering
- Data sets with thin shells, overlapping clusters
  - Fuzzy K-means
  - e.g., current, delinquent, dirty accounts
- Data sets with outliers
  - Fuzzy K-means with extragrades
- Data sets with overlapping clusters, noisy data, other clustering results not consistent with intuitive concepts of compatibility
  - Possibilistic clustering
  - e.g., revolvers, transactors, situational revolvers
- Data sets with non-spherical density distributions, e.g., donuts, spirals
  - Kernel based clustering in feature space
  - e.g., censored data

Using the Soft Clustering Results

- Model building phase
  - Build model for each cluster
  - Use all accounts that have a significant cluster membership probability
  - Model building data weighs each account in proportion to its membership probability
- Scoring phase
  - Score each account using all models from segments for which account had a significant membership probability
  - The final output score for the account is a weighted average of all the segment based scores
- Resulting scores are more stable, particularly for accounts with changing behavior
- Results more robust for clusters that would have few accounts under traditional approaches

Q & A

- Thank you for the opportunity to present