Classification and clustering using belief functions

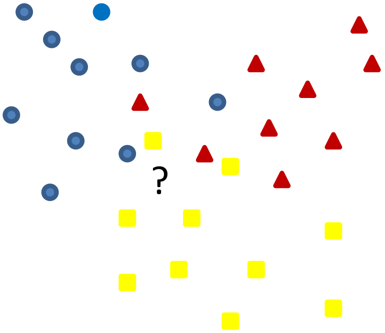
Thierry Denœux

Université de Technologie de Compiègne
HEUDIASYC (UMR CNRS 7253)

https://www.hds.utc.fr/~tdenoeux

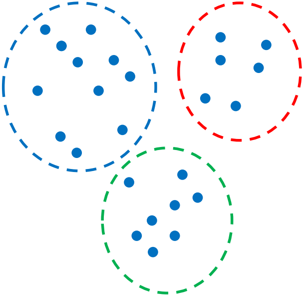
School of Automation
Northwestern Polytechnical University
Xi'an, China, October 26, 2015

Classification problem



- A population is assumed to be partitioned in c groups or classes
- Let $\Omega = \{\omega_1, \ldots, \omega_c\}$ denote the set of classes
- Each instance is described by
  - A feature vector $\boldsymbol{x} \in \mathbb{R}^p$
  - A class label $y \in \Omega$
- Problem: given a learning set $\mathcal{L} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$, predict the class label of a new instance described by $\boldsymbol{x}$

Clustering problem



- n objects described by
  - Attribute vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ (attribute data) or
  - Dissimilarities (proximity data)
- Goal: find a meaningful structure in the data set, usually a partition into c subsets, or a more complex mathematical representation (fuzzy partition, etc.)

Why can belief functions be useful?

1. Exploit the high expressiveness of belief functions to
   (a) Represent more faithfully the uncertainty of the predictions made by a classifier (for, e.g., combining several classifiers, or providing the user with richer information about the uncertainty of the classification)
   (b) Reveal richer information about the data (clustering problems)
2. Represent uncertainty about the data themselves:
   (a) Uncertain class labels (partially supervised learning)
   (b) Clustering of imprecise/uncertain data

Overview of the main approaches

Classification

1. Classifier fusion: convert the outputs from standard classifiers into belief functions and combine them using, e.g., Dempster's rule (e.g., Quost al., 2011)
2. Develop evidence-theoretic classifiers directly providing belief functions as outputs:
   (a) Generalized Bayes theorem, extends the Bayesian classifier when class densities and priors are ill-known (Appriou, 1991; Denœux and Smets, 2008)
   (b) Distance-based classifiers: evidential K-NN rule (Denœux, 1995), evidential neural network classifier (Denœux, 2000)
   (c) Predictive evidential classifiers (e.g., logistic regression, Xu et al., 2015)

Overview of the main approaches

Clustering

Express uncertainty about the membership of objects to clusters using the notion of credal partition

1. Match degrees of conflict with inter-point distances: EVCLUS algorithm (Denoeux and Masson, 2004)
2. Extend prototype-based clustering methods such as the hard or fuzzy c-means: Evidential c-means (Masson and Denoeux, 2008)
3. Decision-directed clustering using the evidential K-NN classifier: EK-NNclus algorithm (Denoeux et al, 2015)

Overview of the main approaches

Uncertain data

1. In classification, partially supervised data (with uncertain class labels) can be handled using the evidential K-NN classifier (Denoeux, 1995; Denoeux and Zouhal, 2001)
2. More general approach: extend maximum likelihood estimation to uncertain data (e.g., with uncertain class labels and/or attributes) using the Evidential Expectation-Maximization (E^2M) algorithm (Denoeux, 2011; Denoeux, 2012)

Outline

1. Evidential classification
   - Evidential K-NN rule
   - Evidential neural network classifier

2. Evidential clustering
   - Evidential c-means
   - EK-NNclus

Outline

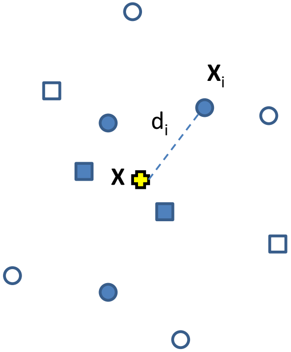
1. Evidential classification
   - Evidential K-NN rule
   - Evidential neural network classifier
2. Evidential clustering
   - Evidential c-means
   - EK-NNclus

Outline

1. Evidential classification
   - Evidential K-NN rule
   - Evidential neural network classifier
2. Evidential clustering
   - Evidential c-means
   - EK-NNclus

Evidential K-NN rule

Principle



- Let $\mathcal{N}_K(\boldsymbol{x}) \subset \mathcal{L}$ denote the set of the K nearest neighbors of $\boldsymbol{x}$ in $\mathcal{L}$, based on some distance measure
- Each $\boldsymbol{x}_i \in \mathcal{N}_K(\boldsymbol{x})$ can be considered as a piece of evidence regarding the class of $\boldsymbol{x}$
- The strength of this evidence decreases with the distance d_i between $\boldsymbol{x}$ and $\boldsymbol{x}_i$

Evidential K-NN rule

Definition

- If $y_i = \omega_k$, the evidence of $(\boldsymbol{x}_i, y_i)$ can be represented by

$$m_i(\{\omega_k\}) = \varphi(d_i)$$
$$m_i(\{\omega_\ell\}) = 0, \quad \forall \ell \neq k$$
$$m_i(\Omega) = 1 - \varphi(d_i)$$

  where and φ is a decreasing function from $[0, +\infty)$ to $[0, 1]$ such that $\lim_{d \to +\infty} \varphi(d) = 0$

- The evidence of the K nearest neighbors of $\boldsymbol{x}$ is pooled using Dempster's rule of combination

$$m = \bigoplus_{\boldsymbol{x}_i \in \mathcal{N}_K(\boldsymbol{x})} m_i$$

- Function φ can be fixed heuristically or selected among a family $\{\varphi_\theta | \theta \in \Theta\}$ using, e.g., cross-validation

Evidential K-NN rule
Decision

- Let $\lambda_{k\ell}$ be the cost of assigning a pattern to class k, if it actually belongs to class ℓ
- Assume that $\lambda_{k\ell} = 0$ if $k = \ell$ and $\lambda_{k\ell} = 1$ otherwise
- Given a mass function m on Ω, the lower and upper expected costs if the pattern is assigned to class k are

$$R_*(\alpha_k) = \sum_{A \subseteq \Omega} m(A) \min_{\omega_\ell \in A} \lambda_{k\ell} = 1 - Pl(\{\omega_k\})$$

$$R^*(\alpha_k) = \sum_{A \subseteq \Omega} m(A) \max_{\omega_\ell \in A} \lambda_{k\ell} = 1 - Bel(\{\omega_k\})$$

- Corresponding decision rules: select the class with the maximum plausibility or the maximum degree of belief
- The maximum plausibility rule has a computational advantage

Evidential K-NN rule

Practical computation

- The plausibilities for each mass function m_i are

$$pl_i(\omega_k) = \begin{cases} 1 & \text{if } y_i = \omega_k \\ 0 & \text{otherwise} \end{cases}, \quad k = 1, \ldots, c$$

- The plausibilities for the combined mass function m are

$$pl(\omega_k) \propto \prod_{\boldsymbol{x}_i \in \mathcal{N}_K(\boldsymbol{x})} (1 - \varphi(d_i))^{1 - s_{ik}}$$

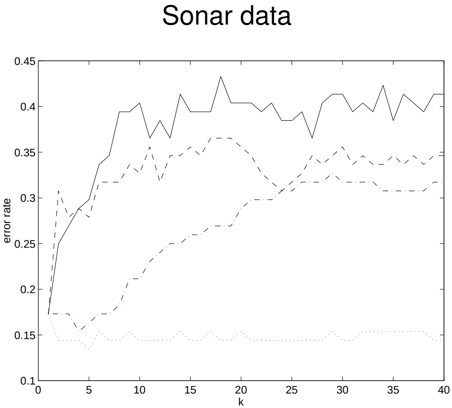
  where $s_{ik} = 1$ if $y_i = \omega_k$ and $s_{ik} = 0$ otherwise

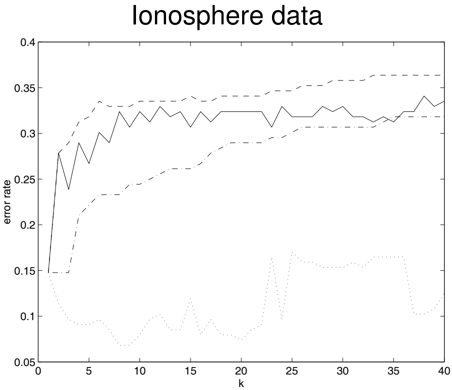
- The log-plausibilities are

$$\ln pl(\omega_k) = - \sum_{\boldsymbol{x}_i \in \mathcal{N}_K(\boldsymbol{x})} s_{ik} \ln (1 - \varphi(d_i)) + C$$

- They can be computed in time proportional to $K|\Omega|$

Performance comparison (UCI database)





Test error rates as a function of K for the voting (-), evidential (:), fuzzy (–) and distance-weighted (-.) k-NN rules

Partially supervised data

- We now consider a learning set of the form

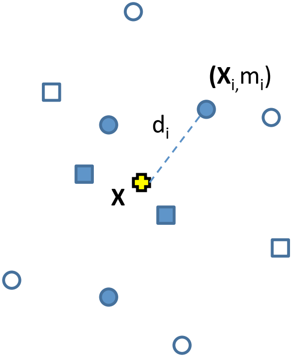
$$\mathcal{L} = \{(\boldsymbol{x}_i, m_i), i = 1, \ldots, n\}$$

where

  - $\boldsymbol{x}_i$ is the attribute vector for instance i, and
  - m_i is a mass function representing uncertain expert knowledge about the class y_i of instance i

- Special cases:
  - $m_i(\{\omega_k\}) = 1$ for all i: supervised learning
  - $m_i(\Omega) = 1$ for all i: unsupervised learning

Evidential k-NN rule for partially supervised data



- Each mass function m_i is discounted (weakened) with a rate depending on the distance d_i

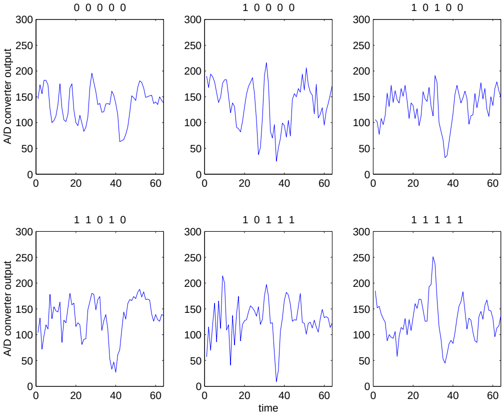
$$m_i'(A) = \varphi(d_i)\, m_i(A), \quad \forall A \subset \Omega$$
$$m_i'(\Omega) = 1 - \sum_{A \subset \Omega} m_i'(A)$$

- The K mass functions m_i' are combined using Dempster's rule

$$m = \bigoplus_{\boldsymbol{x}_i \in \mathcal{N}_K(\boldsymbol{x})} m_i'$$

Example: EEG data

EEG signals encoded as 64-D patterns, 50 % positive (K-complexes), 50 % negative (delta waves), 5 experts.



Results on EEG data
(Denoeux and Zouhal, 2001)

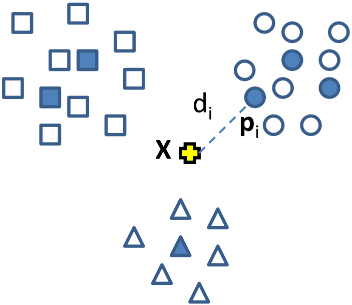
- $c = 2$ classes, $p = 64$
- For each learning instance $\boldsymbol{x}_i$, the expert opinions were modeled as a mass function m_i.
- $n = 200$ learning patterns, 300 test patterns

K	K-NN	w K-NN	Ev. K-NN (crisp labels)	Ev. K-NN (uncert. labels)
9	0.30	0.30	0.31	0.27
11	0.29	0.30	0.29	0.26
13	0.31	0.30	0.31	0.26

Outline

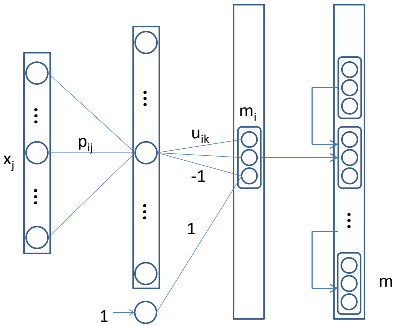
1. Evidential classification
   - Evidential K-NN rule
   - Evidential neural network classifier
2. Evidential clustering
   - Evidential c-means
   - EK-NNclus

Evidential neural network classifier



- The learning set is summarized by r prototypes.
- Each prototype $\boldsymbol{p}_i$ has membership degree u_{ik} to each class ω_k, with $\sum_{k=1}^{c} u_{ik} = 1$.
- Each prototype $\boldsymbol{p}_i$ brings a piece of evidence regarding the class of $\boldsymbol{x}$, whose reliability decreases with the distance d_i between $\boldsymbol{x}$ and $\boldsymbol{p}_i$.

Neural network architecture



- Mass function induced by $\boldsymbol{p}_i$:

$$\begin{aligned} m_i(\{\omega_k\}) &= \alpha_i u_{ik} \exp(-\gamma_i d_i^2), \\ & k = 1, \ldots, c \\ m_i(\Omega) &= 1 - \alpha_i \exp(-\gamma_i d_i^2) \end{aligned}$$

- Combination:

$$m = \bigoplus_{i=1}^{r} m_i$$

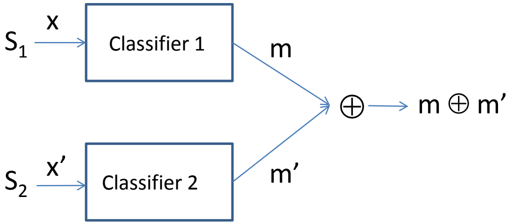
- All parameters are learnt from data by minimizing an error function.

Results on classical data

Vowel data
$c = 11$,
$p = 10$
$n = 568$
test: 462 ex.
(different
speakers)

Classifier	test error rate
Multi-layer perceptron (88 units)	0.49
Radial Basis Function (528 units)	0.47
Gaussian node network (528 units)	0.45
Nearest neighbor	0.44
Linear Discriminant Analysis	0.56
Quadratic Discriminant Analysis	0.53
CART	0.56
BRUTO	0.44
MARS (degree=2)	0.42
Evidential NN (33 prototypes)	0.38
Evidential NN (44 prototypes)	0.37
Evidential NN (55 prototypes)	0.37

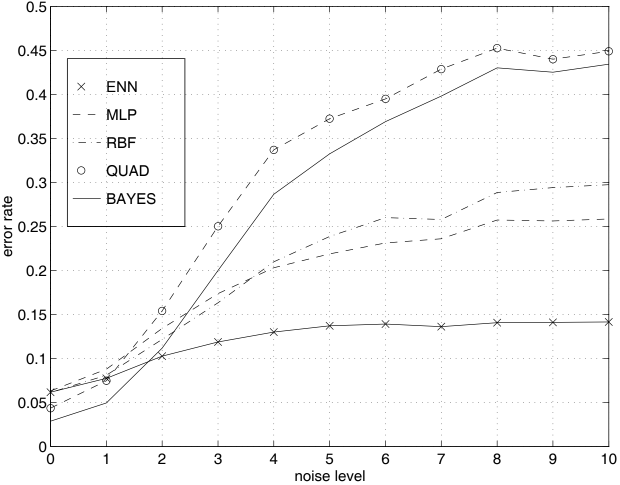
Data fusion example



- $c = 2$ classes
- Learning set ($n = 60$): $\boldsymbol{x} \in \mathbb{R}^5$, $\boldsymbol{x}' \in \mathbb{R}^3$, Gaussian distributions, conditionally independent
- Test set (real operating conditions): $\boldsymbol{x} \leftarrow \boldsymbol{x} + \epsilon$, $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$

Results

Test error rates: $\boldsymbol{x} + \epsilon$, $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$



Outline

Clustering concepts

Hard and fuzzy clustering

- Hard clustering: each object belongs to one and only one group. Group membership is expressed by binary variables u_{ik} such that $u_{ik} = 1$ if object i belongs to group k and $u_{ik} = 0$ otherwise
- Fuzzy clustering: each object has a degree of membership $u_{ik} \in [0, 1]$ to each group, with $\sum_{k=1}^{c} u_{ik} = 1$
- Fuzzy clustering with noise cluster: each object has a degree of membership $u_{ik} \in [0, 1]$ to each group and a degree of membership $u_{i*} \in [0, 1]$ to a noise cluster, with $\sum_{k=1}^{c} u_{ik} + u_{i*} = 1$

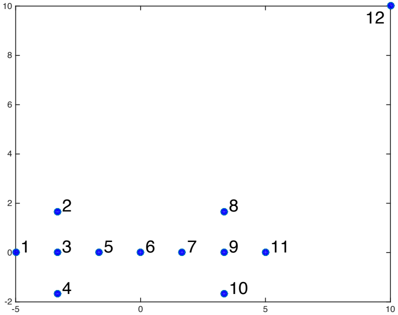
Clustering concepts

Possibilistic, rough, credal clustering

- Possibilistic clustering: the condition $\sum_{k=1}^{c} u_{ik} = 1$ is relaxed. Each number u_{ik} can be interpreted as a degree of possibility that object i belongs to cluster k
- Rough clustering: the membership of object i to cluster k is described by a pair $(\underline{u}_{ik}, \overline{u}_{ik}) \in \{0,1\}^2$, with $\underline{u}_{ik} \leq \overline{u}_{ik}$, indicating its membership to the lower and upper approximations of cluster k
- Evidential clustering: based on Dempster-Shafer (DS) theory (the topic of this talk)

Evidential clustering

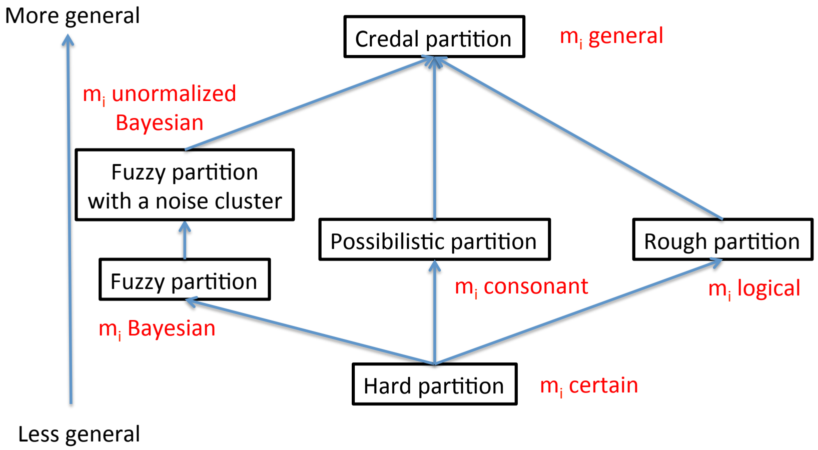
- In evidential/credal clustering, the cluster membership of each object is considered to be uncertain and is described by a (not necessarily normalized) mass function m_i over Ω
- Example:



Credal partition

	$\emptyset$	$\{\omega_1\}$	$\{\omega_2\}$	$\{\omega_1, \omega_2\}$
m_3	0	1	0	0
m_5	0	0.5	0	0.5
m_6	0	0	0	1
m_{12}	0.9	0	0.1	0

Relationship with other clustering structures



Algorithms

1. EVCLUS (Denoeux and Masson, 2004)
   - Proximity (possibly non metric) data
   - Multidimensional scaling approach
   - Variant: Constrained EVCLUS (CEVCLUS) (Antoine et al., 2014): EVCLUS with pairwise constraints
2. Evidential *c*-means (ECM) (Masson and Denoeux, 2008)
   - Attribute data
   - HCM, FCM family (alternate optimization of a cost function)
   - Variants
     - Relational Evidential *c*-means (RECM): (Masson and Denoeux, 2009): ECM for proximity data
     - Constrained Evidential *c*-means (CECM) (Antoine et al., 2011): ECM with pairwise constraints
     - Spatial Evidential C-Means (Lelandais et al., 2014): ECM with spatial constraints, for image segmentation
3. EK-NNclus (Denoeux et al, 2015)
   - Attribute or proximity data
   - Decision-directed clustering algorithm based on the evidential K-NN classifier

Outline

1. Evidential classification
   - Evidential K-NN rule
   - Evidential neural network classifier
2. Evidential clustering
   - Evidential c-means
   - EK-NNclus

Principle

- Problem: generate a credal partition $M = (m_1, \ldots, m_n)$ from attribute data $X = (\boldsymbol{x}_1, ..., \boldsymbol{x}_n)$, $\boldsymbol{x}_i \in \mathbb{R}^p$
- Generalization of hard and fuzzy c-means algorithms:
  - Each class represented by a prototype
  - Alternate optimization of a cost function with respect to the prototypes and to the credal partition

Fuzzy c-means (FCM)

- Minimize

$$J_{\text{FCM}}(U,V)=\sum_{i=1}^{n}\sum_{k=1}^{c} u_{ik}^{\beta} d_{ik}^{2}$$

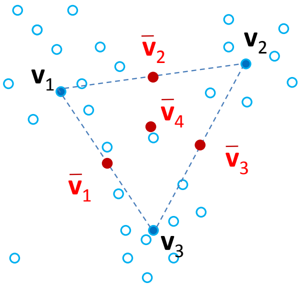
with $d_{ik}=||\boldsymbol{x}_i-\boldsymbol{v}_k||$ under the constraints $\sum_k u_{ik}=1$ for all i

- Alternate optimization algorithm:

$$\boldsymbol{v}_k=\frac{\sum_{i=1}^{n} u_{ik}^{\beta}\boldsymbol{x}_i}{\sum_{i=1}^{n} u_{ik}^{\beta}} \quad k=1,\ldots,c$$

$$u_{ik}=\frac{d_{ik}^{-2/(\beta-1)}}{\sum_{\ell=1}^{c} d_{i\ell}^{-2/(\beta-1)}}$$

ECM algorithm
Principle



- Each class ω_k represented by a prototype $\boldsymbol{v}_k$
- Each non empty set of classes A_j represented by a prototype $\bar{\boldsymbol{v}}_j$ defined as the center of mass of the $\boldsymbol{v}_k$ for all $\omega_k \in A_j$
- Basic ideas:
  - For each non empty $A_j \in \Omega$, $m_{ij} = m_i(A_j)$ should be high if $\boldsymbol{x}_i$ is close to $\bar{\boldsymbol{v}}_j$
  - The distance to the empty set is defined as a fixed value δ

ECM algorithm

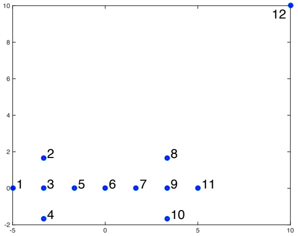
Objective criterion

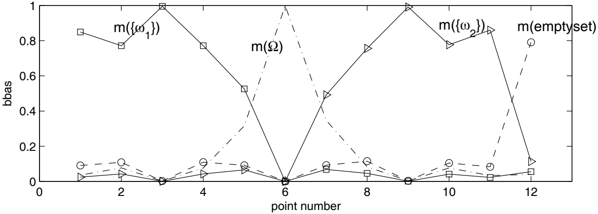
- Criterion to be minimized:

$$J_{\text{ECM}}(M,V)=\sum_{i=1}^{n}\sum_{\{j/A_j\neq\emptyset,A_j\subseteq\Omega\}}|A_j|^{\alpha}m_{ij}^{\beta}d_{ij}^{2}+\sum_{i=1}^{n}\delta^2 m_{i\emptyset}^{\beta}$$

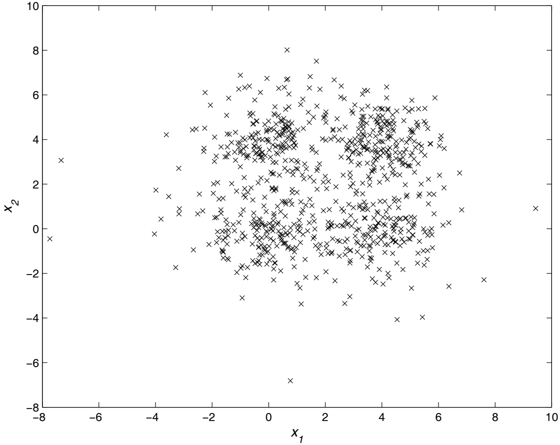
- Parameters:
  - α controls the specificity of mass functions
  - β controls the hardness of the evidential partition
  - δ controls the amount of data considered as outliers
- $J_{\text{ECM}}(M,V)$ can be iteratively minimized with respect to M and V using an alternate optimization scheme

Butterfly dataset



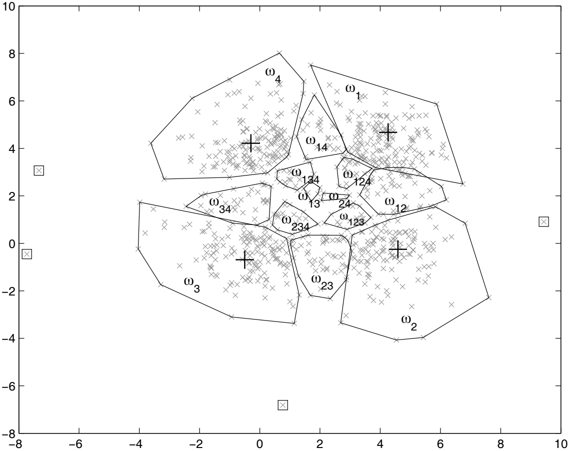


4-class data set



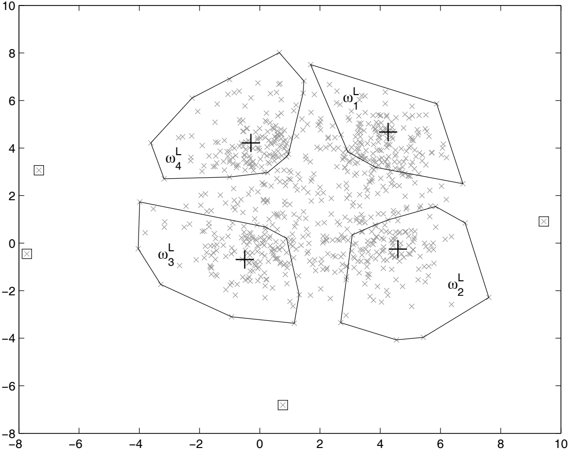
4-class data set

Hard credal partition

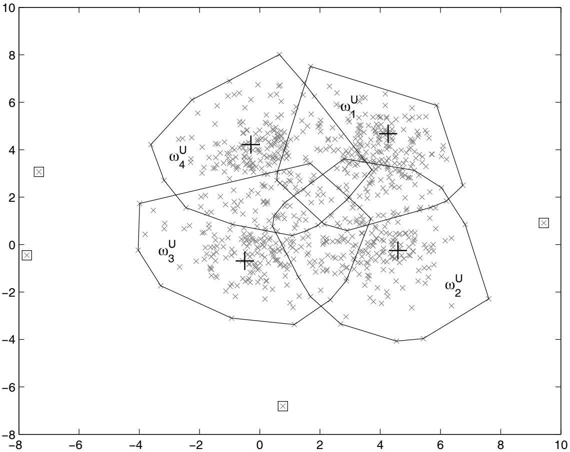


4-class data set

Lower approximation



4-class data set
Upper approximation



Outline

1. Evidential classification
   - Evidential K-NN rule
   - Evidential neural network classifier

2. Evidential clustering
   - Evidential c-means
   - EK-NNclus

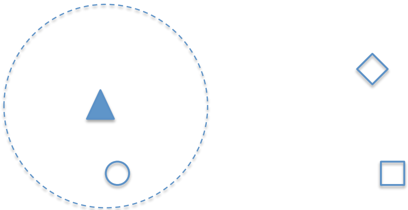
Decision-directed clustering

- Decision-directed approach to clustering:
  - Prior knowledge is used to design a classifier, which is used to label the samples
  - The classifier is then updated, and the process is repeated until no changes occur in the labels
- For instance, the c-means algorithm is based on this principle: here, the nearest-prototype classifier is used to label the samples, and it is updated by taking as prototypes the centers of each cluster
- Idea: apply this principle using the evidential K-NN rule as the base classifier

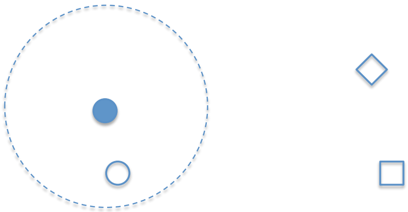
Example
Toy dataset

Example

Iteration 1

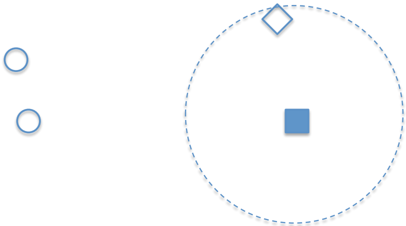


Example
Iteration 1 (continued)

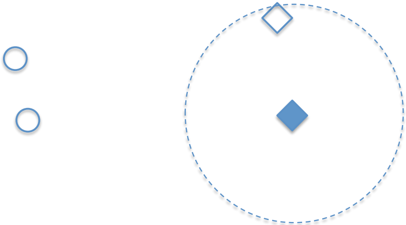


Example

Iteration 2



Example
Iteration 2 (continued)



Example
Result

EK-NNclus algorithm
Step 1: preparation

- Let $D = (d_{ij})$ be a symmetric $n \times n$ matrix of distances or dissimilarities between the n objects
- Given K, we compute the set $N_K(i)$ of indices of the K nearest neighbors of object i.
- We then compute

$$\alpha_{ij} = \begin{cases} \varphi(d_{ij}) & \text{if } j \in N_K(i) \\ 0 & \text{otherwise,} \end{cases}$$

$$v_{ij} = -\ln(1 - \alpha_{ij})$$

  for all $(i, j) \in \{1, \ldots, n\}^2$
- If computing time is not an issue, K can be chosen very large, even equal to $n - 1$

EK-NNclus algorithm

Step 2: initialization

- To initialize the algorithm, the objects are labeled randomly (or using some prior knowledge if available)
- As the number of clusters is usually unknown, it can be set to $c = n$, i.e., we initially assume that there are as many clusters as objects and each cluster contains exactly one object
- If n is very large, we can give c a large value, but smaller than n, and initialize the object labels randomly
- As before, we define cluster-membership binary variables s_{ik} as $s_{ik} = 1$ is object o_i belongs to cluster k, and $s_{ik} = 0$ otherwise

EK-NNclus algorithm

Step 3: iteration

- An iteration of the algorithm consists in updating the object labels in some random order, using the EKNN rule
- For each object o_i, we compute the log-plausibilities of belonging to each cluster (up to an additive constant) as

$$u_{ik} = \sum_{j \in N_K(i)} v_{ij} s_{jk}, \quad k = 1, \ldots, c$$

- We then assign object o_i to the cluster with the highest plausibility, i.e., we update the variables s_{ik} as

$$s_{ik} = \begin{cases} 1 & \text{if } u_{ik} = \max_{k'} u_{ik'} \\ 0 & \text{otherwise} \end{cases}$$

- If the label of at least one object has been changed during the last iteration, then the objects are randomly re-ordered and a new iteration is started. Otherwise, we move to the last step described next, and the algorithm is stopped

EK-NNclus algorithm

Step 4: Computation of the credal partition

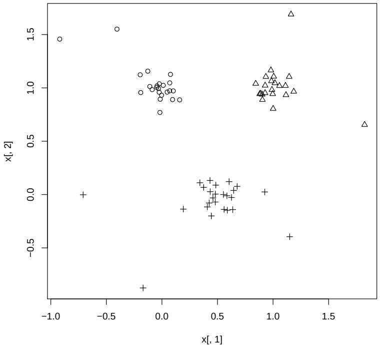
After the algorithm has converged, we can compute the final mass functions

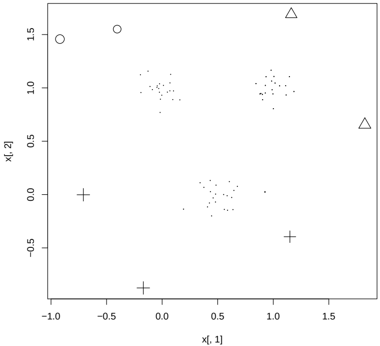
$$m_i = \bigoplus_{j \in N_K(i)} m_{ij}$$

for $i = 1, \ldots, n$, where each m_{ij} is the following mass function,

$$m_{ij}(\{\omega_{k(j)}\}) = \alpha_{ij}$$
$$m_{ij}(\Omega) = 1 - \alpha_{ij}$$

Example





Properties

- The EK-NNclus algorithm can be implemented exactly in a competitive Hopfield neural network model
- The neural network converges a stable state corresponding to a local minimum of the following energy function

$$E(S) = -\frac{1}{2}\sum_{k=1}^{c}\sum_{i=1}^{n}\sum_{j\neq i} v_{ij} s_{ik} s_{jk}$$

  where $S = (s_{ik})$ denotes the $n \times c$ matrix of 0s and 1s encoding the neuron states
- The following relation holds

$$pl(R) = -E(S) + C$$

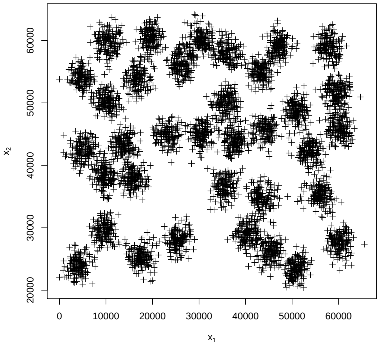
  where $pl(R)$ is the plausibility of the partition encoded by S
- The EK-NNclus algorithm thus searches for the most plausible partition, in the (huge) space of all partitions of the dataset!

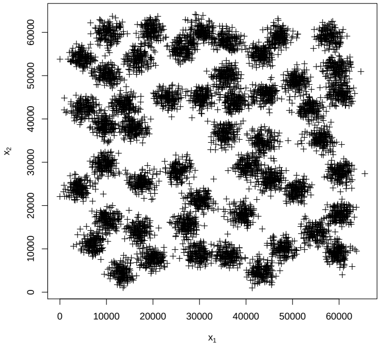
Experiments

- Settings:
  - $\varphi(d_{ij}) = \exp(-\gamma d_{ij}^2)$, where d_{ij} is the Euclidean distance between objects i and j
  - Parameter γ was fixed to the inverse of the q-quantile of the squared distances between an object and its K NN, with $q = 0.9$
  - Number K of neighbors: two to three times $\sqrt{n}$
  - Initialization methods: $c_0 = n$ initial clusters, or $c_0 = 1000$ random initial clusters
- Datasets[1]
  1. A-sets: Two-dimensional datasets with $n \in \{3000, 5250, 7000\}$ objects and $c \in \{20, 35, 50\}$ clusters
  2. DIM-sets: $n = 1024$ objects and 16 Gaussian clusters in 256, 512 and 1024 dimensions

[1] From `http://cs.joensuu.fi/sipu/datasets`

A-sets


60000
50000
40000
30000
20000
x2
0
10000
20000
30000
40000
50000
60000
x1


0
10000
20000
30000
40000
50000
60000
x2
0
10000
20000
30000
40000
50000
60000
x1

Results with the A-sets

- Number of neighbors: $K = 150$ for dataset A1, and $K = 200$ for datasets A2 and A3
- The EK-NNclus algorithm was run 10 times

Dataset	Result	EK-NNclus ($c_0 = n$)	EK-NNclus ($c_0 = 1000$)	`pdfCluster`	model-based	model-based (constrained)
A1	c	20 (0)	20 (0)	17	24	24
$n = 3000$	time	32.9 (3.14)	9.8 (0.2)	84.5	31.8	7.88
A2	c	35 (0)	34 (1)	26	39	39
$n = 5250$	time	193 (9.81)	23.8 (0.6)	298	138	36.2
A3	c	49 (1)	49 (2.5)	34	50	51
$n = 7500$	time	358 (8.23)	35.1 (1.09)	629	412	99.4

Results with the DIM-sets

- Number of neighbors: $K = 50$
- The EK-NNclus algorithm was run 10 times with $c_0 = n$

Dataset	Result	EK-NNclus	c-means	pdfCluster	model-based (constrained)
dim256	c	16 (0)	16 (fixed)	5	16
	ARI	1.0 (0)	0.94	0.23	1
	time	1.4 (0.058)	2.76	11.30	116
dim512	c	16 (0)	16(fixed)	9	16
	ARI	1 (0)	0.94	0.5	1
	time	1.4 (0.11)	13.27	10.9	467
dim1024	c	16 (0)	16 (fixed)	8	18
	ARI	1 (0)	0.94	0.28	0.998
	time	1.4 (0.14)	36.38	11.13	23

Summary

- The theory of belief function has great potential for solving challenging machine learning problems:
  - Classification (supervised learning)
  - Clustering (unsupervised learning) problems
- Belief functions allow us to:
  - Learn from weak information (partially supervised learning, imprecise and uncertain data)
  - Express uncertainty on the outputs of a learning system (e.g., credal partition)
  - Combine the outputs from several learning systems (ensemble classification and clustering)
- Recent developments make it possible to address problems in very large frames (clustering, multilabel classification, preference learning, etc.)

References I

cf. https://www.hds.utc.fr/~tdenoeux

T. Denœux.
A k-nearest neighbor classification rule based on Dempster-Shafer theory.
IEEE Transactions on SMC, 25(05):804-813, 1995.

T. Denœux.
A neural network classifier based on Dempster-Shafer theory.
IEEE transactions on SMC A, 30(2):131-150, 2000.

T. Denœux and M.-H. Masson.
EVCLUS: Evidential Clustering of Proximity Data.
IEEE Transactions on SMC B, 34(1):95-109, 2004.

M.-H. Masson and T. Denœux.
ECM: An evidential version of the fuzzy c-means algorithm.
Pattern Recognition, 41(4):1384-1397, 2008.

References II

cf. https://www.hds.utc.fr/~tdenoeux

V. Antoine, B. Quost, M.-H. Masson and T. Denoeux.
CECM: Constrained Evidential C-Means algorithm.
Computational Statistics and Data Analysis, 56(4):894-914, 2012.

B. Lelandais, S. Ruan, T. Denoeux, P. Vera, I. Gardin.
Fusion of multi-tracer PET images for Dose Painting.
Medical Image Analysis, 18(7):1247-1259, 2014.

T. Denoeux, O. Kanjanatarakul and S. Sriboonchitta.
EK-NNclus: a clustering procedure based on the evidential K-nearest neighbor rule.
Knowledge-Based Systems, Vol. 88, pages 57-69, 2015.