Classification and clustering using Belief functions

Thierry Denœux

Université de Technologie de Compiègne
HEUDIASYC (UMR CNRS 6599)
https://www.hds.utc.fr/~tdenoeux

IUKM 2016
Da Nang, Vietnam
November 29, 2016
Focus of this lecture

- **Dempster-Shafer (DS) theory** (evidence theory, theory of belief functions):
  - A formal framework for reasoning with partial (uncertain, imprecise) information.
  - Has been applied to statistical inference, expert systems, information fusion, classification, clustering, etc.

- **Purpose of these lecture:**
  - Brief introduction or reminder on DS theory;
  - Review the application of belief functions to classification and clustering.
Outline

1. Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2. Evidential classification
   - Evidential $K$-NN rule
   - Evidential neural network classifier
   - Decision analysis

3. Evidential clustering
   - Evidential partition
   - Evidential $c$-means
   - EVCLUS
   - $Ek$-NNclus
Outline

1. Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2. Evidential classification
   - Evidential $K$-NN rule
   - Evidential neural network classifier
   - Decision analysis

3. Evidential clustering
   - Evidential partition
   - Evidential $c$-means
   - EVCLUS
   - Ek-NNclus
Outline

1. Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2. Evidential classification
   - Evidential K-NN rule
   - Evidential neural network classifier
   - Decision analysis

3. Evidential clustering
   - Evidential partition
   - Evidential c-means
   - EVCLUS
   - Ek-NNclus
Mass function

- Let $\Omega$ be a finite set called a frame of discernment.
- A mass function is a function $m : 2^\Omega \rightarrow [0, 1]$ such that
  \[ \sum_{A \subseteq \Omega} m(A) = 1. \]
- The subsets $A$ of $\Omega$ such that $m(A) \neq 0$ are called the focal sets of $m$.
- If $m(\emptyset) = 0$, $m$ is said to be normalized (usually assumed).
A mass function is usually induced by a source, defined a 4-tuple \((S, 2^S, P, \Gamma)\), where

- \(S\) is a finite set;
- \(P\) is a probability measure on \((S, 2^S)\);
- \(\Gamma\) is a multi-valued-mapping from \(S\) to \(2^\Omega\).

\(\Gamma\) carries \(P\) from \(S\) to \(2^\Omega\): for all \(A \subseteq \Omega\),

\[ m(A) = P(\{s \in S | \Gamma(s) = A\}). \]
Interpretation

- $\Omega$ is a set of **possible states of the world**, about which we collect some evidence. Let $\omega$ be the true state.
- $S$ is a set of interpretations of the evidence.
- If $s \in S$ holds, we know that $\omega$ belongs to the subset $\Gamma(s)$ of $\Omega$, and nothing more.
- $m(A)$ is then the **probability of knowing only that $\omega \in A$**.
- In particular, $m(\Omega)$ is the probability of knowing nothing.
Example

- A murder has been committed. There are three suspects: \( \Omega = \{ \text{Peter, John, Mary} \} \).
- A witness saw the murderer going away, but he is short-sighted and he only saw that it was a man. We know that the witness is drunk 20% of the time.

We have \( \Gamma(\neg\text{drunk}) = \{ \text{Peter, John} \} \) and \( \Gamma(\text{drunk}) = \Omega \), hence

\[
m(\{\text{Peter, John}\}) = 0.8, \quad m(\Omega) = 0.2
\]
Special cases

- A mass function $m$ is said to be:
  - **logical** if it has only one focal set; it is then equivalent to a set.
  - **Bayesian** if all focal sets are singletons; it is equivalent to a probability distribution.

- A mass function can thus be seen as
  - a generalized set, or as
  - a generalized probability distribution.
Belief function

- If the evidence tells us that the truth is in $A$, and $A \subseteq B$, we say that the evidence supports $B$.

- Given a normalized mass function $m$, the probability that the evidence supports $B$ is thus

  \[ Bel(B) = \sum_{A \subseteq B} m(A) \]

- The number $Bel(B)$ is called the degree of belief in $B$, and the function $B \rightarrow Bel(B)$ is called a belief function.
Plausibility function

- If the evidence does not support $\bar{B}$, it is **consistent** with $B$.

\[
\Omega \\
A_1 \quad A_2 \quad A_3 \quad A_4 \\
B
\]

- The probability that the evidence is consistent with $B$ is thus

\[
Pl(B) = 1 - Bel(\bar{B}) = \sum_{A \cap B \neq \emptyset} m(A).
\]

- The number $Pl(B)$ is called the plausibility of $B$, and the function $B \rightarrow Pl(B)$ is called a **plausibility function**.
Two-dimensional representation

- The uncertainty on a proposition $B$ is represented by two numbers: $\text{Bel}(B)$ and $\text{Pl}(B)$, with $\text{Bel}(B) \leq \text{Pl}(B)$.
- The intervals $[\text{Bel}(B), \text{Pl}(B)]$ have maximum length when $m$ is the vacuous mass function. Then,

$$[\text{Bel}(B), \text{Pl}(B)] = [0, 1]$$

for all subset $B$ of $\Omega$, except $\emptyset$ and $\Omega$.
- The intervals $[\text{Bel}(B), \text{Pl}(B)]$ are reduced to points when the focal sets of $m$ are singletons ($m$ is then said to be Bayesian); then,

$$\text{Bel}(B) = \text{Pl}(B)$$

for all $B$, and $\text{Bel}$ is a probability measure.
Consonant mass functions

- If the focal sets of $m$ are nested ($A_1 \subset A_2 \subset \ldots \subset A_n$), $m$ is said to be consonant. $Pl$ is then a possibility measure:

$$Pl(A \cup B) = \max(Pl(A), Pl(B))$$

for all $A, B \subseteq \Omega$ and $Bel$ is the dual necessity measure, i.e.,

$$Bel(A \cap B) = \min(Bel(A), Bel(B))$$

- The corresponding possibility distribution is the contour function

$$pl(\omega) = Pl(\{\omega\}) \text{ for all } \omega \in \Omega.$$  

- We have

$$Pl(A) = \max_{\omega \in A} pl(\omega) \text{ for all } A \subseteq \Omega.$$
Belief-probability transformations

- It may be useful to transform a mass function $m$ into a probability distribution for approximation or decision-making.
- Two main belief-probability transformations:
  1. Plausibility-probability transformation
     \[ p_m(\omega) = \frac{pl(\omega)}{\sum_{\omega \in \Omega} pl(\omega)} \]
     Property: $p_{m_1 \oplus m_2} = p_{m_1} \oplus p_{m_2}$.
  2. Pignistic transformation
     \[ betp_m(\omega) = \sum_{A \ni \omega} \frac{m(A)}{|A|} \]
     Property: The corresponding probability measure $Betp_m$ is the center of mass of all probability measures $P$ such that $Bel \leq P \leq Pl$. 

A probability measure is precise, in so far as it represents the uncertainty of the proposition $\omega \in A$ by a single number $P(A)$.

In contrast, a mass function is imprecise (it assigns probabilities to subsets).

As a result, in DS theory, the uncertainty about a subset $A$ is represented by two numbers $(Bel(A), Pl(A))$, with $Bel(A) \leq Pl(A)$.

This model has some connections with possibility theory (it is more general) and with rough set theory, in which a set is approximated by lower and upper approximations, due to coarseness of a knowledge base.
Outline

1. Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2. Evidential classification
   - Evidential $K$-NN rule
   - Evidential neural network classifier
   - Decision analysis

3. Evidential clustering
   - Evidential partition
   - Evidential $c$-means
   - EVCLUS
   - Ek-NNclus
Dempster’s rule
Murder example continued

The first item of evidence gave us: $m_1(\{Peter, John\}) = 0.8$, $m_1(\Omega) = 0.2$.

New piece of evidence: a blond hair has been found.

There is a probability 0.6 that the room has been cleaned before the crime: $m_2(\{John, Mary\}) = 0.6$, $m_2(\Omega) = 0.4$.

How to combine these two pieces of evidence?
If interpretations $s_1 \in S_1$ and $s_2 \in S_2$ both hold, then $X \in \Gamma_1(s_1) \cap \Gamma_2(s_2)$.

If the two pieces of evidence are independent, then the probability that $s_1$ and $s_2$ both hold is $P_1({s_1})P_2({s_2})$.

If $\Gamma_1(s_1) \cap \Gamma_2(s_2) = \emptyset$, we know that $s_1$ and $s_2$ cannot hold simultaneously.

The joint probability distribution on $S_1 \times S_2$ must be conditioned to eliminate such pairs.
Dempster’s rule

Definition

- Let $m_1$ and $m_2$ be two mass functions and

$$\kappa = \sum_{B \cap C = \emptyset} m_1(B)m_2(C)$$

their degree of conflict.

- If $K < 1$, then $m_1$ and $m_2$ can be combined as

$$ (m_1 \oplus m_2)(A) = \frac{1}{1 - \kappa} \sum_{B \cap C = A} m_1(B)m_2(C), \quad \forall A \neq \emptyset, $$

and $(m_1 \oplus m_2)(\emptyset) = 0.$
Dempster’s rule

Properties

- Commutativity, associativity. Neutral element: \( m_\Omega \).
- Generalization of intersection: if \( m_A \) and \( m_B \) are logical mass functions and \( A \cap B \neq \emptyset \), then
  \[
  m_A \oplus m_B = m_{A \cap B}
  \]
- Generalization of probabilistic conditioning: if \( m \) is a Bayesian mass function and \( m_A \) is a logical mass function, then \( m \oplus m_A \) is a Bayesian mass function corresponding to the conditioning of \( m \) by \( A \).
- Notation for conditioning (special case):
  \[
  m \oplus m_A = m(\cdot | A).
  \]
- Contour functions: if \( pl \) and \( pl' \) are the contour functions of \( m \) and \( m' \), and \( pl'' \) is the contour function of \( m'' = m \oplus m' \), then
  \[
  pl'' \propto pl \cdot pl'.
  \]
Outline

1. Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2. Evidential classification
   - Evidential K-NN rule
   - Evidential neural network classifier
   - Decision analysis

3. Evidential clustering
   - Evidential partition
   - Evidential c-means
   - EVCLUS
   - Ek-NNclus
Problem formulation

- A decision problem can be formalized by defining:
  - A set of acts $\mathcal{A} = \{a_1, \ldots, a_s\}$;
  - A set of states of the world $\Omega$;
  - A loss function $L : \mathcal{A} \times \Omega \rightarrow \mathbb{R}$, such that $L(a, \omega)$ is the loss incurred if we select act $a$ and the true state is $\omega$.

- Bayesian framework
  - Uncertainty on $\Omega$ is described by a probability measure $P$;
  - Define the risk of each act $a$ as the expected loss if $a$ is selected:
    \[
    R_P(a) = \mathbb{E}_P[L(a, \cdot)] = \sum_{\omega \in \Omega} L(a, \omega) P(\{\omega\}).
    \]
  - Select an act with minimal risk.

- Extension when uncertainty on $\Omega$ is described by a belief function?
Lower and upper risks

- **Lower expectation** (optimistic):

  \[
  R(a) = \sum_{A \subseteq \Omega} m(A) \min_{\omega \in A} L(a, \omega)
  \]

- **Upper expectation** (pessimistic):

  \[
  \overline{R}(a) = \sum_{A \subseteq \Omega} m(A) \max_{\omega \in A} L(a, \omega)
  \]
Compromising between the lower and upper risks

- **Hurwicz criterion:**
  \[
  R_\rho(a) = (1 - \rho)\underline{R}(a) + \rho\overline{R}(a),
  \]
  where \( \rho \in [0, 1] \) is a **pessimism index** describing the attitude of the decision maker in the face of ambiguity.

- **Pignistic expectation**
  \[
  R_{bet}(a) = \sum_{A \subseteq \Omega} \left( m(A) \frac{1}{|A|} \sum_{\omega \in A} L(a, \omega) \right) 
  = \sum_{\omega \in \Omega} L(a, \omega) betp_m(\omega)
  \]
Decision strategies

- Minimization of lower risk (optimistic):
  \[ a \succeq a' \text{ iff } R(a) \leq R(a') \]

- Minimization of upper risk (pessimistic):
  \[ a \succeq a' \text{ iff } \overline{R}(a) \leq \overline{R}(a') \]

- Hurwicz criterion:
  \[ a \succeq a' \text{ iff } R_\rho(a) \leq R_\rho(a') \]

- Minimization of pignistic risk:
  \[ a \succeq a' \text{ iff } R_{bet}(a) \leq R_{bet}(a') \]
Interval dominance rule

- Act $a$ dominates $a'$ ($a \succeq a'$) if $\overline{R}(a) \leq \underline{R}(a')$.
- If the intervals $[\underline{R}(a), \overline{R}(a)]$ and $[\underline{R}(a'), \overline{R}(a')]$ intersect, $a$ and $a'$ are not comparable. We thus get a partial preorder.
- The interval dominance rule selects the set of non dominated acts (the set of acts $a$ such that no act is strictly preferred to $a$)

$$\{ a \in A | \forall a' \in A, \neg (a' \succ a) \}$$
Outline

1. Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2. Evidential classification
   - Evidential $K$-NN rule
   - Evidential neural network classifier
   - Decision analysis

3. Evidential clustering
   - Evidential partition
   - Evidential c-means
   - EVCLUS
   - Ek-NNclus
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 $\mathbf{x} \in \mathbb{R}^p$
- A class label $y \in \Omega$

Problem: given a learning set $\mathcal{L} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, predict the class label of a new instance described by $\mathbf{x}$.
Outline

1. Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2. Evidential classification
   - Evidential K-NN rule
   - Evidential neural network classifier
   - Decision analysis

3. Evidential clustering
   - Evidential partition
   - Evidential c-means
   - EVCLUS
   - Ek-NNclus
Let $\mathcal{N}_K(\mathbf{x}) \subset \mathcal{L}$ denote the set of the $K$ nearest neighbors of $\mathbf{x}$ in $\mathcal{L}$, based on some distance measure.

Each $\mathbf{x}_i \in \mathcal{N}_K(\mathbf{x})$ can be considered as a piece of evidence regarding the class of $\mathbf{x}$.

The strength of this evidence decreases with the distance $d_i$ between $\mathbf{x}$ and $\mathbf{x}_i$. 

**Principle**
Definition

- If \( y_i = \omega_k \), the evidence of \((x_i, y_i)\) can be represented by
  \[
  m_i(\{\omega_k\}) = \varphi_k(d_i)
  \]
  \[
  m_i(\{\omega_\ell\}) = 0, \quad \forall \ell \neq k
  \]
  \[
  m_i(\Omega) = 1 - \varphi(d_i)
  \]

  where \( \varphi_k, k = 1, \ldots, c \) are decreasing functions from \([0, +\infty)\) to \([0, 1]\) such that \( \lim_{d \to +\infty} \varphi_k(d) = 0 \)

- The evidence of the \( K \) nearest neighbors of \( x \) is pooled using Dempster's rule of combination
  \[
  m = \bigoplus_{x_i \in \mathcal{N}_K(x)} m_i
  \]

- Decision: any of the decision rules mentioned in the first part.
- With 0-1 losses and no rejection, the optimistic, pessimistic and pignistic rules yield the same decisions.
Learning

- Choice of functions $\varphi_k$: for instance, $\varphi_k(d) = \alpha \exp(-\gamma_k d^2)$.
- Parameters $\gamma_1, \ldots, \gamma_c$ can be optimized (see below).
- Parameter $\gamma = (\gamma_1, \ldots, \gamma_c)$ can be learnt from the data by minimizing the following cost function

$$C(\gamma) = \sum_{i=1}^{n} \sum_{k=1}^{c} (pl_{(-i)}(\omega_k) - t_{ik})^2,$$

where

- $pl_{(-i)}$ is the contour function obtained by classifying $x_i$ using its $K$ nearest neighbors in the learning set.
- $t_{ik} = 1$ is $y_i = k$, $t_{ik} = 0$ otherwise.
- Function $C(\gamma)$ can be minimized by an iterative nonlinear optimization algorithm.
Computation of $pl_{(-i)}$

- Contour function from each neighbor $x_j \in \mathcal{N}_K(x_i)$:

$$pl_j(\omega_k) = \begin{cases} 
1 & \text{if } y_j = \omega_k \\
1 - \varphi_k(d_{ij}) & \text{otherwise}
\end{cases}, \quad k = 1, \ldots, c$$

- Contour function of the combined mass function

$$pl_{(-i)}(\omega_k) \propto \prod_{x_j \in \mathcal{N}_K(x_i)} (1 - \varphi_k(d_{ij}))^{1-t_{jk}}$$

where $t_{jk} = 1$ if $y_j = \omega_k$ and $t_{jk} = 0$ otherwise

- It can be computed in time proportional to $K|\Omega|$
Example 1: Vehicles dataset

- The data were used to distinguish 3D objects within a 2-D silhouette of the objects.
- Four classes: bus, Chevrolet van, Saab 9000 and Opel Manta.
- 846 instances, 18 numeric attributes.
- The first 564 objects are training data, the rest are test data.
Vehicles datasets: result

Vehicles data

- EK-NN
- Voting K-NN
Example 2: Ionosphere dataset

- This dataset was collected by a radar system and consists of phased array of 16 high-frequency antennas with a total transmitted power of the order of 6.4 kilowatts.
- The targets were free electrons in the ionosphere. "Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not.
- There are 351 instances and 34 numeric attributes. The first 175 instances are training data, the rest are test data.
Ionosphere datasets: result

Ionosphere data

- **EK-NN**
- **voting K-NN**

Test error rate

- EK-NN
- Voting K-NN

Ionosphere datasets: result
Implementation in R

library("evclass")

data("ionosphere")
xapp<-ionosphere$x[1:176,]
yapp<-ionosphere$y[1:176]
xtst<-ionosphere$x[177:351,]
ytst<-ionosphere$y[177:351]

opt<-EkNNfit(xapp,yapp,K=10)
class<-EkNNval(xapp,yapp,xtst,K=10,ytst,opt$param)

> class$err
0.07428571
> table(ytst,class$ypred)
ytst 1 2
 1 106 6
 2 7 56
Partially supervised data

We now consider a learning set of the form

\[ \mathcal{L} = \{(x_i, m_i), i = 1, \ldots, n\} \]

where

- \( x_i \) 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
  \[
  m = \bigoplus_{x_i \in N_K(x)} m_i'
  \]
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 $x_i$, the expert opinions were modeled as a mass function $m_i$.
- $n = 200$ learning patterns, 300 test patterns

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>0.30</td>
<td>0.30</td>
<td>0.31</td>
<td>0.27</td>
</tr>
<tr>
<td>11</td>
<td>0.29</td>
<td>0.30</td>
<td>0.29</td>
<td>0.26</td>
</tr>
<tr>
<td>13</td>
<td>0.31</td>
<td>0.30</td>
<td>0.31</td>
<td>0.26</td>
</tr>
</tbody>
</table>
Outline

1 Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2 Evidential classification
   - Evidential $K$-NN rule
   - Evidential neural network classifier
   - Decision analysis

3 Evidential clustering
   - Evidential partition
   - Evidential $c$-means
   - EVCLUS
   - Ek-NNclus
The learning set is summarized by \( r \) prototypes.

Each prototype \( p_i \) has membership degree \( u_{ik} \) to each class \( \omega_k \), with \( \sum_{k=1}^{c} u_{ik} = 1 \).

Each prototype \( p_i \) is a piece of evidence about the class of \( x \), whose reliability decreases with the distance \( d_i \) between \( x \) and \( p_i \).
Mass function induced by prototype $p_i$:

$$m_i(\{\omega_k\}) = \alpha_i u_{ik} \exp(-\gamma_i d_i^2), \quad k = 1, \ldots, c$$

$$m_i(\Omega) = 1 - \alpha_i \exp(-\gamma_i d_i^2)$$

Combination:

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

The computation of $m_i$ requires $O(rp)$ arithmetic operations (where $p$ denotes the number of inputs), and the combination can be performed in $O(rc)$ operations. Hence, the overall complexity is $O(r(p + c))$ operations to compute the output for one input pattern.

The combined mass function $m$ has as focal sets the singletons $\{\omega_k\}$, $k = 1, \ldots, c$ and $\Omega$. 
Neural network implementation
The parameters are the
- The prototypes $p_i$, $i = 1, \ldots, r$ ($rp$ parameters)
- The membership degrees $u_{ik}$, $i = 1, \ldots, r$, $k = 1, \ldots, c$ ($rc$ parameters)
- The $\alpha_i$ and $\gamma_i$, $i = 1 \ldots, r$ (2$r$ parameters).

Let $\theta$ denote the vector of all parameters. It can be estimated by minimizing a cost function such as

$$C(\theta) = \sum_{i=1}^{n} (pl_{ik} - t_{ik})^2 + \mu \sum_{i=1}^{r} \alpha_i$$

where $pl_{ik}$ is the output plausibility for instance $i$ and class $k$, $t_{ik} = 1$ if $y_i = k$ and $t_{ik} = 0$ otherwise, and $\mu$ is a regularization coefficient (hyperparameter).

The hyperparameter $\mu$ can be optimized by cross-validation.
library("evclass")

data(glass)
xtr<-glass$x[1:89,]
ytr<-glass$y[1:89]
xtst<-glass$x[90:185,]
ytst<-glass$y[90:185]

param0<-proDSinit(xtr,ytr,nproto=7)
fit<-proDSfit(x=xtr,y=ytr,param=param0)
val<-proDSval(xtst,fit$param,ytst)

> print(val$err)
0.3333333

> table(ytst,val$ypred)
ytst 1 2 3 4
1 30 6 4 0
2 6 27 1 3
3 4 3 1 0
4 0 5 0 6
Results on the Iris data

Mass on \{\omega_1\}
Results on the Iris data

Mass on \{\omega_2\}
Results on the Iris data

Mass on $\{\omega_3\}$
Results on the Iris data

Mass on $\Omega$
Results on the Iris data

Plausibility of $\{\omega_1\}$
Results on the Iris data

Plausibility of $\{\omega_2\}$

![Graph showing Plausibility of $\{\omega_2\}$ for Petal.Length vs. Petal.Width. The graph includes contour lines representing different plausibility values.]
Results on the Iris data

Plausibility of $\{\omega_3\}$
Outline

1. Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2. Evidential classification
   - Evidential K-NN rule
   - Evidential neural network classifier
   - Decision analysis

3. Evidential clustering
   - Evidential partition
   - Evidential c-means
   - EVCLUS
   - Ek-NNclus
Simple decision setting

- To formalize the decision problem, we need to define:
  - The acts
  - The loss matrix
- For instance, let the acts be
  - \( a_k = \text{assignment to class } \omega_k, \ k = 1, \ldots, c \)
- And the loss matrix (for \( c = 3 \))

<table>
<thead>
<tr>
<th></th>
<th>( a_1 )</th>
<th>( a_2 )</th>
<th>( a_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \omega_1 )</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>( \omega_2 )</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>( \omega_3 )</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

- \( R(a_i) = 1 - Pl(\{\omega_i\}) \) and \( \overline{R}(a_i) = 1 - Bel(\{\omega_i\}) \).
- The optimistic, pessimistic and pignistic decision rules yield the same result.
### Implementation in R

```r
param0 <- proDSinit(x, y, 6)
fit <- proDSfit(x, y, param0)

val <- proDSval(xtst, fit$param)
L <- 1 - diag(c)
D <- decision(val$m, L = L, rule = 'upper')
```
Decision regions (Iris data)
Decision with rejection

Let the acts now be:
- $a_k =$ assignment to class $\omega_k$, $k = 1, \ldots, c$
- $a_0 =$ rejection

And the loss matrix (for $c = 3$)

<table>
<thead>
<tr>
<th></th>
<th>$a_1$</th>
<th>$a_2$</th>
<th>$a_3$</th>
<th>$a_0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_1$</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>$\lambda_0$</td>
</tr>
<tr>
<td>$\omega_2$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>$\lambda_0$</td>
</tr>
<tr>
<td>$\omega_3$</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>$\lambda_0$</td>
</tr>
</tbody>
</table>
Implementation in R

```r
param0 <- proDSinit(x, y, 6)
fit <- proDSfit(x, y, param0)

val <- proDSval(xtst, fit$param)
L <- cbind(1 - diag(c), rep(0.3, c))
D1 <- decision(val$m, L = L, rule = 'upper')
D2 <- decision(val$m, L = L, rule = 'lower')
D3 <- decision(val$m, L = L, rule = 'pignistic')
D4 <- decision(val$m, L = L, rule = 'hurwicz', rho = 0.5)
```
Decision regions (Iris data)

Lower risk
Decision regions (Iris data)

Upper risk
Decision regions (Iris data)

Pignistic risk

![Decision regions graph](image)

- Petal.Length vs Petal.Width
- Decision regions for different classes:
  - Red: Class 1
  - Blue: Class 2
  - Green: Class 3

Thierry Denœux
Classification and clustering using Belief functions

IUKM 2016, November 29, 2016 65 / 145
Decision regions (Iris data)

Hurwicz strategy ($\rho = 0.5$)
Decision with rejection and novelty detection

- Assume that there exists an unknown class $\omega_u$, not represented in the learning set.
- Let the acts now be:
  - $a_k = \text{assignment to class } \omega_k$, $k = 1, \ldots, c$
  - $a_u = \text{assignment to class } \omega_u$
  - $a_0 = \text{rejection}$
- And the loss matrix:

\[
\begin{array}{c|ccccc}
 & a_1 & a_2 & a_3 & a_0 & a_u \\
\hline
\omega_1 & 0 & 1 & 1 & \lambda_0 & \lambda_u \\
\omega_2 & 1 & 0 & 1 & \lambda_0 & \lambda_u \\
\omega_3 & 1 & 1 & 0 & \lambda_0 & \lambda_u \\
\omega_u & 1 & 1 & 1 & \lambda_0 & 0 \\
\end{array}
\]
Implementation in R

```r
param0 <- proDSinit(x, y, 6)
fit <- proDSfit(x, y, param0)

val <- proDSval(xtst, fit$param)
L <- cbind(1 - diag(c), rep(0.3, c), rep(0.32, c))
L <- rbind(L, c(1, 1, 1, 0.3, 0))
D1 <- decision(val$m, L = L, rule = 'lower')
D2 <- decision(val$m, L = L, rule = 'pignistic')
D3 <- decision(val$m, L = L, rule = 'hurwicz', rho = 0.5)
```
Decision regions (Iris data)
Decision regions (Iris data)
Decision regions (Iris data)
T. Denœux.
A k-nearest neighbor classification rule based on Dempster-Shafer theory.

T. Denœux.
A neural network classifier based on Dempster-Shafer theory.

T. Denœux.
Analysis of evidence-theoretic decision rules for pattern classification.

C. Lian, S. Ruan and T. Denœux.
An evidential classifier based on feature selection and two-step classification strategy.
References on classification II

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

C. Lian, S. Ruan and T. Denœux.
Dissimilarity metric learning in the belief function framework.
Outline

1. Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2. Evidential classification
   - Evidential K-NN rule
   - Evidential neural network classifier
   - Decision analysis

3. Evidential clustering
   - Evidential partition
   - Evidential c-means
   - EVCLUS
   - Ek-NNclus
Clustering

$n$ objects described by
- Attribute vectors $x_1, \ldots, x_n$ (attribute data) or
- Dissimilarities (proximity data).

Goals:
1. Discover groups in the data;
2. Assess the uncertainty in group membership.
Hard and soft clustering concepts

**Hard clustering:** no representation of uncertainty. Each object is assigned to one and only one group. Group membership is represented 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 $\sum_{k=1}^{c} u_{ik} = 1$. The $u_{ik}$’s can be interpreted as probabilities.

**Fuzzy clustering with noise cluster:** the above equality is replaced by $\sum_{k=1}^{c} u_{ik} \leq 1$. The number $1 - \sum_{k=1}^{c} u_{ik}$ is interpreted as a degree of membership (or probability of belonging to) to a noise cluster.
Evidential clustering

Possibilistic clustering: the $u_{ik}$ are free to take any value in $[0, 1]$. Each number $u_{ik}$ is interpreted as a degree of possibility that object $i$ belongs to group $k$.

Rough clustering: each cluster $\omega_k$ is characterized by a lower approximation $\omega_k$ and an upper approximation $\overline{\omega}_k$, with $\omega_k \subseteq \omega_k$; the membership of object $i$ to cluster $k$ is described by a pair $(u_{ik}, \overline{u}_{ik}) \in \{0, 1\}^2$, with $u_{ik} \leq \overline{u}_{ik}$, $\sum_{k=1}^{c} u_{ik} \leq 1$ and $\sum_{k=1}^{c} \overline{u}_{ik} \geq 1$. 
Clustering and belief functions

<table>
<thead>
<tr>
<th>clustering structure</th>
<th>uncertainty framework</th>
</tr>
</thead>
<tbody>
<tr>
<td>fuzzy partition</td>
<td>probability theory</td>
</tr>
<tr>
<td>possibilistic partition</td>
<td>possibility theory</td>
</tr>
<tr>
<td>rough partition</td>
<td>(rough) sets</td>
</tr>
<tr>
<td>?</td>
<td>belief functions</td>
</tr>
</tbody>
</table>

As belief functions extend probabilities, possibilities and sets, could the theory of belief functions provide a more general and flexible framework for cluster analysis?

Objectives:
- **Unify** the various approaches to clustering
- **Achieve** a richer and more accurate representation of uncertainty
- **New clustering algorithms** and new tools to compare and combine clustering results.
Outline

1. Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2. Evidential classification
   - Evidential $k$-NN rule
   - Evidential neural network classifier
   - Decision analysis

3. Evidential clustering
   - Evidential partition
   - Evidential $c$-means
   - EVCLUS
   - $Ek$-NNclus
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 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 $\Omega$.
- The n-tuple $\mathcal{M} = (m_1, \ldots, m_n)$ is called a credal partition.
- Example:

![Butterfly data](image_url)

<table>
<thead>
<tr>
<th>Credal partition</th>
<th>$\emptyset$</th>
<th>${\omega_1}$</th>
<th>${\omega_2}$</th>
<th>${\omega_1, \omega_2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_3$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$m_5$</td>
<td>0</td>
<td>0.5</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>$m_6$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$m_{12}$</td>
<td>0.9</td>
<td>0</td>
<td>0.1</td>
<td>0</td>
</tr>
</tbody>
</table>
Relationship with other clustering structures

- More general
  - $m_i$ unnormalized Bayesian
    - Fuzzy partition with a noise cluster
  - $m_i$ Bayesian
    - Fuzzy partition

- Less general
  - $m_i$ Bayesian
  - Fuzzy partition
  - Possibilistic partition
  - Hard partition
  - Rough partition

- $m_i$ certain
- $m_i$ consonant
- $m_i$ logical
- $m_i$ general
Rough clustering as a special case

\[ m(\{\omega_1\}) = 1 \quad m(\{\omega_1, \omega_2\}) = 1 \quad m(\{\omega_2\}) = 1 \]

Lower approximations

Upper approximations

\[ \omega_1^L \quad \omega_2^L \quad \omega_1^U \quad \omega_2^U \]
Summarization of a credal partition

More complex
- Unnormalized pignistic/plausibility transformation

Credal partition
- Interval dominance or maximum mass

Fuzzy partition with a noise cluster
- Normalization

Fuzzy partition
- Maximum probability

Possibilistic partition
- Contour function

Hard partition
- Maximum plausibility

Rough partition

Less complex
From evidential to rough clustering

- For each $i$, let $A_i \subseteq \Omega$ be the set of non dominated clusters

\[ A_i = \{ \omega \in \Omega | \forall \omega' \in \Omega, Bel_i^*(\{\omega'\}) \leq Pl_i^*(\{\omega\}) \}, \]

where $Bel_i^*$ and $Pl_i^*$ are the normalized belief and plausibility functions.

- Lower approximation:

\[ u_{ik} = \begin{cases} 1 & \text{if } A_i = \{\omega_k\} \\ 0 & \text{otherwise.} \end{cases} \]

- Upper approximation:

\[ \bar{u}_{ik} = \begin{cases} 1 & \text{if } \omega_k \in A_i \\ 0 & \text{otherwise.} \end{cases} \]

- The outliers can be identified separately as the objects for which $m_i(\emptyset) \geq m_i(A)$ for all $A \neq \emptyset$. 
Algorithms

1. **Evidential c-means (ECM)**: (Masson and Denoeux, 2008):
   - Attribute data
   - HCM, FCM family

2. **EVCLUS** (Denoeux and Masson, 2004; Denoeux et al., 2016):
   - Attribute or proximity (possibly non metric) data
   - Multidimensional scaling approach

3. **EK-NNclus** (Denoeux et al, 2015)
   - Attribute or proximity data
   - Searches for the most plausible partition of a dataset
Outline

1 Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2 Evidential classification
   - Evidential $K$-NN rule
   - Evidential neural network classifier
   - Decision analysis

3 Evidential clustering
   - Evidential partition
   - Evidential $c$-means
     - EVCLUS
     - Ek-NNclus
Problem: generate a credal partition \( M = (m_1, \ldots, m_n) \) from attribute data \( X = (x_1, \ldots, x_n), \ x_i \in \mathbb{R}^p \).

Generalization of hard and fuzzy \( c \)-means algorithms:
- Each cluster is represented by a prototype.
- Cyclic coordinate descent algorithm: optimization of a cost function alternatively 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 \( d_{ik} = ||x_i - v_k|| \) subject to the constraints \( \sum_k u_{ik} = 1 \) for all \( i \).

- Alternate optimization algorithm:

\[ v_k = \frac{\sum_{i=1}^{n} u_{ik}^{\beta} x_i}{\sum_{i=1}^{n} u_{ik}^{\beta}} \]

\[ u_{ik} = \frac{d_{ik}^{-2/\beta} - 2}{\sum_{\ell=1}^{c} d_{i\ell}^{-2/\beta} - 2}. \]
Each cluster $\omega_k$ represented by a prototype $v_k$.
Each nonempty set of clusters $A_j$ represented by a prototype $\bar{v}_j$ defined as the center of mass of the $v_k$ for all $\omega_k \in A_j$.

Basic ideas:
- For each nonempty $A_j \in \Omega$, $m_{ij} = m_i(A_j)$ should be high if $x_i$ is close to $\bar{v}_j$.
- The distance to the empty set is defined as a fixed value $\delta$. 

ECM algorithm
Principle
ECM algorithm: objective criterion

Define the nonempty focal sets \( \mathcal{F} = \{A_1, \ldots, A_f\} \subseteq 2^\Omega \setminus \{\emptyset\} \).

Minimize

\[
J_{ECM}(M, V) = \sum_{i=1}^{n} \sum_{j=1}^{f} |A_j|^\alpha m_{ij}^\beta d_{ij}^2 + \sum_{i=1}^{n} \delta^2 m_{i\emptyset}^\beta
\]

subject to the constraints \( \sum_{j=1}^{f} m_{ij} + m_{i\emptyset} = 1 \) for all \( i \).

Parameters:

- \( \alpha \) controls the specificity of mass functions (default: 1)
- \( \beta \) controls the hardness of the credal partition (default: 2)
- \( \delta \) controls the proportion of data considered as outliers

\( J_{ECM}(M, V) \) can be iteratively minimized with respect to \( M \) and to \( V \).
ECM algorithm: update equations

Update of $M$:

$$m_{ij} = \frac{c_j^{-\alpha/(\beta-1)} d_{ij}^{-2/(\beta-1)}}{\sum_{k=1}^{f} c_k^{-\alpha/(\beta-1)} d_{ik}^{-2/(\beta-1)} + \delta^{-2/(\beta-1)}},$$

for $i = 1, \ldots, n$ and $j = 1, \ldots, f$, and

$$m_{i\emptyset} = 1 - \sum_{j=1}^{f} m_{ij}, \quad i = 1, \ldots, n$$

Update of $V$: solve a linear system of the form

$$HV = B,$$

where $B$ is a matrix of size $c \times p$ and $H$ a matrix of size $c \times c$. 
```r
library(evclust)
data('butterfly')
n<-nrow(butterfly)

clus<-ecm(butterfly[,1:2],c=2,delta=sqrt(20))
```
Butterfly dataset

![Butterfly data](image-url)
data("fourclass")
clus<-ecm(fourclass[,1:2],c=4,type='pairs',delta=5)

plot(clus,X=fourclass[,1:2],ytrue=fourclass[,3],Outliers = TRUE, approx=2)
4-class data set
Handling a large number of clusters

- If no restriction is imposed on the focal sets, the number of parameters to be estimated in evidential clustering grows exponentially with the number $c$ of clusters, which makes it intractable unless $c$ is small.

- If we allow masses to be assigned to all pairs of clusters, the number of focal sets becomes proportional to $c^2$, which is manageable for moderate values of $c$ (say, until 10), but still impractical for larger $n$.

- Idea: assign masses only to pairs of contiguous clusters.
Method

1. In the first step, ECM is run in the basic configuration, with focal sets of cardinalities 0, 1 and (optionally) $c$. A credal partition $\mathcal{M}_0$ is obtained.

2. The similarity between each pair of clusters $(\omega_j, \omega_\ell)$ is computed as

$$S(j, \ell) = \sum_{i=1}^{n} p_{ij} p_{i\ell},$$

where $p_{ij}$ and $p_{i\ell}$ are the normalized plausibilities that object $i$ belongs, respectively, to clusters $j$ and $\ell$. We then determine the set $P_K$ of pairs $\{\omega_j, \omega_\ell\}$ that are mutual $K$ nearest neighbors.

3. ECM is run again, starting from the previous evidential partition $\mathcal{M}_0$, and adding as focal sets the pairs in $P_K$. 

Thierry Denœux
Classification and clustering using Belief functions
Example in R: step 1

data(s2)
clus<-ecm(x=s2,c=15,type='simple',Omega=FALSE,delta=1,disp=FALSE)
plot(x=clus,X=s2,Outliers = TRUE)
Result after Step 1
Example in R: steps 2 and 3

```r
P <- createPairs(clus, k=2)

clus1 <- ecm(x=s2, c=15, type='pairs', Omega=FALSE, pairs=P$pairs, g0=clus$g, delta=1, disp=FALSE)

plot(x=clus1, X=s2, Outliers = TRUE, approx=2)
```
Final result
Determining the number of groups

- If a proper number of groups is chosen, the prototypes will cover the clusters and most of the mass will be allocated to singletons of $\Omega$.
- On the contrary, if $c$ is too small or too high, the mass will be distributed to subsets with higher cardinality or to $\emptyset$.

**Nonspecificity** of a mass function:

$$N(m) \triangleq \sum_{A \in 2^\Omega \setminus \emptyset} m(A) \log_2 |A| + m(\emptyset) \log_2 |\Omega|$$

- Proposed **validity index** of a credal partition:

$$N^*(c) \triangleq \frac{1}{n \log_2(c)} \sum_{i=1}^n \left[ \sum_{A \in 2^\Omega \setminus \emptyset} m_i(A) \log_2 |A| + m_i(\emptyset) \log_2(c) \right]$$
Example (Four-class dataset)

```
C<-2:7
N<-rep(0,length(C))
for(k in 1:length(C)){
    clus<-ecm(fourclass[,1:2],c=C[k],type='pairs',alpha=2,
                delta=5,disp=FALSE)
    N[k]<-clus$N
}
plot(C,N,type='b',xlab='c',ylab='nonspecificity')
```
Results for the 4-class dataset

![Graph showing nonspecificity as a function of c]

The graph above illustrates the nonspecificity for different values of c. As c increases, the nonspecificity decreases, indicating improved clustering performance. The values shown are for c=2 to c=7, with nonspecificity values ranging from 0.18 to 0.28.
Constrained Evidential $c$-means

- In some cases, we may have some prior knowledge about the group membership of some objects.

Such knowledge may take the form of instance-level constraints of two kinds:

1. **Must-link** (ML) constraints, which specify that two objects certainly belong to the same cluster;
2. **Cannot-link** (CL) constraints, which specify that two objects certainly belong to different clusters.

- How to take into account such constraints?
Modified cost-function

- To take into account ML and CL constraints, we can modify the cost function of ECM as

$$J_{CECM}(M, V) = (1 - \xi)J_{ECM}(M, V) + \xi J_{CONST}(M)$$

with

$$J_{CONST}(M) = \frac{1}{|M| + |C|} \left[ \sum_{(x_i, x_j) \in M} pl_{ij}(\neg S) + \sum_{(x_i, x_j) \in C} pl_{ij}(S) \right]$$

where

- $M$ and $C$ are, respectively, the sets of ML and CL constraints.
- $pl_{ij}(S)$ and $pl_{ij}(\neg S)$ are computed from the pairwise mass function $m_{ij}$

Minimizing $J_{CECM}(M, V)$ w.r.t. $M$ is a quadratic programming problem.
Active learning

- ML and CL constraints are sometimes given in advance, but they can sometimes be elicited from the user using an active learning strategy.
- For instance, we may select pairs of objects such that
  - The first object is classified with high uncertainty (e.g., an object such that $m_i$ has high nonspecificity);
  - The second object is classified with low uncertainty (e.g., an object that is close to a cluster center).
- The user is then provided with this pair of objects, and enters either a ML or a CL constraint.
Results

Glass data

Ionosphere data

Average Rand Index computed on 100 trials
Rand Index obtained with Active Learning
Other variants of ECM

Relational Evidential $c$-Means (RECM) for (metric) proximity data (Masson and Denœux, 2009).

ECM with adaptive metrics to obtain non-spherical clusters (Antoine et al., 2012). Specially useful with CECM.

Spatial Evidential C-Means (SECM) for image segmentation (Lelandais et al., 2014).

Credal $c$-means (CCM): different definition of the distance between a vector and a meta-cluster (Liu et al., 2014).

Median evidential $c$-means (MECM): different cost criterion, extension of the median hard and fuzzy $c$-means (Zhou et al., 2015).
Outline

1. Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2. Evidential classification
   - Evidential $K$-NN rule
   - Evidential neural network classifier
   - Decision analysis

3. Evidential clustering
   - Evidential partition
   - Evidential $c$-means
   - EVCLUS
   - $Ek$-NNclus
Problem: given the dissimilarity matrix $D = (d_{ij})$, how to build a “reasonable” credal partition?

We need a model that relates cluster membership to dissimilarities.

Basic idea: “The more similar two objects, the more plausible it is that they belong to the same group”.

How to formalize this idea?
Formalization

- Let $m_i$ and $m_j$ be mass functions regarding the group membership of objects $o_i$ and $o_j$.
- The plausibility of the proposition $S_{ij}$: “objects $o_i$ and $o_j$ belong to the same group” can be shown to be equal to:

$$pl(S_{ij}) = \sum_{A \cap B \neq \emptyset} m_i(A)m_j(B) = 1 - \kappa_{ij}$$

where $\kappa_{ij} =$ degree of conflict between $m_i$ and $m_j$.
- Problem: find a credal partition $M = (m_1, \ldots, m_n)$ such that larger degrees of conflict $\kappa_{ij}$ correspond to larger dissimilarities $d_{ij}$. 
Cost function

- Approach: **minimize the discrepancy** between the dissimilarities \(d_{ij}\) and the degrees of conflict \(\kappa_{ij}\).

- Example of a cost (stress) function:

\[
J(M) = \sum_{i<j} (\kappa_{ij} - \varphi(d_{ij}))^2
\]

where \(\varphi\) is an increasing function from \([0, +\infty)\) to \([0, 1]\), for instance

\[
\varphi(d) = 1 - \exp(-\gamma d^2).
\]
Butterfly example

Data and dissimilarities

Determination of $\gamma$ in $\varphi(d) = 1 - \exp(-\gamma d^2)$: fix $\alpha \in (0, 1)$ and $d_0$ such that, for any two objects $(o_i, o_j)$ with $d_{ij} \geq d_0$, the plausibility that they belong to the same cluster is at least $1 - \alpha$. 

Butterfly data

![Butterfly data plot](image-url)
Butterfly example

Credal partition

Butterfly data

Objects

Masses

$m(\emptyset)$

$m(\omega_1)$

$m(\omega_2)$

$m(\Omega)$

Thierry Denœux

Classification and clustering using Belief functions

IUKM 2016, November 29, 2016 117 / 145
Butterfly example
Shepard diagram
Optimization algorithm

How to minimize $J(M)$? Two methods:

1. Using a gradient descent or quasi-Newton algorithm (slow).
2. Using a cyclic coordinate descent algorithm minimizing $J(M)$ with respect to each $m_i$ at a time.

The latter approach exploits the particular approach of the problem (a quadratic programming problem is solved at each step), and it is thus much more efficient.
Implementation in R

```r
library(evclust)
data(protein)

clus <- kevclus(D=protein$D,c=4,type='simple',d0=max(protein$D))

z <- cmdscale(protein$D,k=2)

plot(clus,X=z,mfrow=c(2,2),ytrue=protein$y,
Outliers=FALSE,approx=1)
```
Proteins data

- Nonmetric dissimilarity matrix derived from the structural comparison of 213 protein sequences.
- Ground truth: 4 classes of globins.
- Only 2 errors.
Proteins data: Shepard diagram
Example with a four-class dataset (2000 objects)
Handling large datasets

- EVCLUS requires to store the whole dissimilarity matrix: it is inapplicable to large proximity data.
- Idea: compute the differences between degrees of conflict and dissimilarities, for only a subset of randomly sampled dissimilarities.
- Let \( j_1(i), \ldots, j_k(i) \) be \( k \) integers sampled at random from the set \( \{1, \ldots, i - 1, i + 1, \ldots, n\} \), for \( i = 1, \ldots, n \). Let \( J_k \) the following stress criterion,

\[
J_k(M) = \sum_{i=1}^{n} \sum_{r=1}^{k} (\kappa_{i,j_{r}(i)} - \delta_{i,j_{r}(i)})^2.
\]

- The calculation of \( J_k(M) \) requires only \( O(nk) \) operations.
- If \( k \) can be kept constant as \( n \) increases, then time and space complexity is reduced from quadratic to linear.
Zongker Digit dissimilarity data

- Similarities between 2000 handwritten digits in 10 classes, based on deformable template matching.
- $k$-EVCLUS was run with $c = 10$ and different values of $k$.
- Parameter $d_0$ was fixed to the 0.3-quantile of the dissimilarities.
- $k$-EVCLUS was run 10 times with random initializations.
Outline

1. Dempster-Shafer theory
   - Mass, belief and plausibility functions
   - Dempster’s rule
   - Decision analysis

2. Evidential classification
   - Evidential $K$-NN rule
   - Evidential neural network classifier
   - Decision analysis

3. Evidential clustering
   - Evidential partition
   - Evidential $c$-means
   - EVCLUS
   - $Ek$-NNclus
Reasoning in the space of all partitions

- Assuming there is a true unknown partition, our frame of discernment should be the set $\mathcal{R}$ of all equivalent relations ($\equiv$ partitions) of the set of $n$ objects.
- But this set is huge!
Can we implement evidential reasoning in such a large space?
Model

- Evidence: $n \times n$ matrix $D = (d_{ij})$ of dissimilarities between the $n$ objects.

- Assumptions
  1. Two objects have all the more chance to belong to the same group, that they are more similar:

     
     \[
     m_{ij}({\{S\}}) = \varphi(d_{ij}), \\
     m_{ij}(\Theta) = 1 - \varphi(d_{ij}),
     \]

     where $\varphi$ is a non-increasing mapping from $[0, +\infty)$ to $[0, 1)$.

  2. The mass functions $m_{ij}$ are independent.

- How to combine these $n(n-1)/2$ mass functions to find the most plausible partition of the $n$ objects?
Evidence combination

- Let $R_{ij}$ denote the set of partitions of the $n$ objects such that objects $o_i$ and $o_j$ are in the same group ($r_{ij} = 1$).
- Each mass function $m_{ij}$ can be **vacuously extended** to the space $R$ of equivalence relations:

$$m_{ij}(\{S\}) \rightarrow R_{ij}$$

$$m_{ij}(\Theta) \rightarrow R$$

- The extended mass functions can then be combined by Dempster’s rule.
- Resulting contour function:

$$pl(R) \propto \prod_{i<j} (1 - \varphi(d_{ij}))^{1-r_{ij}}$$

for any $R \in R$. 
The logarithm of the contour function can be written as

$$\log p(l(R)) = - \sum_{i<j} r_{ij} \log(1 - \varphi(d_{ij})) + C$$

Finding the most plausible partition is thus a binary linear programming problem. It can be solved exactly only for small $n$.

However, the problem can be solved approximately using a heuristic greedy search procedure: the $Ek$-NNclus algorithm.

This is a decision-directed clustering procedure, using the evidential $k$-nearest neighbor ($Ek$-NN) rule as a base classifier.
Example
Toy dataset
Example

Iteration 1
Example
Iteration 1 (continued)
Example
Iteration 2
Example
Iteration 2 (continued)
Example

Result
Ek-NNclus

Starting from a random initial partition, classify each object in turn, using the Ek-NN rule.

The algorithm converges to a local maximum of the contour function $pl(R)$ if $k = n - 1$.

With $k < n - 1$, the algorithm converges to a local maximum of an objective function that approximates $pl(R)$.

Implementation details:

- Number $k$ of neighbors: two to three times $\sqrt{n}$.
- $\varphi(d) = 1 - \exp(-\gamma d^2)$, with $\gamma$ fixed to the inverse of the $q$-quantile of the distances $d_{ij}^2$ between an object and its $k$ NN. Typically, $q \geq 0.5$.
- The number of clusters does not need to be fixed in advance.
Evidential clustering

Ek-NNclus in R

data(fourclass)
x<-fourclass[,1:2]
n<-nrow(x)
y0<-1:n
clus<-EkNNclus(x, D, K=50, y0, ntrials = 1, q = 0.5, p = 1)
plot(x[,1],x[,2],pch=clus$y.pl,col=clus$y.pl)

c<-ncol(clus$mass)-1
plot(x[,1],x[,2],pch=clus$y,col=clus$y.pl,
cex=0.1+2*clus$mass[,c+1])
Example
References on clustering I

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

ECM: An evidential version of the fuzzy c-means algorithm. 

RECM: Relational Evidential c-means algorithm. 

B. Lelandais, S. Ruan, T. Denoeux, P. Vera, I. Gardin.
Fusion of multi-tracer PET images for Dose Painting. 

EVCLUS: Evidential Clustering of Proximity Data. 
T. Denœux, S. Sriboonchitta and O. Kanjanatarakul
Evidential clustering of large dissimilarity data.


T. Denoeux, O. Kanjanatarakul and S. Sriboonchitta.
EK-NNclus: a clustering procedure based on the evidential K-nearest neighbor rule.

Summary

- The theory of belief function has great potential in data analysis and challenging machine learning:
  - 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), or combine data with expert knowledge (constrained clustering)
- R packages evclass and evclust available from CRAN at https://cran.r-project.org/web/packages
The evclass package

**evclass: Evidential Distance-Based Classification**

Different evidential distance-based classifiers, which provide outputs in the form of Dempster-Shafer mass functions. The methods are: the evidential K-nearest neighbor rule and the evidential neural network.

**Version:** 1.1.0  
**Depends:** R (≥ 3.1.0)  
**Imports:** FNN  
**Suggests:** knitr, rmarkdown, datasets  
**Published:** 2016-07-01  
**Author:** Thierry Denœux  
**Maintainer:** Thierry Denœux <tdenoeux at utc.fr>  
**License:** GPL-3  
**NeedsCompilation:** no  
**In views:** MachineLearning  
**CRAN checks:** evclass results  

**Downloads:**

- Reference manual: evclass.pdf  
- Vignettes: Introduction to the evclass package  
- Package source: evclass_1.1.0.tar.gz  
- Windows binaries: r-devel: evclass_1.1.0.zip, r-release: evclass_1.1.0.zip, r-oldrel: evclass_1.1.0.zip  
- OS X Mavericks binaries: r-release: evclass_1.1.0.tgz, r-oldrel: evclass_1.1.0.tgz  
- Old sources: evclass archive
The **evclust package**

**evclust: Evidential Clustering**

Various clustering algorithms that produce a credal partition, i.e., a set of Dempster-Shafer mass functions representing the membership of objects to clusters. The mass functions quantify the cluster-membership uncertainty of the objects. The algorithms are: Evidential c-Means (ECM), Relational Evidential c-Means (RECM), Constrained Evidential c-Means (CECM), EVCLUS and EK-NNclus.

**Version:** 1.0.3  
**Depends:** R (≥ 3.1.0)  
**Imports:** FNN, R.utils, limSolve, Matrix  
**Suggests:** knitr, rmarkdown  
**Published:** 2016-09-04  
**Author:** Thierry Denoeux  
**Maintainer:** Thierry Denoeux <tdenoeux at utc.fr>  
**License:** GPL-3  
**NeedsCompilation:** no  
**In views:** Cluster  
**CRAN checks:** evclust results

https://cran.r-project.org/web/packages