Classification and clustering

Thierry Denœux

Université de technologie de Compiègne, France
Institut Universitaire de France
https://www.hds.utc.fr/~tdenoeux

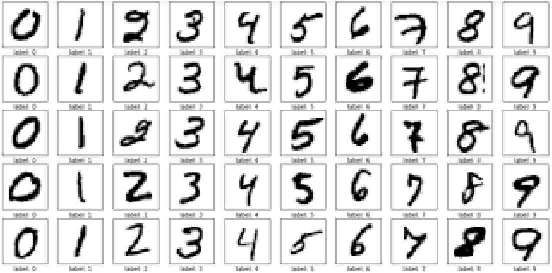
Fifth School on Belief Functions and their Applications
Sienna, Italy, October 29, 2019

Classification

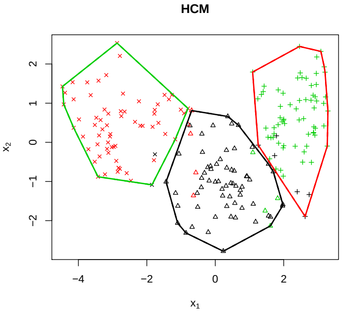
- We consider a population of objects partitioned in c groups (classes). Each object is described by a feature vector $X = (X_1, \ldots, X_d) \in \mathcal{X}$ of d features and a class variable $Y \in \Theta$ indicating group membership.
- Problem: given a learning set $\{(x_i, y_i)\}_{i=1}^n$ containing observations of X and Y for n objects, build a classifier

$$C : \mathcal{X} \longrightarrow \Theta$$

  that predicts the value of Y given X.
- Example: digit recognition, $\mathcal{X} = [0,1]^{16\times 16}$, $\Theta = \{0, \ldots, 9\}$.



Clustering



- *n* objects described by
  - Attribute vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$ (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
   1. Quantify prediction uncertainty (for, e.g., combining several classifiers, or providing the user with richer information about the uncertainty of the classification)
   2. Reveal richer information about the data (clustering problems)
2. Represent uncertainty about the data themselves:
   1. Uncertain/soft class labels (partially supervised learning)
   2. 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. Evidential classifiers directly providing belief functions as outputs:
   - Generalized Bayes theorem, extends the Bayesian classifier when class densities and priors are ill-known (Appriou, 1991; Denœux and Smets, 2008)
   - Distance-based classifiers: evidential K-NN rule (Denœux, 1995), evidential neural network classifier (Denœux, 2000)
   - Neural networks and many other machine learning models are evidential classifiers! (Denœux, 2019)

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 (Denœux and Masson, 2004; Denœux et al., 2016)
2. Extend prototype-based clustering methods such as the hard or fuzzy c-means: Evidential c-means (Masson and Denœux, 2008)
3. Decision-directed clustering using the evidential K-NN classifier: EK-NNclus algorithm (Denœux et al, 2015)

Outline

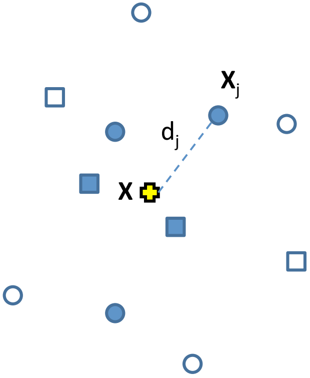
1. Evidential distance-based classifiers
   - Evidential K-NN rule
   - Contextual Discounting Evidential K-NN
   - Evidential neural network classifier
2. Neural networks as evidential classifiers
   - Logistic regression and extensions
   - Binomial classifiers
   - Multinomial classifers
3. Clustering
   - Credal partition
   - EVCLUS

Outline

Outline

1. Evidential distance-based classifiers
   - Evidential K-NN rule
   - Contextual Discounting Evidential K-NN
   - Evidential neural network classifier
2. Neural networks as evidential classifiers
   - Logistic regression and extensions
   - Binomial classifiers
   - Multinomial classifers
3. Clustering
   - Credal partition
   - EVCLUS

Principle



- 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 $\mathbf{x}_j \in \mathcal{N}_K(\mathbf{x})$ can be considered as a piece of evidence regarding the class of $\mathbf{x}$
- The strength of this evidence decreases with the distance d_j between $\mathbf{x}$ and $\mathbf{x}_j$

Definition

- Frame of discernment: $\Theta = \{\theta_1, \ldots, \theta_c\}$.
- The evidence of $(\mathbf{x}_j, y_j)$ can be represented by the following mass function on Θ:

$$\widehat{m}_j(\{\theta_k\}) = \varphi_k(d_j)\, y_{jk}, \quad k = 1, \ldots, c$$
$$\widehat{m}_j(\Theta) = 1 - \varphi_k(d_j)$$

  where
  - $y_{jk} = I(y_j = \theta_k)$
  - φ_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 $\mathbf{x}$ is pooled using Dempster's rule of combination

$$\widehat{m} = \bigoplus_{\mathbf{x}_j \in \mathcal{N}_K(\mathbf{x})} \widehat{m}_j$$

- Decision: maximum plausibility.

Learning

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

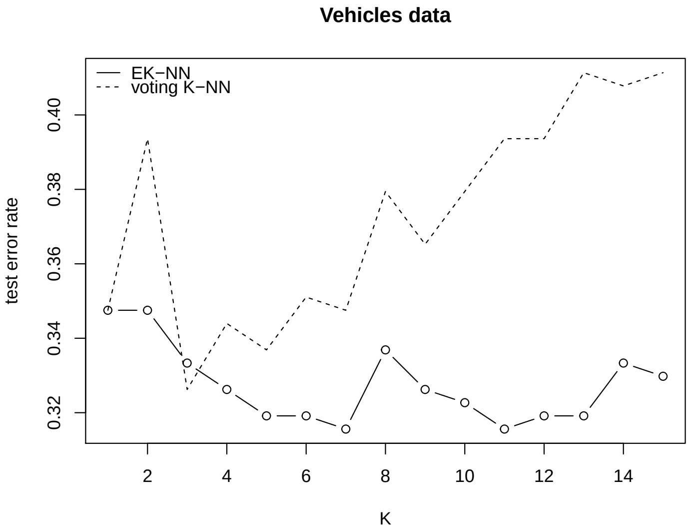
$$C(\boldsymbol{\gamma}) = \sum_{i=1}^{n} \sum_{k=1}^{c} (\widehat{pl}_i(\omega_k) - y_{ik})^2,$$

  where $\widehat{pl}_i$ is the contour function corresponding to $\widehat{m}_i$ computed using the K-NN of observation $\mathbf{x}_i$.
- Function $C(\boldsymbol{\gamma})$ can be minimized by an iterative nonlinear optimization algorithm.

Example: 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



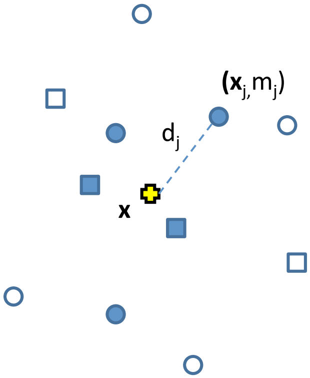
Partially supervised data

- We now consider a learning set of the form

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

  where
  - $\mathbf{x}_i$ 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 (soft label)
- Special cases:
  - $m_i(\{\omega_k\}) = 1$ for all i: supervised data
  - $m_i(\Omega) = 1$ for all i: unsupervised data

Evidential k-NN rule for partially supervised data



- Each mass function m_j is discounted with a rate depending on the distance d_j:

$$\widehat{m}_j(A) = \varphi(d_i)\, m_j(A), \quad \forall A \subset \Theta$$

$$\widehat{m}_j(\Theta) = 1 - \sum_{A \subset \Omega} \widehat{m}_j(A)$$

- The K mass functions $\widehat{m}_i$ are combined using Dempster's rule:

$$\widehat{m} = \bigoplus_{\mathbf{x}_j \in \mathcal{N}_K(\mathbf{x})} \widehat{m}_j$$

Outline

Contextual Discounting Evidential K-NN

- A recent variant introduced by Denoeux and Kanjanatarajul (2019).
- We consider partially labeled data $\mathcal{L} = \{(x_i, m_i)\}_{i=1}^n$.
- The mass function $\widehat{m}_j$ induced by $x_j \in \mathcal{N}_K(x)$ is now obtained from m_j by the contextual discounting operation with discount rates $1 - \beta_k(d_j)$, with

$$\beta_k(d_j) = \alpha \exp(-\gamma_k d_j^2), \quad k = 1, \ldots, c,$$

  with $\alpha \in [0, 1]$ and $\gamma_k \geq 0$, $k = 1, \ldots, c$.
- Combined contour function:

$$\widehat{pl}(\theta_k) \propto \prod_{x_j \in \mathcal{N}_K(x)} \left[1 - \beta_k(d_j) + \beta_k(d_j) pl_j(\theta_k)\right], \quad k = 1, \ldots, c.$$

- $\widehat{pl}$ can be computed, up to a multiplicative constant, in time proportional to the number K of neighbors and the number of c of classes.

Learning

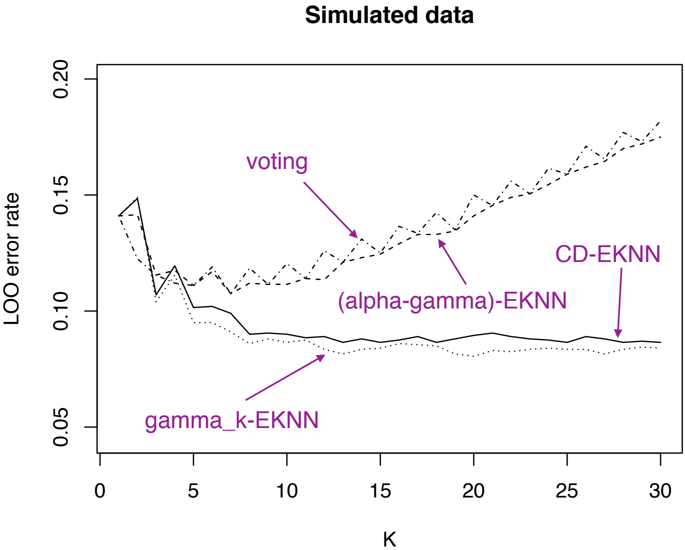
- To learn the parameters $\psi = (\alpha, \gamma_1, \ldots, \gamma_c)$ of the CD-EKNN classifier, we maximize the evidential likelihood function introduced in by Denoeux (2013).
- Case of fully supervised data $\mathcal{L} = \{(x_i, y_i)\}_{i=1}^n$: the conditional likelihood after observing the true class labels $y_1, \ldots, y_n$ is

$$L_c(\psi) = \prod_{i=1}^{n} \prod_{k=1}^{c} \widehat{p}_i(\theta_k)^{y_{ik}} = \prod_{i=1}^{n} \sum_{k=1}^{c} \widehat{p}_i(\theta_k) y_{ik},$$

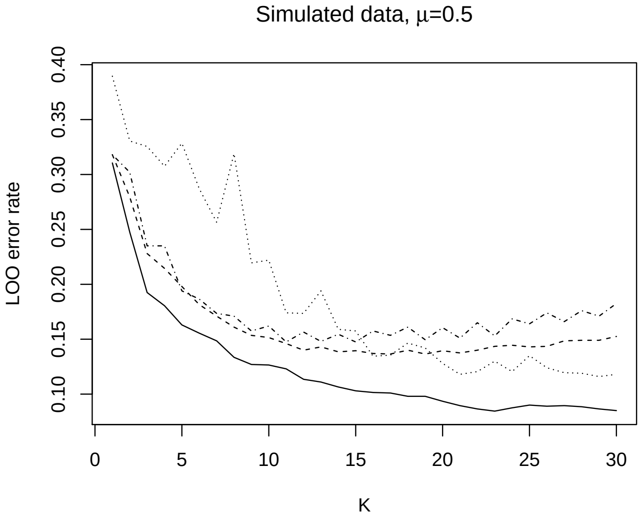
  where $\widehat{p}_i$ be the probability distribution obtained from $\widehat{pl}_i$ by normalization.
- Extension to partially supervised data $\mathcal{L} = \{(x_i, m_i)\}_{i=1}^n$:

$$L_e(\psi) = \prod_{i=1}^{n} \underbrace{\sum_{k=1}^{c} \widehat{p}_i(\theta_k) pl_i(\theta_k)}_{\text{expected plausibility}},$$

Results: simulated data with hard labels



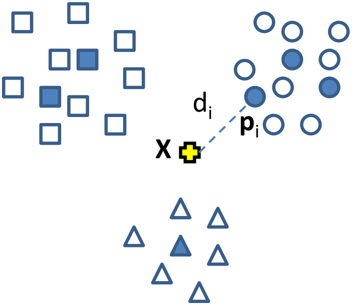
Results: simulated data with soft labels



Outline

1. Evidential distance-based classifiers
   - Evidential K-NN rule
   - Contextual Discounting Evidential K-NN
   - Evidential neural network classifier
2. Neural networks as evidential classifiers
   - Logistic regression and extensions
   - Binomial classifiers
   - Multinomial classifers
3. Clustering
   - Credal partition
   - EVCLUS

Principle



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

Propagation equations

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

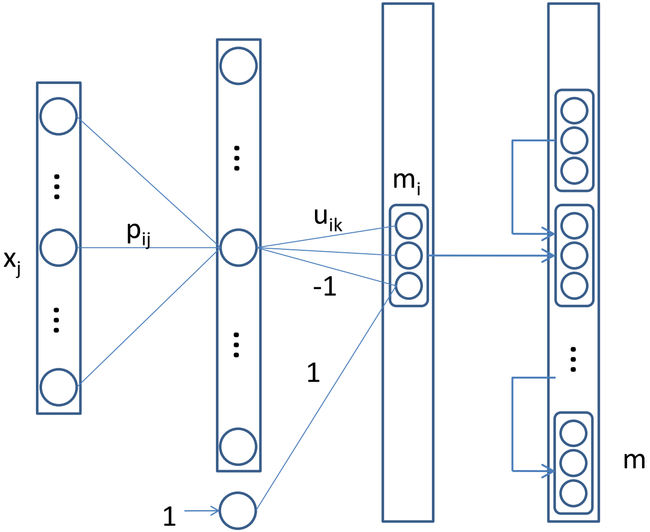
$$m_i(\{\theta_k\}) = \alpha_i u_{ik} \exp(-\gamma_i d_i^2), \quad k = 1, \ldots, c$$
$$m_i(\Theta) = 1 - \alpha_i \exp(-\gamma_i d_i^2)$$

- Combination:

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

- The combined mass function m has as focal sets the singletons $\{\theta_k\}$, $k = 1, \ldots, c$ and Θ.

Neural network implementation



Learning

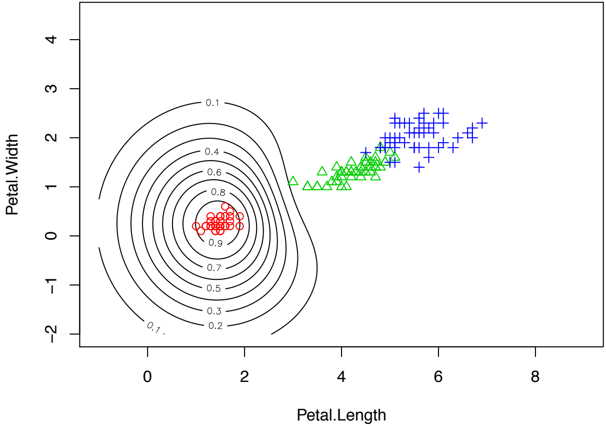
- The parameters are the
  - The prototypes $\mathbf{p}_i$, $i = 1, \ldots, r$ (rp parameters)
  - The membership degrees u_{ik}, $i = 1, \ldots, r$, $k = 1 \ldots, c$ (rc parameters)
  - The α_i and γ_i, $i = 1 \ldots, r$ ($2r$ parameters).
- Let ψ denote the vector of all parameters. It can be estimated by minimizing a cost function such as

$$C(\psi) = \sum_{i=1}^{n} \sum_{k=1}^{c} (pl_{ik} - y_{ik})^2 + \lambda \sum_{i=1}^{r} \alpha_i$$

  where pl_{ik} is the output plausibility for instance i and class k, and μ is a regularization coefficient (hyperparameter).
- The hyperparameter λ can be optimized by cross-validation.

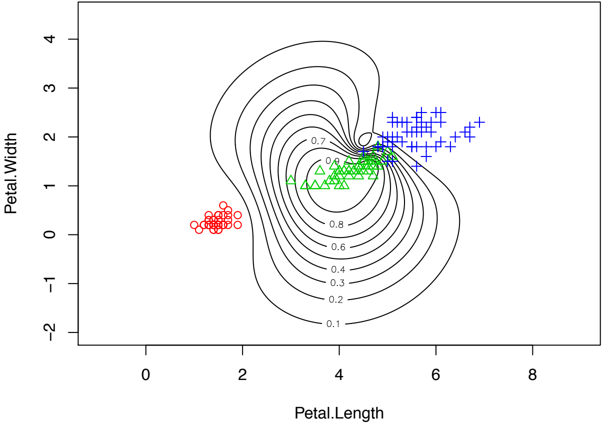
Results on the Iris data

Mass on $\{\theta_1\}$



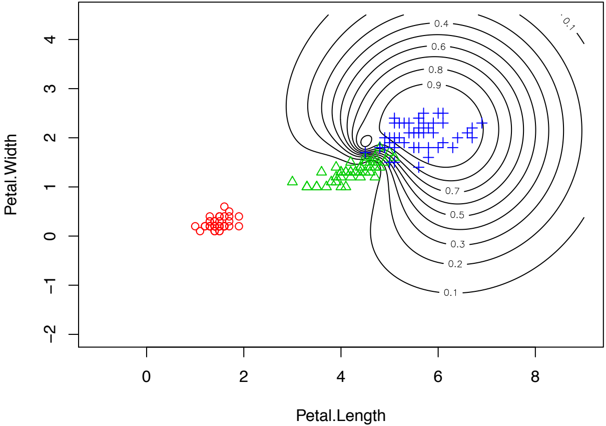
Results on the Iris data

Mass on $\{\theta_2\}$



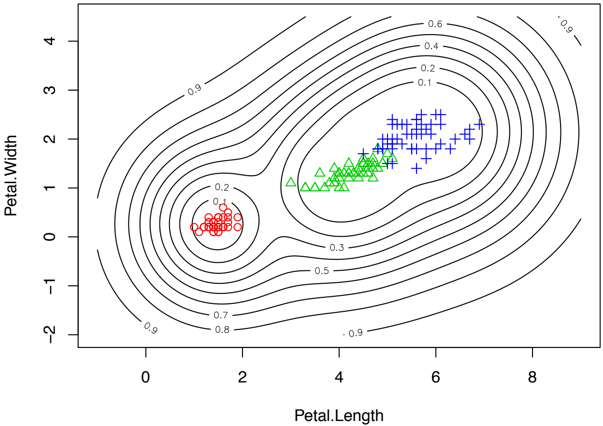
Results on the Iris data

Mass on $\{\theta_3\}$



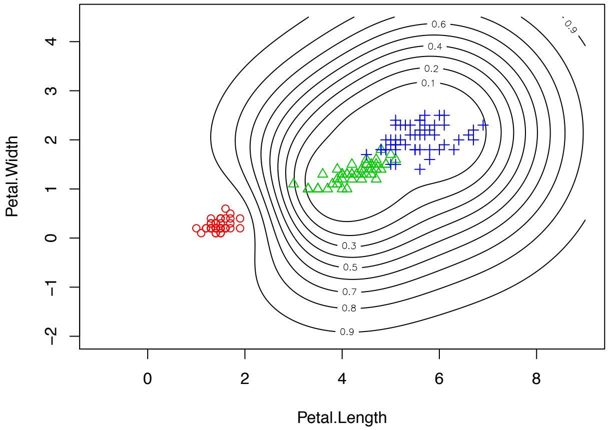
Results on the Iris data

Mass on Θ



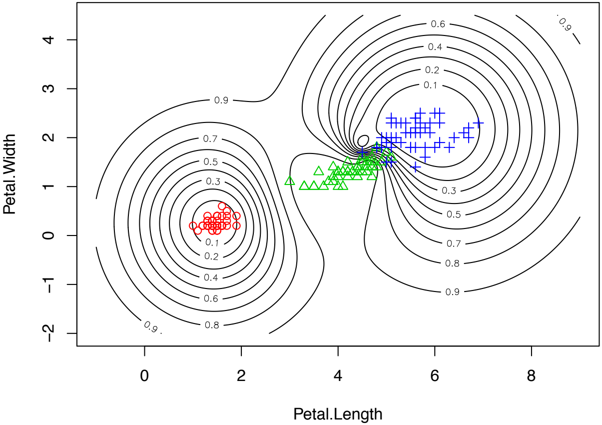
Results on the Iris data

Plausibility of $\{\theta_1\}$



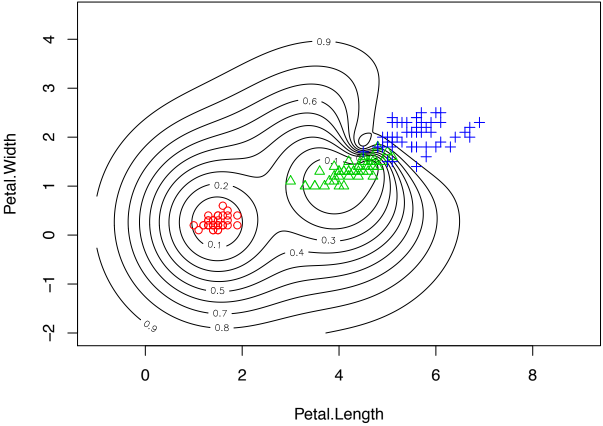
Results on the Iris data

Plausibility of $\{\theta_2\}$



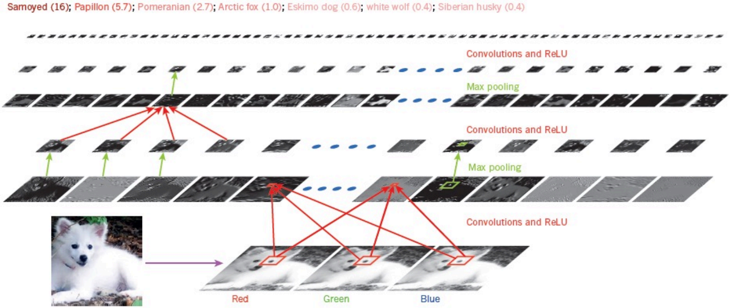
Results on the Iris data

Plausibility of $\{\theta_3\}$



Outline

Deep Learning



(From Le Cun et al., *Nature*, 2015)

- In recent years, applications of Machine Learning (ML) have been flourishing following new developments in deep learning technology.
- A lot of progress has been made in extracting high-order features from data, so as to solve very complex classification problems.

Some challenges

- ML algorithms (and especially deep learning models) are essentially black boxes.
- Major challenges:
  1. Make ML algorithms more transparent so that machine predictions can be interpreted (and trusted) by humans
  2. Assess the uncertainty of the predictions, to make ML algorithms reliable and suitable for safety-critical applications.
- To meet these challenges, we need new perspectives on how classification algorithms actually work.
- One such perspective is provided by the theory of belief functions.

Outline

Binomial Logistic regression

- Consider a binary classification problem with $Y \in \Theta = \{\theta_1, \theta_2\}$.
- Let $p(x)$ denote the probability that $Y = \theta_1$ given that $X = x$.
- (Binomial) Logistic Regression (LR) model:

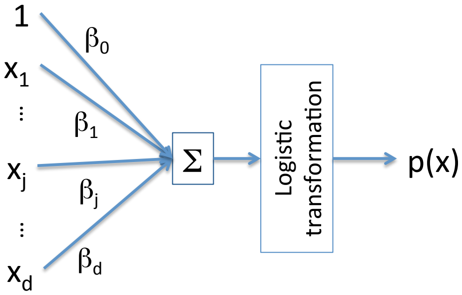
$$\ln \frac{p(x)}{1 - p(x)} = \boldsymbol{\beta}^T x + \beta_0,$$

with $\boldsymbol{\beta} \in \mathbb{R}^d$ and $\beta_0 \in \mathbb{R}$. Equivalently,

$$p(x) = \sigma(\boldsymbol{\beta}^T x + \beta_0),$$

where $\sigma(u) = (1 + \exp(-u))^{-1}$ is the logistic function.

Binomial Logistic Regression (continued)



Given a learning set $\{(x_i, y_i)\}_{i=1}^n$, parameters β and β_0 are usually estimated by minimizing the cross-entropy error function:

$$C(\beta, \beta_0) = -\sum_{i=1}^{n} \{I(y_i = \theta_1) \ln p(x_i) + I(y_i = \theta_2) \ln [1 - p(x_i)]\}$$

Multinomial Logistic Regression

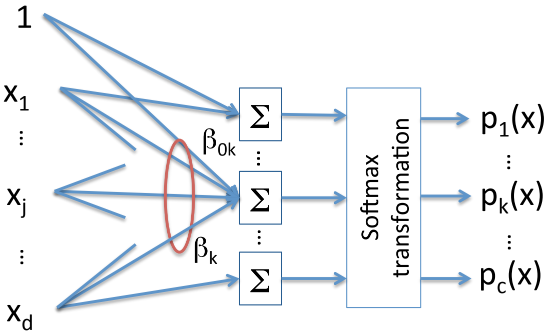
- Multinomial logistic regression (MLR) extends binomial LR to $c > 2$ classes by assuming the following model:

$$\ln p_k(x) = \boldsymbol{\beta}_k^T x + \beta_{k0} + \gamma,$$

  where $p_k(x) = \mathbb{P}(Y = \theta_k | X = x)$, $\boldsymbol{\beta}_k \in \mathbb{R}^d$, $\beta_{k0} \in \mathbb{R}$ and $\gamma \in \mathbb{R}$ is a constant that does not depend on k.
- The posterior probability of class θ_k can then be expressed using the softmax transformation as

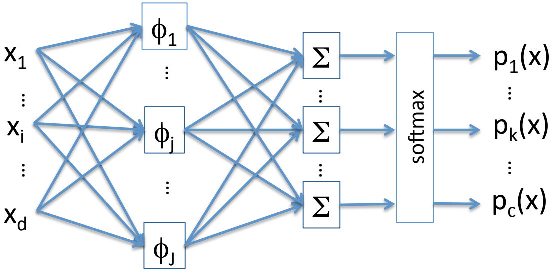
$$p_k(x) = \frac{\exp(\boldsymbol{\beta}_k^T x + \beta_{k0})}{\sum_{l=1}^K \exp(\boldsymbol{\beta}_l^T x + \beta_{l0})}.$$

Multinomial Logistic Regression (continued)



Parameters $(\boldsymbol{\beta}_k, \beta_{k0})$, $k = 1\ldots, c$ can be estimated by minimizing the cross-entropy as in the binomial case.

Nonlinear generalized LR classifiers



- LR classifiers are linear classifiers (they separate classes in feature space by hyperplanes).
- LR can be applied to transformed features $\phi_j(x)$, $j = 1, \ldots, J$, where the ϕ_j's are nonlinear mappings from $\mathbb{R}^d$ to $\mathbb{R}$. We get nonlinear generalized LR classifiers.
- Both the new features $\phi_j(x)$ and the coefficients (β_k, β_{k0}) are usually learnt simultaneously by minimizing some cost function.

Generalized LR models

Generalized additive models:

$$\phi_j(x) = \varphi_j(x_j)$$

Radial basis function networks:

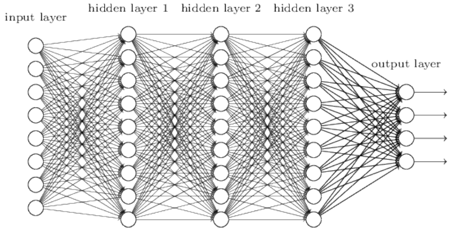
$$\phi_j(x) = \varphi(\|x - v_j\|)$$

Support vector machines:

$$\phi_j(x) = \mathcal{K}(x, x_j)$$

Multilayer feedforward neural networks (NNs)

Multilayer feedforward neural networks



- Feedforward NNs are models composed of elementary computing units (or "neurons") arranged in layers. Each layer computes a vector of new features as functions of the outputs from the previous layer as

$$\phi_j^{(l)} = h\left(w_j^{(l)T}\phi^{(l-1)} + w_{j0}^{(l)}\right), \quad j = 1, \ldots, J_l,$$

  where $\phi^{(l-1)} \in \mathbb{R}^{J_{l-1}}$ is the vector of outputs from the previous layer.
- For c-class classification, the output layer is typically a softmax layer with c output units.

Relation with DS theory?

- LR and NN models seem totally unrelated to DS theory.
- Yet...

Outline

Features as evidence

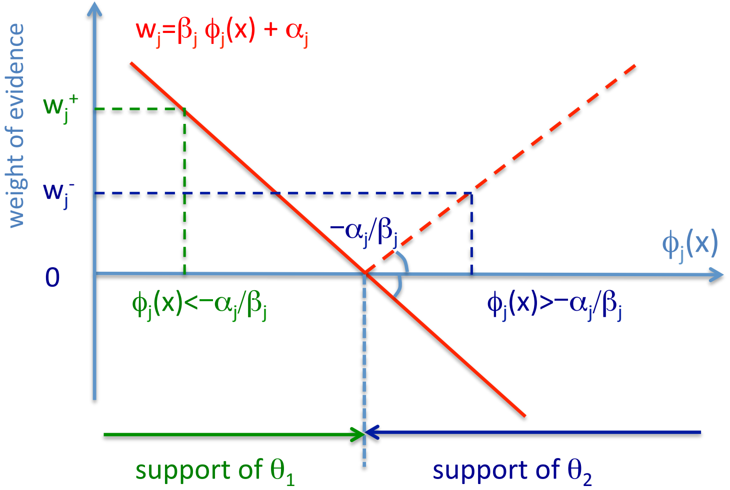
- Consider a binary classification problem with $c = 2$ classes in $\Theta = \{\theta_1, \theta_2\}$. Let $\phi(x) = (\phi_1(x), \ldots, \phi_J(x))$ be a vector of J features.
- Each feature value $\phi_j(x)$ is a piece of evidence about the class $Y \in \Theta$ of the instance under consideration.
- Assume that this evidence points either to θ_1 or θ_2 depending on the sign of

$$w_j := \beta_j \phi_j(x) + \alpha_j,$$

where β_j and α_j are two coefficients:

  - If $w_j \geq 0$, feature ϕ_j supports class θ_1 with weight of evidence w_j
  - If $w_j < 0$, feature ϕ_j supports class θ_2 with weight of evidence $-w_j$

Features as evidence (continued)



Feature-based latent mass function

Under this model, the consideration of feature ϕ_j induces a simple mass function

$$m_j = \{\theta_1\}^{w_j^+} \oplus \{\theta_2\}^{w_j^-},$$

where

- $w_j^+ = \max(0, w_j)$ is the positive part of w_j and
- $w_j^- = \max(0, -w_j)$ is the negative part.

Combined latent mass function

Assuming that the values of the J features can be considered as independent pieces of evidence, the feature-based latent mass functions can be combined by Dempster's rule:

$$
\begin{aligned}
m &= \bigoplus_{j=1}^{J} \left(\{\theta_1\}^{w_j^+} \oplus \{\theta_2\}^{w_j^-} \right) \\
&= \left(\bigoplus_{j=1}^{J} \{\theta_1\}^{w_j^+} \right) \oplus \left(\bigoplus_{j=1}^{J} \{\theta_2\}^{w_j^-} \right) \\
&= \{\theta_1\}^{w^+} \oplus \{\theta_2\}^{w^-},
\end{aligned}
$$

where

- $w^+ := \sum_{j=1}^{J} w_j^+$ is the total weight of evidence supporting θ_1
- $w^- := \sum_{j=1}^{J} w_j^-$ is the total weight of evidence supporting θ_2.

Expression of *m*

$$m(\{\theta_1\}) = \frac{[1 - \exp(-w^+)]\exp(-w^-)}{1-\kappa}$$

$$m(\{\theta_2\}) = \frac{[1 - \exp(-w^-)]\exp(-w^+)}{1-\kappa}$$

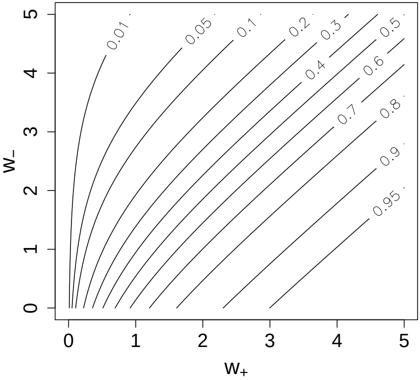
$$m(\Theta) = \frac{\exp(-w^+ - w^-)}{1-\kappa}$$

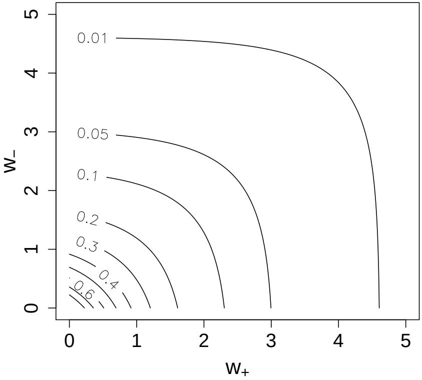
where κ is the degree of conflict:

$$\kappa = [1 - \exp(-w^+)][1 - \exp(-w^-)]$$

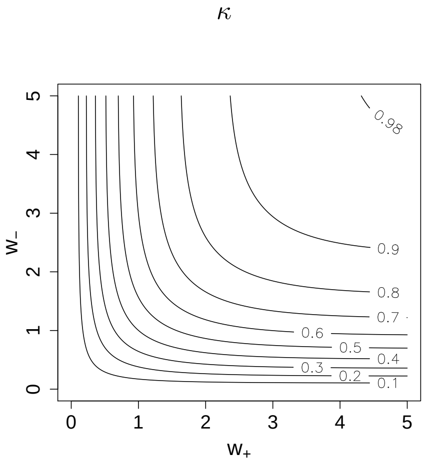
$m(\{\theta_1\})$ and $m(\Theta)$ vs. weights of evidence

$m(\{\theta_1\})$ $m(\Theta)$





Degree of conflict vs. weights of evidence



Normalized plausibilities

The normalized plausibility of class θ_1 as

$$\frac{Pl(\{\theta_1\})}{Pl(\{\theta_1\}) + Pl(\{\theta_2\})} = \frac{m(\{\theta_1\}) + m(\Theta)}{m(\{\theta_1\}) + m(\{\theta_2\}) + 2m(\Theta)}$$

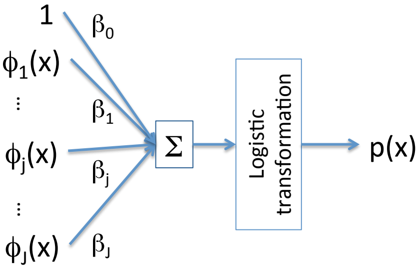
$$= \underbrace{\frac{1}{1 + \exp[-(\boldsymbol{\beta}^T \phi(x) + \beta_0)]}}_{\text{logistic transformation}} = p(x)$$

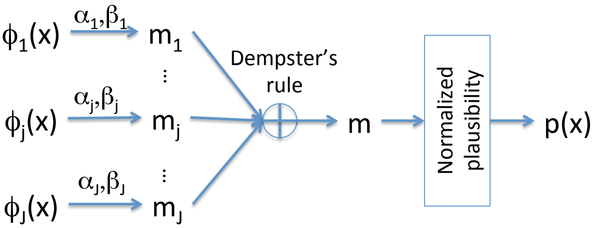
with $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_J)$ and $\beta_0 = \sum_{j=1}^{J} \alpha_j$.

Proposition

The normalized plausibilities are equal to the posterior class probabilities of the binomial LR model: the two models are equivalent.

Two Views of Binomial Logistic Regression





Parameter identification

- As explained before, parameters $\beta_0, \beta_1, \ldots, \beta_J$ can be estimated by maximizing the likelihood. Let $\widehat{\beta}_0, \widehat{\beta}_1, \ldots, \widehat{\beta}_J$ be the corresponding MLEs.
- However, the DS model has J more additional parameters $\alpha_1, \ldots, \alpha_J$ linked to β_0 by the relation $\sum_{i=1}^{J} \alpha_j = \beta_0$: the problem is underdetermined.
- Solution: find the parameter values $\alpha_1^*, \ldots, \alpha_J^*$ that give us the least informative mass function.
- The least informative mass function is defined as the one based on the smallest weights of evidence.

Minimizing the sum of squared weights of evidence

- Let $\{(x_i, y_i)\}_{i=1}^n$ be the learning set and let $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_J)$.
- The values α_j^* minimizing the sum of squared weights of evidence can be found by solving the following minimization problem:

$$\min f(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \sum_{j=1}^{J} \left(\widehat{\beta}_j \phi_j(x_i) + \alpha_j \right)^2$$

subject to $\sum_{j=1}^J \alpha_j = \widehat{\beta}_0$.

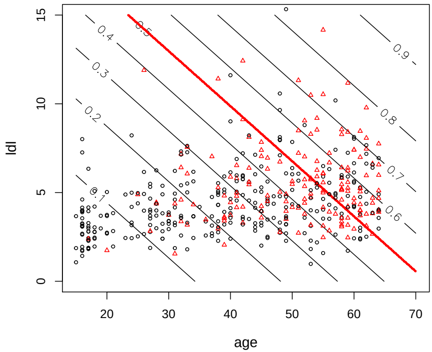
- Solution:

$$\alpha_j^* = \frac{\widehat{\beta}_0}{J} + \frac{1}{J} \sum_{q=1}^{J} \widehat{\beta}_q \mu_q - \widehat{\beta}_j \mu_j$$

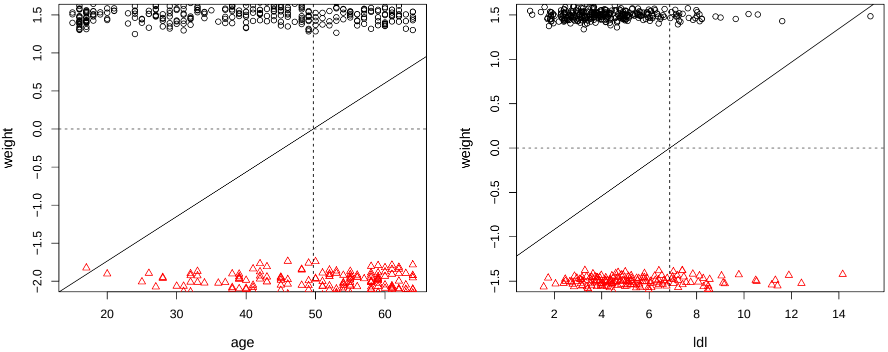
with $\mu_j = \frac{1}{n} \phi_j(x_i)$.

Example

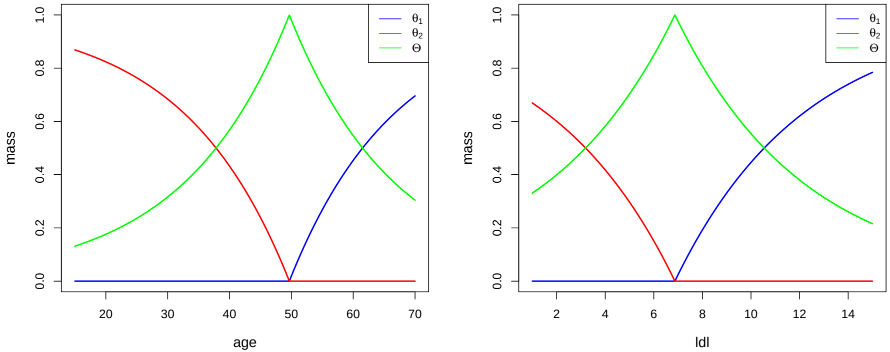
- Data about the intensity of ischemic heart disease risk factors in a rural area of South Africa. Population: white males between 15 and 64. Response variable: presence or absence of myocardial infarction (MI).
- Two variables: age and LDL ("bad" cholesterol).



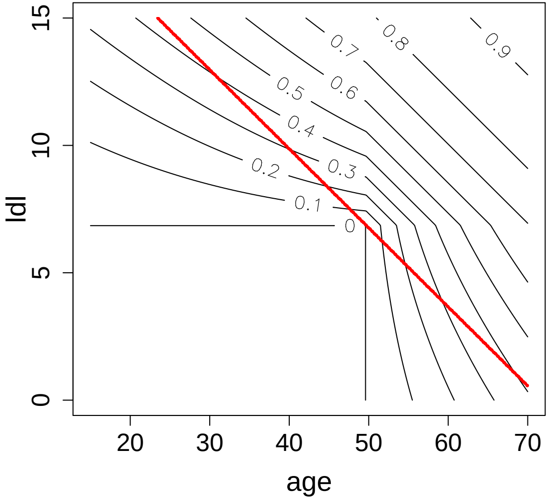
Weights of evidence



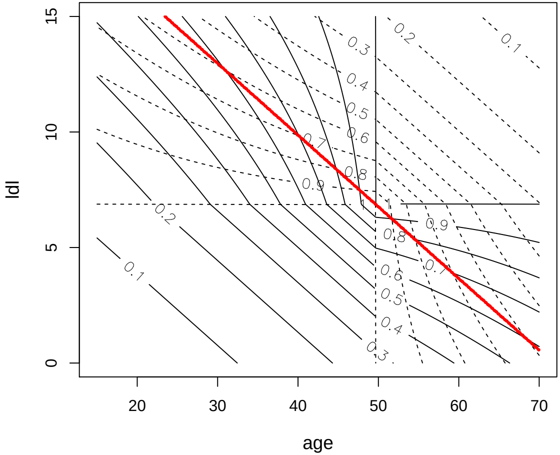
Feature mass functions



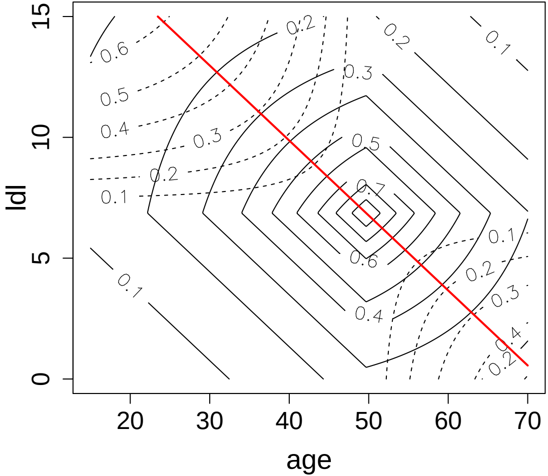
Degrees of belief (positive class)



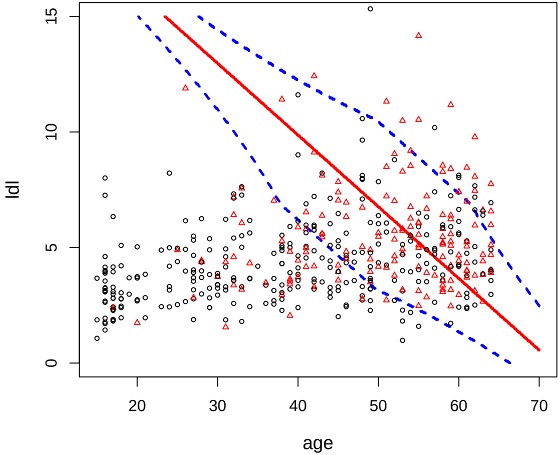
Degrees of Plausibility (positive class)



Mass on Θ and degree of conflict



Decision regions



Outline

Model

- Let $\Theta = \{\theta_1, \ldots, \theta_c\}$ with $c > 2$.
- Each feature ϕ_j now induces c simple mass functions $m_{j1}, \ldots, m_{jc}$.
- Mass function m_{jk} points either to the singleton $\{\theta_k\}$ or to its complement $\overline{\{\theta_k\}}$, depending on the sign of

$$w_{jk} = \beta_{jk}\phi_j(x) + \alpha_{jk},$$

where $(\beta_{jk}, \alpha_{jk})$, $k = 1, \ldots, c$, $j = 1, \ldots, J$ are parameters.

- Expression of m_{jk}:

$$m_{jk} = \{\theta_k\}^{w_{jk}^+} \oplus \overline{\{\theta_k\}}^{w_{jk}^-}$$

Combined latent mass function

- The latent mass function induced by feature ϕ_j is

$$m_j = \bigoplus_{k=1}^{c} \left(\{\theta_k\}^{w_{jk}^+} \oplus \overline{\{\theta_k\}}^{w_{jk}^-} \right).$$

- Assuming the evidence from the J features to be independent, the combined mass function is

$$m = \bigoplus_{j=1}^{J} \bigoplus_{k=1}^{c} \left(\{\theta_k\}^{w_{jk}^+} \oplus \overline{\{\theta_k\}}^{w_{jk}^-} \right)$$
$$= \bigoplus_{k=1}^{c} \left(\{\theta_k\}^{w_k^+} \oplus \overline{\{\theta_k\}}^{w_k^-} \right),$$

where

- $w_k^+ = \sum_{j=1}^J w_{jk}^+$ is the total weight of evidence for class θ_k
- $w_k^- = \sum_{j=1}^J w_{jk}^-$ is the total weight of evidence against class θ_k

Link with multinomial logistic regression

The normalized plausibility of class θ_k is:

$$\frac{Pl(\{\theta_k\})}{\sum_{l=1}^{c} Pl(\{\theta_l\})} = \underbrace{\frac{\exp\left(\sum_{j=1}^{J} \beta_{jk}\phi_j(x) + \beta_{0k}\right)}{\sum_{l=1}^{c} \exp\left(\sum_{j=1}^{J} \beta_{jl}\phi_j(x) + \beta_{0l}\right)}}_{\text{softmax transformation}} = p_k(x),$$

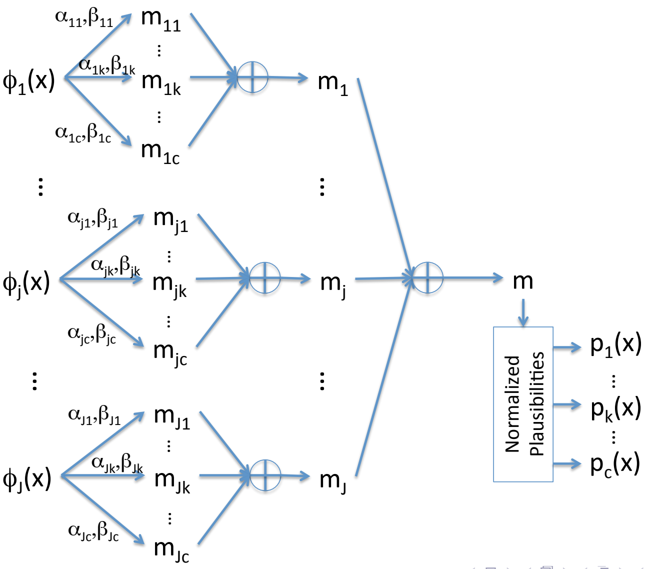
with

$$\beta_{0k} = \sum_{j=1}^{J} \alpha_{jk}.$$

Proposition

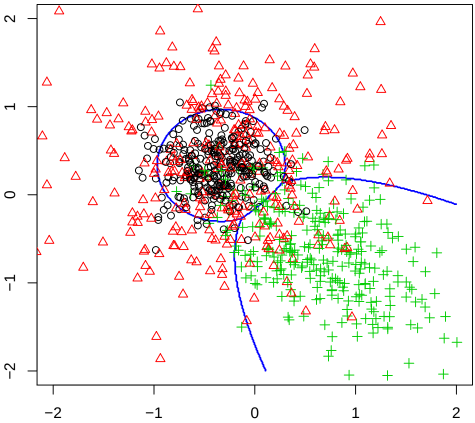
The normalized plausibilities are equal to the posterior class probabilities of the multinomial LR model: the two models are equivalent.

Multinomial Logistic Regression: DS view



Example

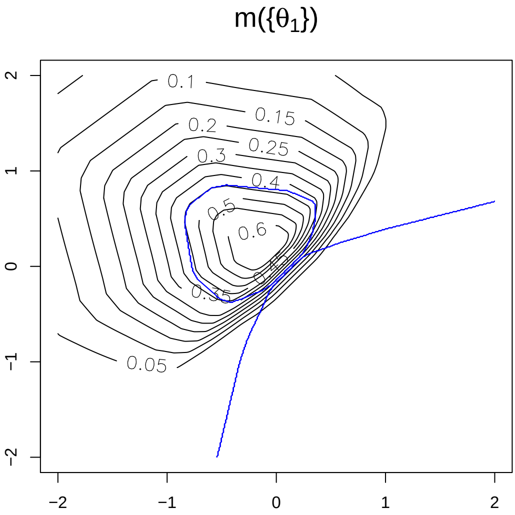
Dataset: 900 instances, 3 equiprobable classes with Gaussian distributions



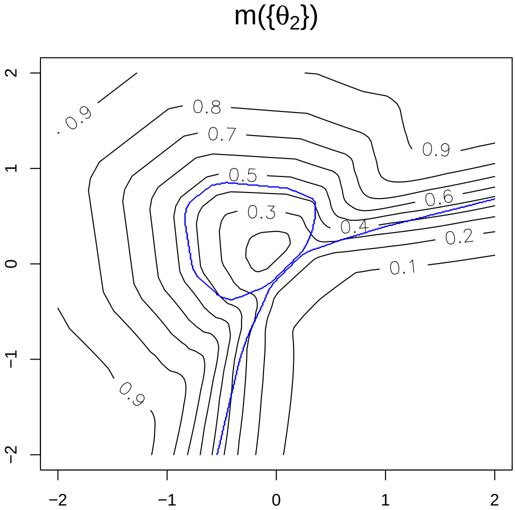
NN model

- NN with 2 layers of 20 and 10 neurons
- ReLU activation functions in hidden layers, softmax output layer
- Batch learning, minibatch size=100
- L_2 regularization in the last layer ($\lambda = 1$).

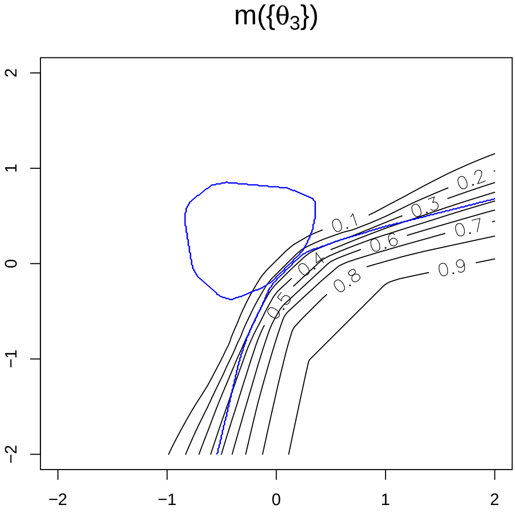
Mass on $\{\theta_1\}$



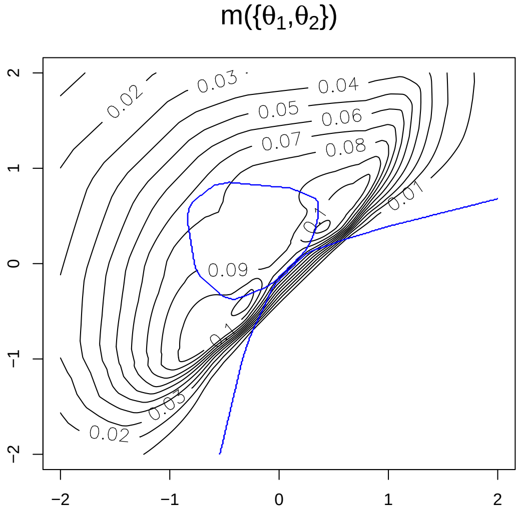
Mass on $\{\theta_2\}$



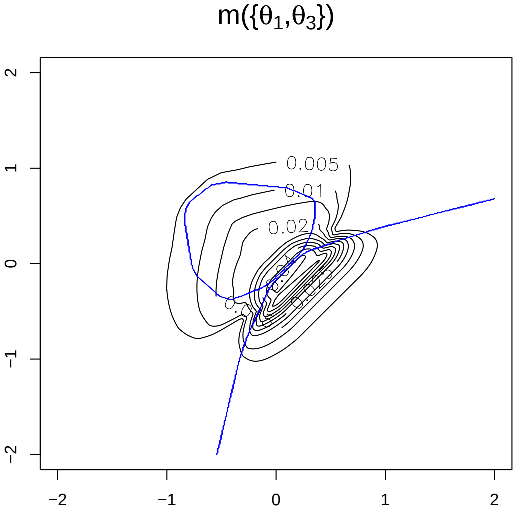
Mass on $\{\theta_3\}$



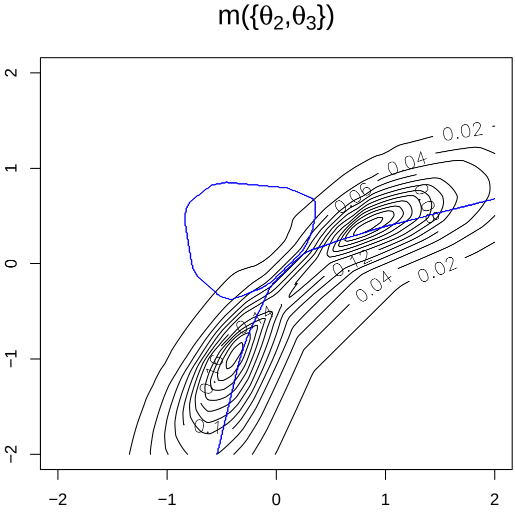
Mass on $\{\theta_1, \theta_2\}$



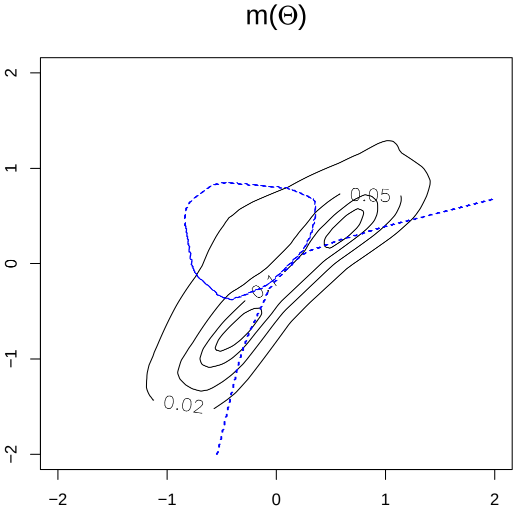
Mass on $\{\theta_1, \theta_3\}$



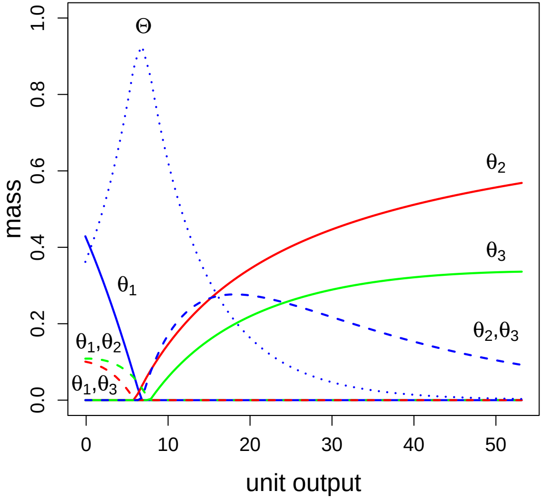
Mass on $\{\theta_2, \theta_3\}$



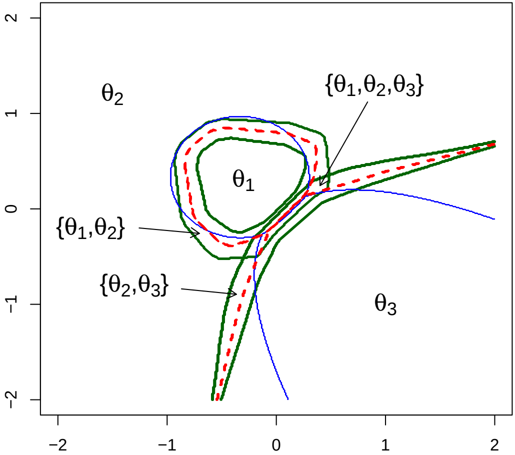
Mass on Θ



Hidden unit 2



Decision regions



Outline

1. Evidential distance-based classifiers
   - Evidential K-NN rule
   - Contextual Discounting Evidential K-NN
   - Evidential neural network classifier
2. Neural networks as evidential classifiers
   - Logistic regression and extensions
   - Binomial classifiers
   - Multinomial classifers
3. Clustering
   - Credal partition
   - EVCLUS

Hard and soft clustering concepts

Clustering = finding groups in data.

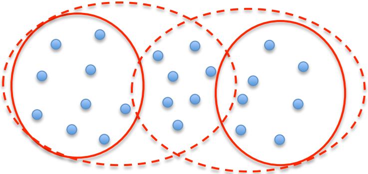
Hard clustering: no representation of uncertainty. Each object is assigned to one and only one group. Group membership is represented by binary variables u_{ik} such that $u_{ik} = 1$ if object i belongs to group k and $u_{ik} = 0$ otherwise.

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.

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

Hard and soft clustering concepts

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



Clustering and belief functions

clustering structure	uncertainty framework
fuzzy partition	probability theory
possibilistic partition	possibility theory
rough partition	(rough) sets
?	belief functions

- 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

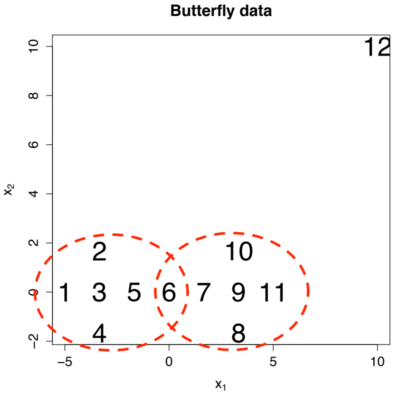
Evidential clustering

- Let $O = \{o_1, \ldots, o_n\}$ be a set of n objects and $\Omega = \{\omega_1, \ldots, \omega_c\}$ be a set of c groups (clusters).
- Each object o_i belongs to at most one group.
- Evidence about the group membership of object o_i is represented by a mass function m_i on Ω:
  - for any nonempty set of clusters $A \subseteq \Omega$, $m_i(A)$ is the probability of knowing only that o_i belong to one of the clusters in A.
  - $m_i(\emptyset)$ is the probability of knowing that o_i does not belong to any of the c groups.

Definition

The n-tuple $M = (m_1, \ldots, m_n)$ is called a credal partition.

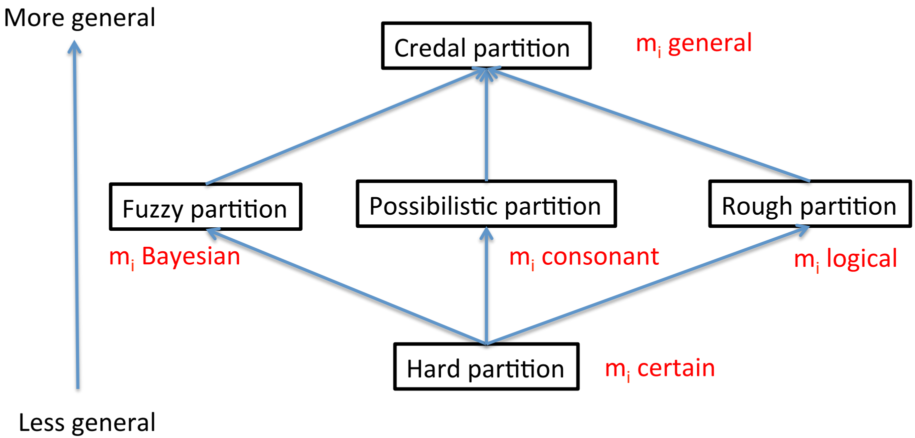
Example



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

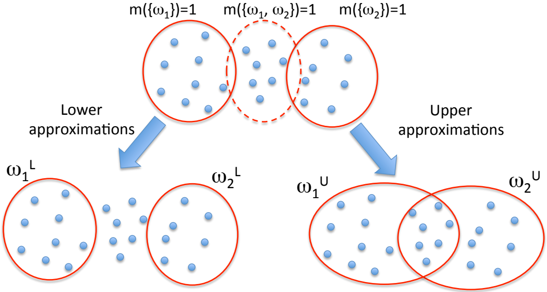


Rough clustering as a special case

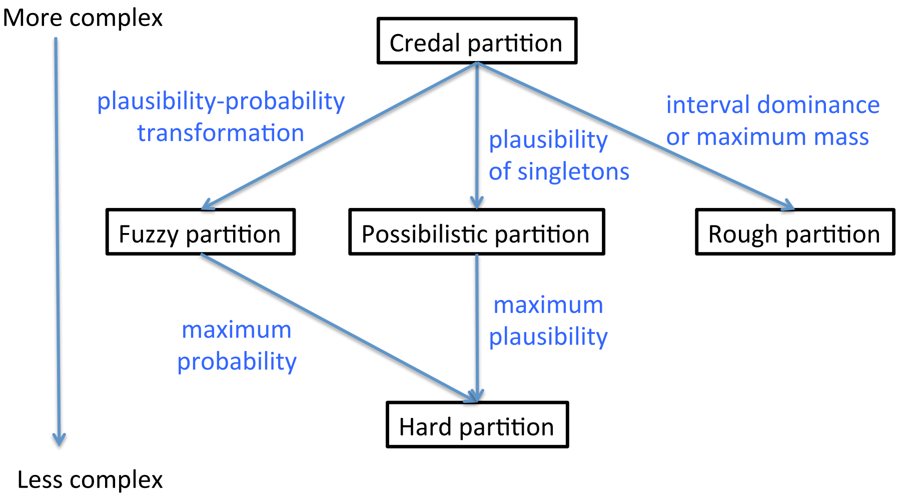
- Assume that each m_i is logical, i.e., $m_i(A_i) = 1$ for some $A_i \subseteq \Omega$, $A_i \neq \emptyset$.
- We can then define the lower and upper approximations of cluster ω_k as

$$\underline{\omega}_k = \{o_i \in O \mid A_i = \{\omega_k\}\}, \quad \overline{\omega}_k = \{o_i \in O \mid \omega_k \in A_i\}.$$

- The membership values to the lower and upper approximations of cluster ω_k are $\underline{u}_{ik} = Bel_i(\{\omega_k\})$ and $\overline{u}_{ik} = Pl_i(\{\omega_k\})$.



Summarization of a credal partition



Evidential clustering 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

Learning a Credal Partition from proximity data

- 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.
- It can be shown that the plausibility that objects o_i and o_j belong to the same group is

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

  where κ_{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 κ_{ij} correspond to larger dissimilarities d_{ij}.

Cost function

- Approach: minimize the discrepancy between the dissimilarities d_{ij} and the degrees of conflict κ_{ij}.
- Example of a cost (stress) function:

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

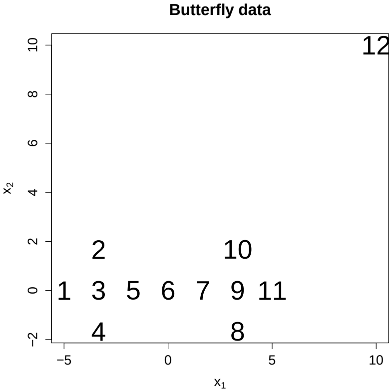
where φ is an increasing function from $[0, +\infty)$ to $[0, 1]$, for instance

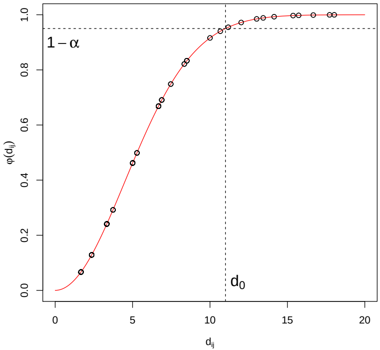
$$\varphi(d) = 1 - \exp(-\gamma d^2).$$

Butterfly example

Data and dissimilarities

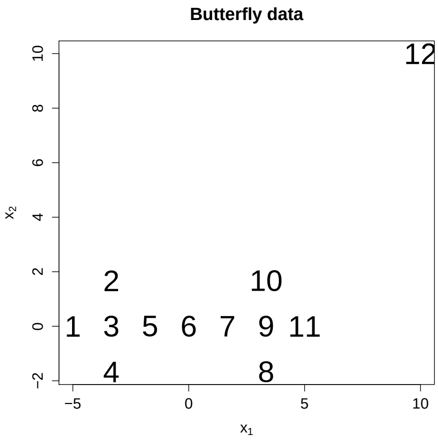
Determination of γ in $\varphi(d) = 1 - \exp(-\gamma d^2)$: fix $\alpha \in (0,1)$ and d_0 such that, for any two objects (o_i, o_j) with $d_{ij} \geq d_0$, the plausibility that they belong to the same cluster is at least $1 - \alpha$.

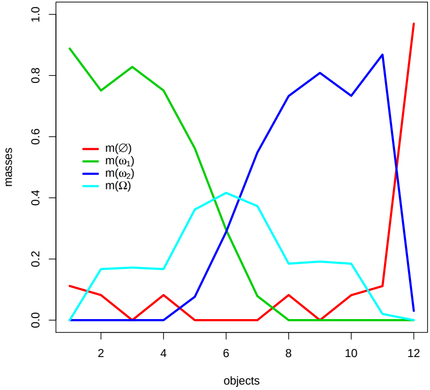




Butterfly example

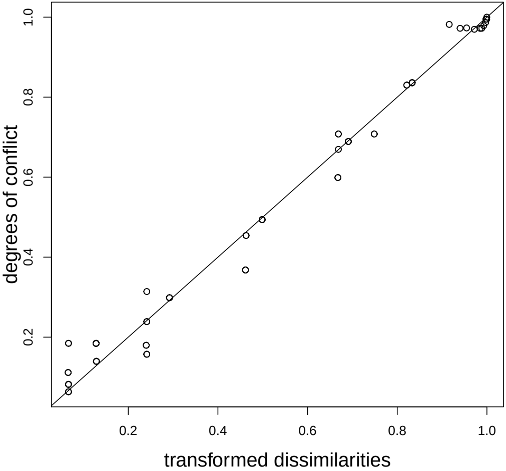
Credal partition



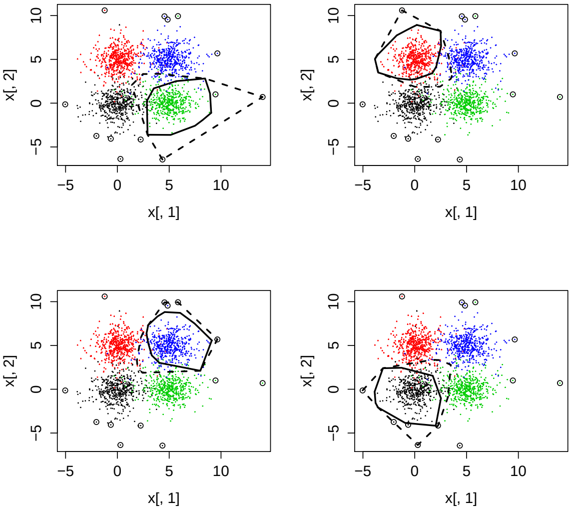


Butterfly example

Shepard diagram



Example with a four-class dataset (2000 objects)



Modifications of EVCLUS for large datasets

- Initially, EVCLUS used a gradient descent algorithm to minimize the stress function, and it required to store the whole dissimilarity matrix: it was limited to small sets of proximity data (a few hundreds of objects).
- Recent improvements to EVCLUS (Denœux et al., 2016) make it applicable to large datasets ($\sim 10^4 - 10^5$ objects and hundreds of classes).

Summary

- The theory of belief function has great potential for solving challenging machine learning problems:
  - Classification (supervised learning)
  - Clustering (unsupervised learning)
- Belief functions allow us to:
  - Learn from weak information (partially supervised learning, imprecise and uncertain data)
  - Quantify uncertainty on the outputs of a learning system (e.g., prediction uncertainty,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 (multilabel classification, clustering, preference learning, etc.)
- R packages `evclass` and `evclust` available from CRAN at

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

References I

cf. http://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.

C. Lian, S. Ruan and T. Denoeux.
Dissimilarity metric learning in the belief function framework.
IEEE Transactions on Fuzzy Systems, 24(6):1555–1564, 2016.

T. Denœux.
Maximum likelihood estimation from Uncertain Data in the Belief Function Framework.
IEEE Transactions on Knowledge and Data Engineering, Vol. 25, Issue 1, pages 119-130, 2013.

References II

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

T. Denœux, O. Kanjanatarakul and S. Sriboonchitta.
A New Evidential K-Nearest Neighbor Rule based on Contextual Discounting with Partially Supervised learning.
International Journal of Approximate Reasoning, 113:287–302, 2019.

T. Denœux.
Logistic Regression, Neural Networks and Dempster-Shafer Theory: a New Perspective.
Knowledge-Based Systems, 176:54–67, 2019.

T. Denœux, S. Sriboonchitta and O. Kanjanatarakul
Evidential clustering of large dissimilarity data.
Knowledge-Based Systems, 106:179–195, 2016.

T. Denœux, S. Li and S. Sriboonchitta.
Evaluating and Comparing Soft Partitions: an Approach Based on Dempster-Shafer Theory.
IEEE Transactions on Fuzzy Systems, 26(3):1231–1244, 2018.