Belief functions and Machine Learning: A brief introduction

Thierry Denœux

Université de technologie de Compiègne, Compiègne, France
Institut Universitaire de France, Paris, France

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

SUM 2019, Compiègne, France
December 16, 2019

Theory of belief functions

- Also referred to as Dempster-Shafer (DS) theory, evidence theory, Transferable Belief Model.
- Originates from Dempster's seminal work on statistical inference in the late 1960's. Formalized by Shafer in his seminal 1976 book. Further developed and popularized by Smets in the 1990's and early 2000's.
- DS theory has a level of generality that makes it applicable to a wide range problems involving uncertainty. It has been applied in may areas, including statistical inference, knowledge representation, information fusion, etc.

How can the theory of belief functions contribute to Machine Learning?

Key features of DS theory

I will not attempt to give an exhaustive account of the already large number of applications of DS theory to ML. Instead, I will give a few examples to illustrate the following key features of DS theory:

Generality: DS theory is based on the idea of combining sets and probabilities. It extends both

- Propositional logic, computing with sets (interval analysis)
- Probabilistic reasoning

Everything than can be done with sets or with probabilities alone can be done with belief functions, but DS theory can do much more!

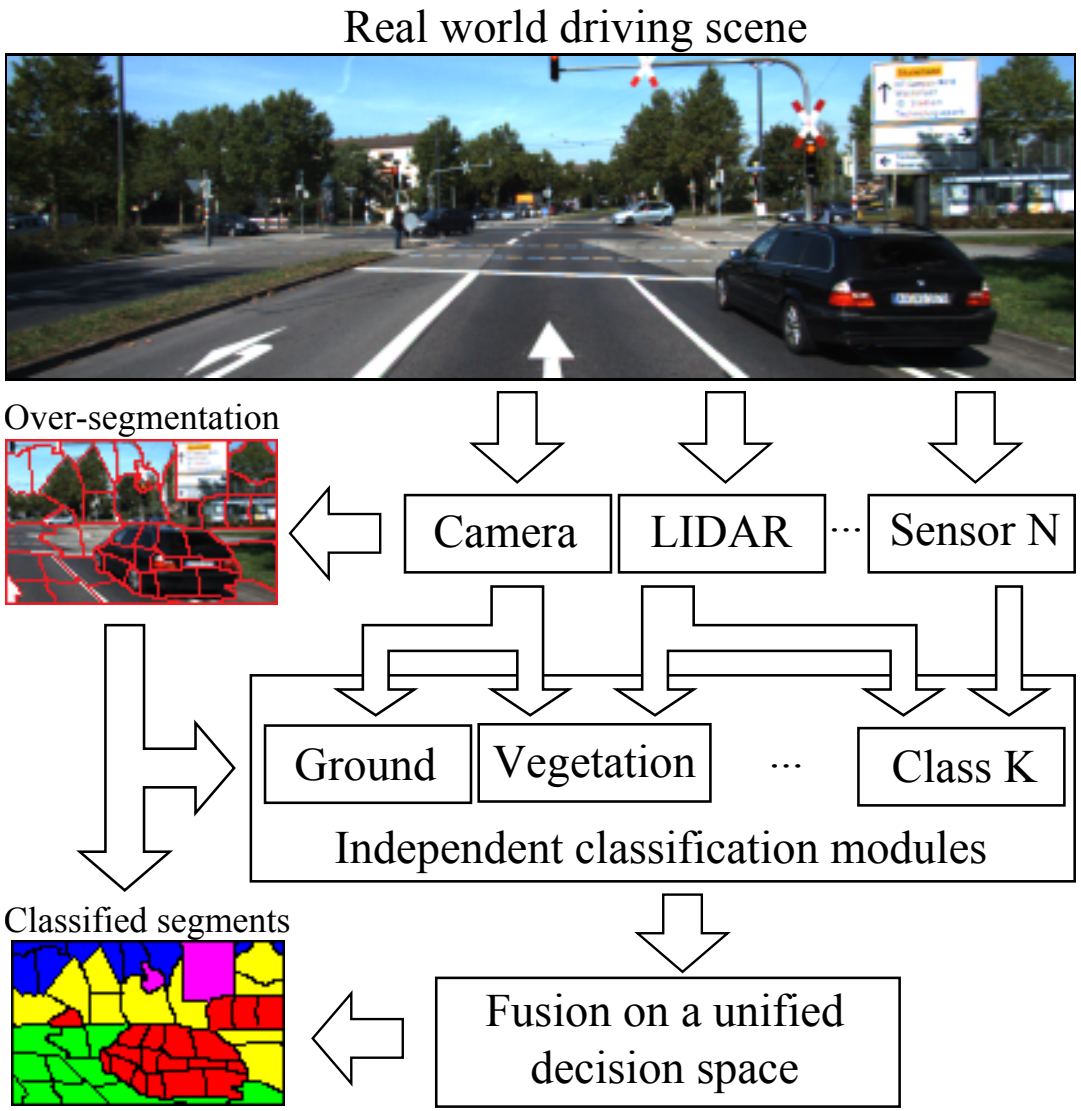
Operationality: DS theory is easily put in practice by breaking down the available evidence into elementary pieces of evidence, and combining them by a suitable operator called Dempster's rule of combination.

Scalability: Contrary to a widespread misconception, evidential reasoning can be applied to very large problems.

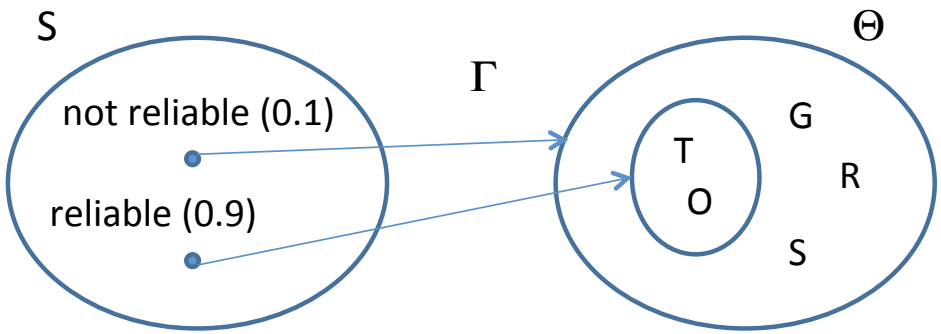
Outline

Outline

Road scene analysis



Uncertain sensor information



- Let $\Theta = \{G, R, T, O, S\}$, corresponding to the possibilities **G**rass, **R**oad, **T**ree/Bush, **O**bstacle, **S**ky. Let Y be the true answer.
- A Lidar sensor tells us that $Y \in \{T, O\}$ and nothing more. There is probability 0.1 that the sensor is not reliable.
- With probability 0.9, the sensor evidence tells us that $Y \in \{T, O\}$, and nothing more. With probability 0.1, it tells us nothing.
- This can be formalized by the mapping $m : 2^\Theta \to [0, 1]$, called a mass function:

$$m(\{T, O\}) = 0.9, \quad m(\Theta) = 0.1, \quad m(B) = 0 \text{ for all other } B \subset \Theta$$

General definition

Definition (Mass function)

A mass function on a finite set Θ is a mapping $m : 2^\Theta \to [0, 1]$ such that

$$\sum_{A \subseteq \Theta} m(A) = 1.$$

If $m(\emptyset) = 0$, m is normalized (usually assumed).
Every subset A of Θ such that $m(A) > 0$ is a focal set.

- Interpretation: if Θ is the domain of a variable Y, $m(A)$ is the probability that the meaning of the evidence is exactly "$Y \in A$"; it is thus the measure of the belief one is willing to commit exactly to A.
- Total ignorance is represented by the vacuous mass function $m_?$ verifying $m_?(\Theta) = 1$.
- Special cases:
  - Logical: only one focal set
  - Bayesian: all focal sets are singletons
  - Consonant: the focal sets are nested

Fundamental assumption

Assumption (Representability of evidence by a mass function)

Any piece of evidence about a variable $Y \in \Theta$ induces the same state of knowledge as a randomly coded message that may have different meanings of the form "$Y \in A_i$" for $i = 1, \ldots, n$, with probabilities $p_1, \ldots, p_n$ such that $\sum_{i=1}^{n} p_i = 1$.
It can thus be represented by a mass function m with focal sets $A_1, \ldots, A_n$ and masses $m(A_i) = p_i$, $i = 1, \ldots, n$.

Belief and plausibility functions

Definition

Given a normalized mass function m on Θ, the belief and plausibility functions are defined, respectively, as

$$Bel(A) := \sum_{B \subseteq A} m(B)$$

$$Pl(A) := \sum_{B \cap A \neq \emptyset} m(B) = 1 - Bel(\overline{A}),$$

for all $A \subseteq \Theta$.

- Interpretation:
  - $Bel(A)$ is the probability that "$Y \in A$" can be deduced from the evidence; it is a measure of total support in A
  - $Pl(A)$ is a measure of the lack of support in $\overline{A}$ (or consistency with A)
- Total ignorance: $Bel_?(A) = 0$ for all $A \neq \Theta$ and $Pl_?(A) = 1$ for all $A \neq \emptyset$.

The contour function

Definition

Given a mass function m on Θ, the contour function is the mapping pl from Θ to $[0,1]$ defined by

$$pl(\theta) = Pl(\{\theta\}), \quad \forall \theta \in \Theta.$$

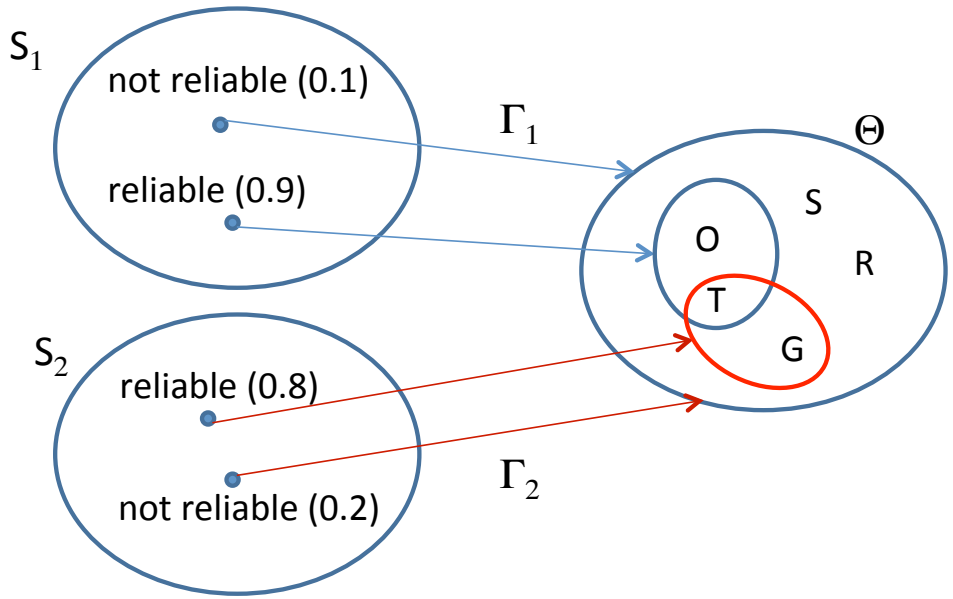
The contour function plays an important role in many applications of belief functions:

- It is a simple summary of belief function, useful for decision making
- It can be computed very efficiently when combining several mass functions (see below).

Outline

Combining Mass Functions

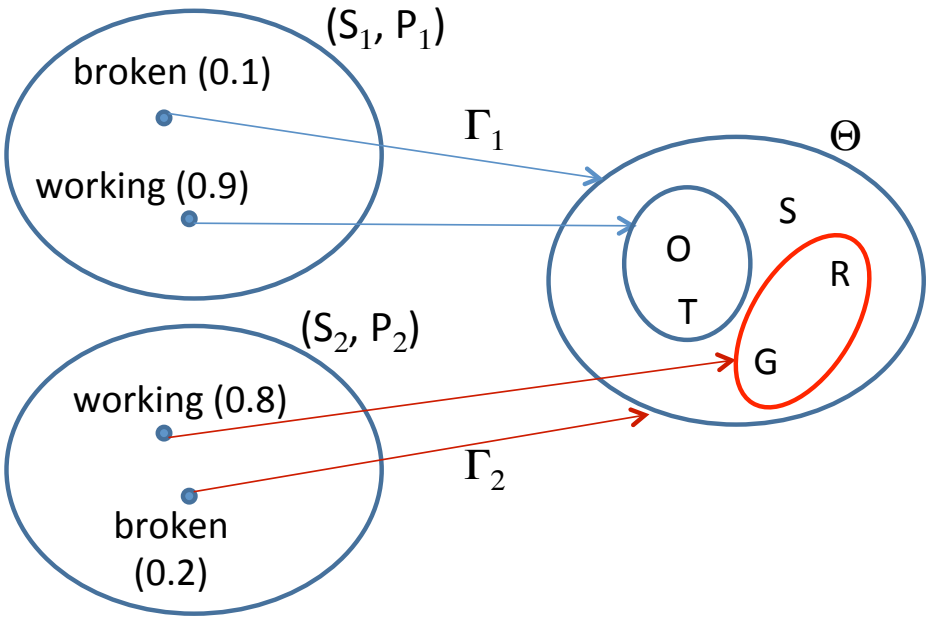
Two independent sensors:



What do we know?

		S_2 reliable [0.8]	S_2 not reliable [0.2]
S_1	reliable [0.9]	$\{T\}$ [0.72]	$\{T, O\}$ [0.18]
	not reliable [0.1]	$\{T, G\}$ [0.08]	Θ [0.02]

Case of conflicting pieces of evidence



		S_2	
		reliable [0.8]	not reliable [0.2]
S_1	reliable [0.9]	$\emptyset$ [~~0.72~~ $\rightarrow$ 0]	$\{T, O\}$ [0.18/0.28]
	not reliable [0.1]	$\{R, G\}$ [0.08/0.28]	Θ [0.02/0.28]

Dempster's rule

Definition (Dempster's rule)

Let m_1 and m_2 be two mass functions. Their orthogonal sum is the mass function defined by

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

and $(m_1 \oplus m_2)(\emptyset) = 0$, where κ is the degree of conflict defined as

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

Remark: $m_1 \oplus m_2$ exists iff $\kappa < 1$.

Properties

Proposition

1. *If several pieces of evidence are combined, the order does not matter:*

$$m_1 \oplus m_2 = m_2 \oplus m_1$$

$$m_1 \oplus (m_2 \oplus m_3) = (m_1 \oplus m_2) \oplus m_3$$

2. *A mass function m is not changed if combined with $m_?$:*

$$m \oplus m_? = m.$$

3. *Let m_1 and m_2 be two mass functions with contour functions pl_1 and pl_2. The contour function of $m_1 \oplus m_2$ is*

$$pl_1 \oplus pl_2 = \frac{1}{1-\kappa}\, pl_1 \cdot pl_2 \propto pl_1 pl_2.$$

Weights of evidence

Dempster's rule can often be easily computed by adding weights of evidence.

Definition (Weight of evidence)

Given a simple mass function of the form

$$m(A) = s$$
$$m(\Theta) = 1 - s,$$

the quantity $w = -\log(1-s)$ *is called the weight of evidence for A.*

Mass function m is denoted by A^w.

Proposition

The orthogonal sum of two simple mass functions A^{w_1} *and* A^{w_2} *is*

$$A^{w_1} \oplus A^{w_2} = A^{w_1+w_2}$$

Outline

Two approaches to clustering

Two main approaches for applying belief functions to clustering:

1. Assume that there exists one true partition and apply evidential reasoning to the frame of all partitions of a dataset. The goal is to find the most plausible partition. This is the approach followed by the EK-NNclus algorithm[1].
2. Generalize the notion of partition to account for the uncertainty in the assignment of objects to clusters. This idea leads to the notion of credal partition[2], a generalization of hard, fuzzy and rough partitions.

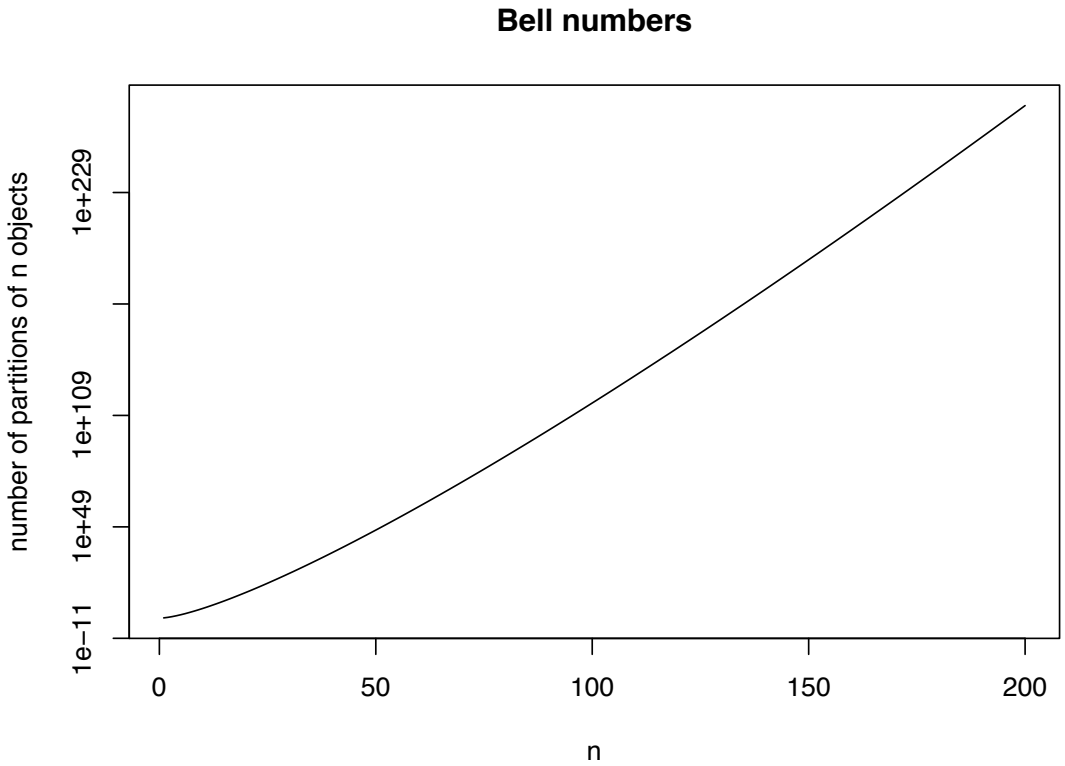
[1] T. Denoeux *et al.* EK-NNclus: a clustering procedure based on the evidential K-nearest neighbor rule. *KBS*, 88:57–69, 2015.

[2] T. Denœux and M.-H. Masson. EVCLUS: Evidential Clustering of Proximity Data. *IEEE TSMC B*, 34(1):95–109, 2004.

Outline

Reasoning in the space of all partitions

- Let $\mathcal{O}$ be a set of n object. Assuming there is a true unknown partition, the frame of discernment is the set $\mathcal{R}$ of all partitions ($\equiv$ equivalence relations) of $\mathcal{O}$. 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.
- Given two objects o_i and o_j, let $\Theta_{ij} = \{s_{ij}, \neg s_{ij}\}$, where s_{ij} is the hypothesis that o_i and o_j belong to the same class.
- Assumptions
  1. Two objects are all the more likely to belong to the same class, that they are more similar; given any two object i and j with dissimilarity d_{ij}, we thus have the simple mass function
  $$m_{ij} := \{s_{ij}\}^{w_{ij}}$$
  where the weight of evidence $w_{ij} = \varphi(d_{ij}) \in (0, +\infty)$ is a decreasing function of d_{ij}
  2. The mass functions m_{ij} for all $i > j$ represent independent items of evidence.
- How to combine these $n(n-1)/2$ mass functions to find the most plausible partition of the n objects?

Evidence combination

- Let $\mathcal{R}_{ij}$ 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 $\mathcal{R}$ of equivalence relations:

$$\begin{array}{ccc} m_{ij}(\{s_{ij}\}) & \longrightarrow & \mathcal{R}_{ij} \\ m_{ij}(\Theta_{ij}) & \longrightarrow & \mathcal{R} \end{array}$$

- Combining the extended mass functions by Dempster's rule, we get

$$m = \bigoplus_{i<j} \mathcal{R}_{ij}^{w_{ij}},$$

with corresponding contour function:

$$pl(R) \propto \prod_{i<j} pl_{ij}(R) = \prod_{i<j} (e^{-w_{ij}})^{1-r_{ij}}$$

for any $R = (r_{ij}) \in \mathcal{R}$.

Decision

- The logarithm of the contour function can be written as

$$\log pl(R) = \sum_{i<j} r_{ij} w_{ij} + C$$

- Finding the most plausible partition is thus a binary linear programming problem. It can be solved exactly only for small n.
- For large n, the problem can be solved approximately using a heuristic greedy search procedure: the EK-NNclus algorithm[3].

[3] Available in the R package `evclust`.

EK-NNclus algorithm

1. Initialization: $c = n$, $u_{ik} := I(i = k)$ (each cluster contains exactly one object)
2. For each object i: compute the set $N_K(i)$ of indices of its K nearest neighbors
3. For each object pair (i, j): set $w_{ij} := \varphi(d_{ij}) I(j \in N_K(i))$
4. For each object i (picked in random order): compute

$$t_{ik} := \sum_{j \in N_K(i)} w_{ij} u_{jk}, \quad k = 1, \ldots, c.$$

   Set $u_{ik} := 1$ if $t_{ik} = \max_{k'} t_{ik'}$ and $u_{ik} := 0$ otherwise
5. Update c and variables u_{ik} accordingly
6. Return to Step 4 while at least one label has changed

Convergence of EK-NNclus

The EK-NNclus algorithm can be implemented in a Hopfield neural network, with n groups of c neurons (one for each cluster). The state of neuron k of group i is u_{ik}. At each iteration, the states in each group are updated. The network minimizes the energy function

$$E(R) := -\frac{1}{2}\sum_{k=1}^{c}\sum_{i=1}^{n}\sum_{j\neq i} w_{ij}u_{ik}u_{jk} = -\sum_{i<j} r_{ij}w_{ij} = -\log pl(R) + C.$$

Consequence:

Theorem

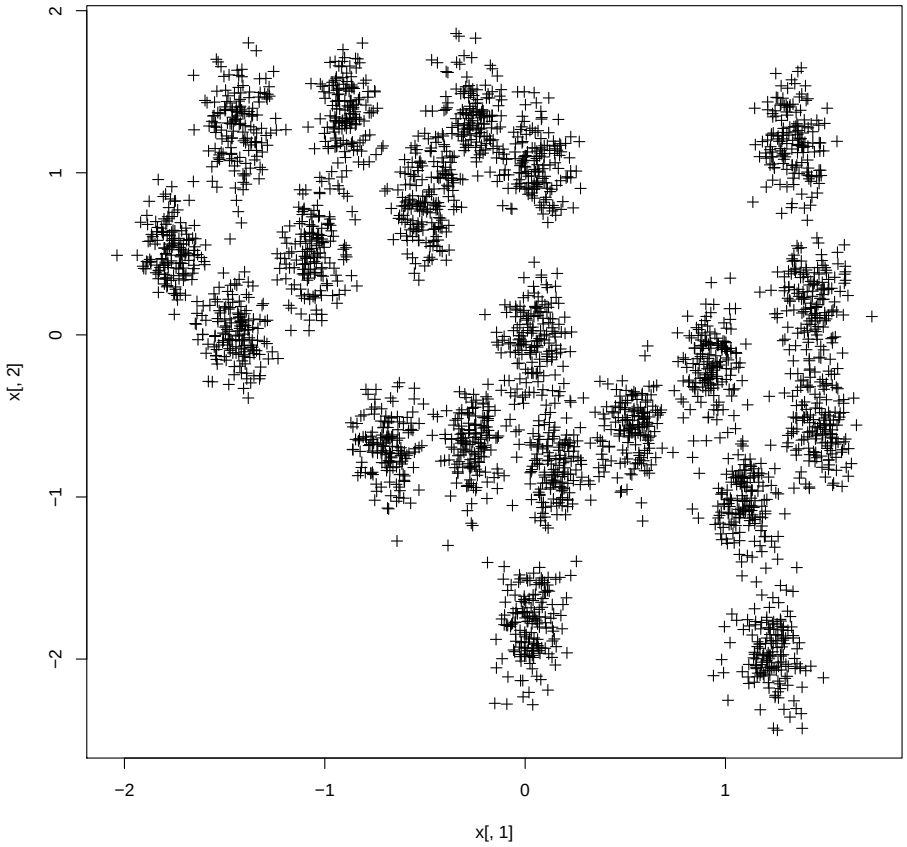
If $K = n - 1$ the EK-NNclus algorithm converges in a finite number of iterations to a partition R corresponding to a local maximum of $pl(R)$.

Conjecture

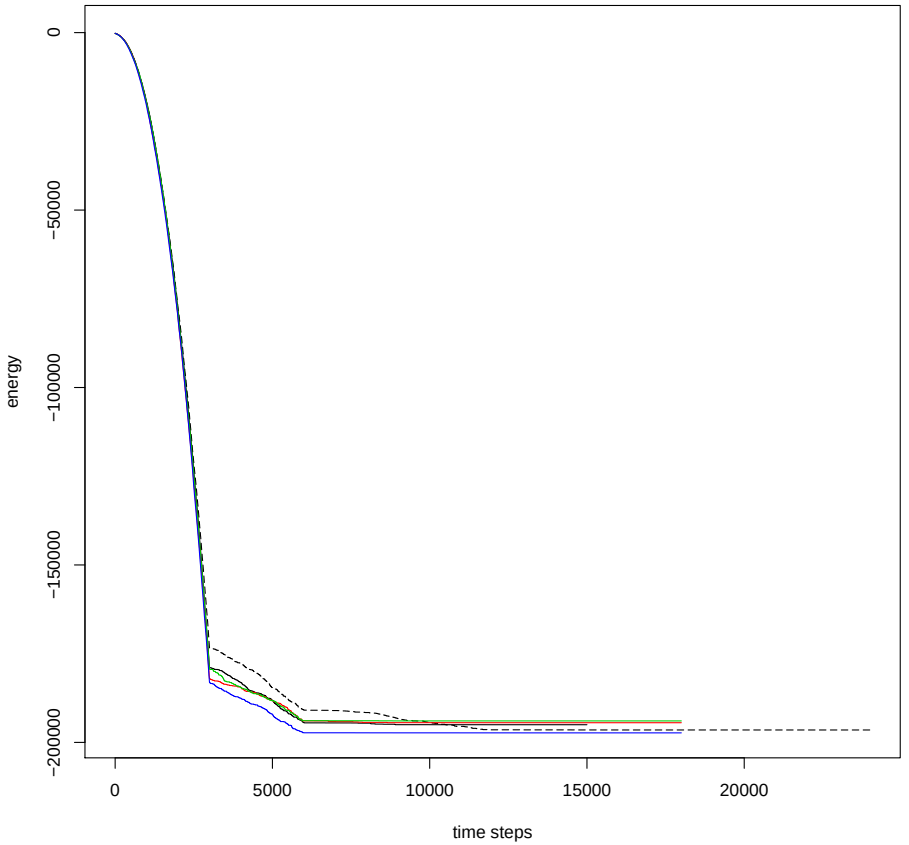
The above results holds also if $K < n - 1$.

Example

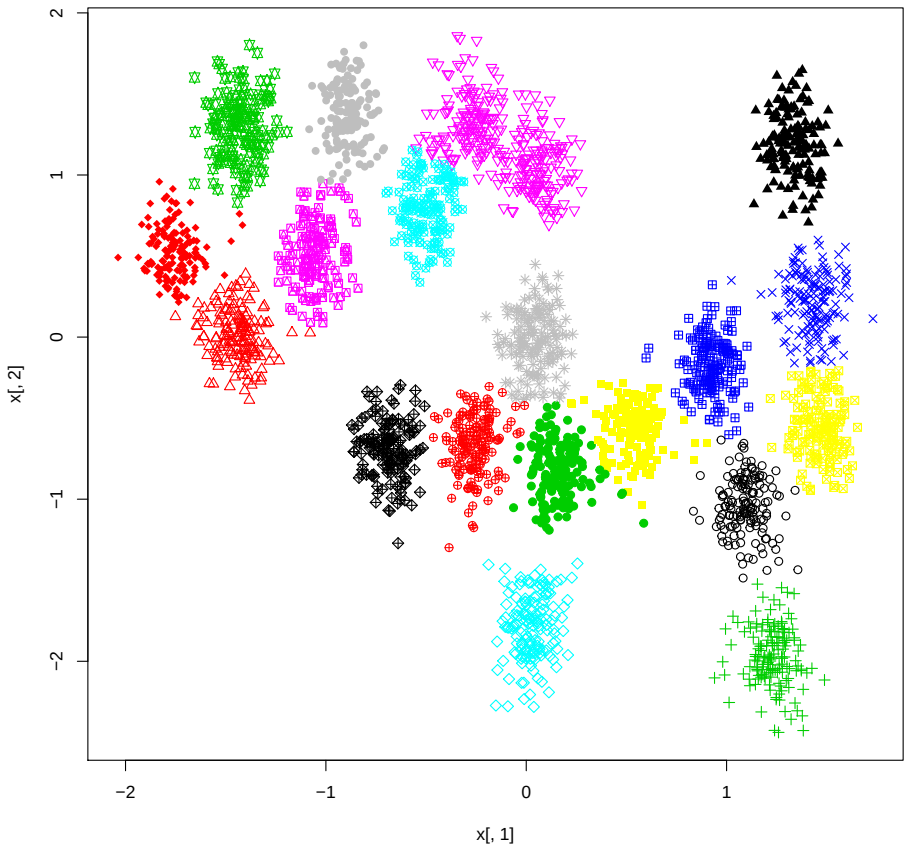
Dataset



Energy function (5 runs)



Final partition



Outline

Clustering and belief functions

- Several "soft" generalizations of the notion of hard partition have been proposed over the years:
  - Fuzzy partition: $u_{ik} \in [0,1]$, $\sum_{k=1}^{c} u_{ik} = 1$
  - Possibilistic partition: $u_{ik} \in [0,1]$
  - Rough partition: $(\underline{u}_{ik}, \overline{u}_{ik}) \in \{0,1\}^2$, with $\underline{u}_{ik} \leq \overline{u}_{ik}$, $\sum_{k=1}^{c} \underline{u}_{ik} \leq 1$ and $\sum_{k=1}^{c} \overline{u}_{ik} \geq 1$
- These notions can be further generalized in the DS framework, with the following main objectives:
  - Unify the various approaches to clustering, and derive new tools to compare and/or combine different soft partitions [4]
  - Achieve a richer and more accurate representation of the uncertainty of a clustering structure.

[4] T. Denœux *et al.*. Evaluating and Comparing Soft Partitions: an Approach Based on Dempster-Shafer Theory. *IEEE TFS*, 26(3):1231–1244, 2018.

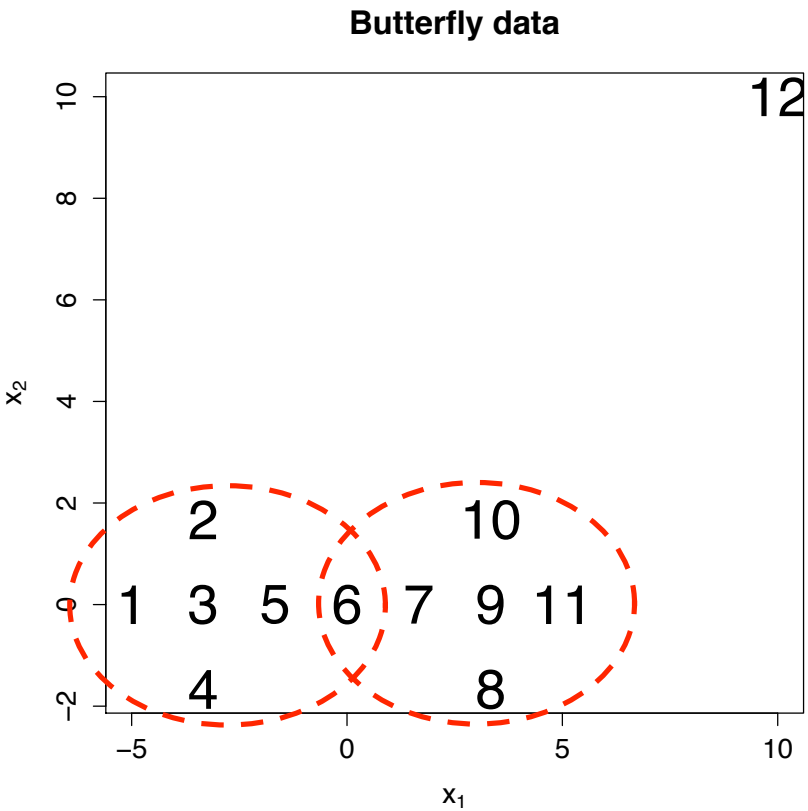
Credal partition

- 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).
- Assumption: each object o_i belongs to at most one group.

Definition

A credal partition is an n-tuple $M := (m_1, \ldots, m_n)$, where each m_i is a (not necessarily normalized) mass function on Ω representing evidence about the cluster membership of object o_i .

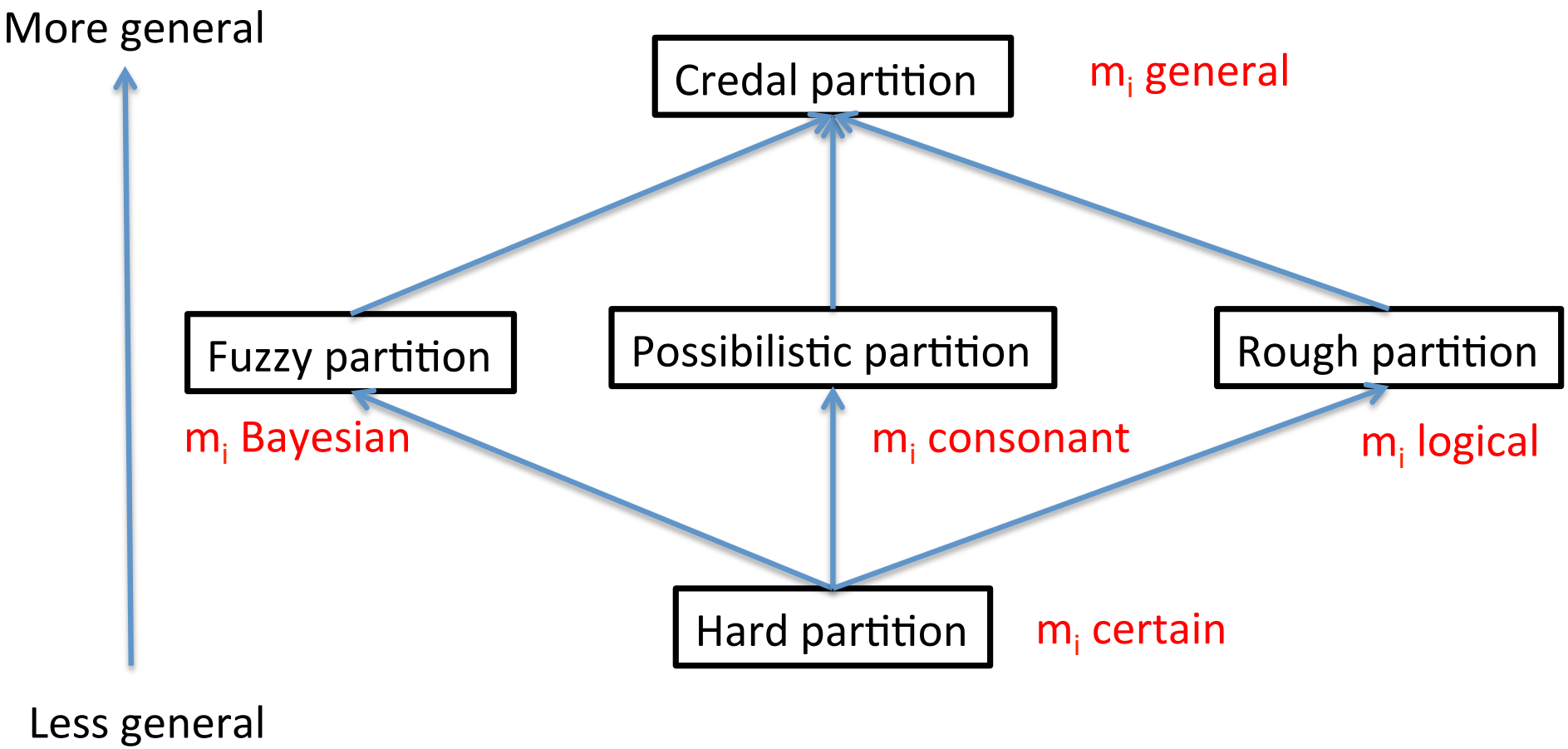
Example



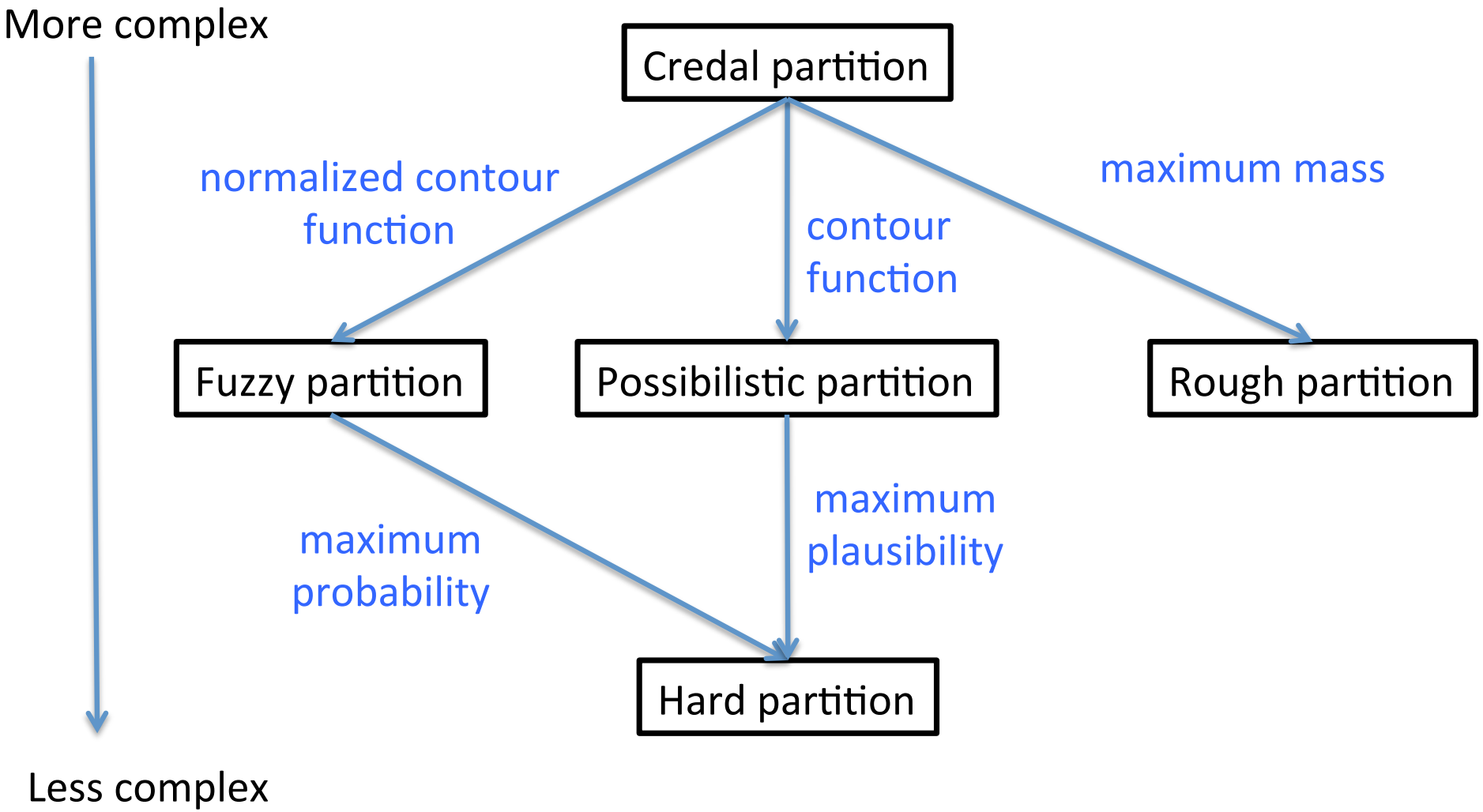
Credal partition

	$\emptyset$	$\{\omega_1\}$	$\{\omega_2\}$	$\{\omega_1, \omega_2\}$
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



Summarization of a credal partition



Evidential clustering algorithms

1. Evidential *c*-means (ECM)[5]:
   - Attribute data
   - HCM, FCM family
2. EVCLUS[6]:
   - Attribute or proximity (possibly non metric) data
   - Multidimensional scaling approach
3. Bootstrapping approach[7]
   - Based on a mixture models and bootstrap confidence intervals
   - The resulting credal partition has frequentist properties

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

[6] T. Denœux *et al.* Evidential clustering of large dissimilarity data. *KBS*, 106:179–195, 2016.

[7] T. Denœux. Calibrated model-based evidential clustering using bootstrapping. Preprint arXiv:1912.06137, 2019.

Outline

Basic idea

- Objective: account for clustering uncertainty.
- Model-based clustering allows us to estimate probabilities of cluster membership. The result is a fuzzy partition that describes first-order uncertainty.
- To represent second-order uncertainty (uncertainty about the probability estimates), we need a more general model. Here, we exploit the expressiveness of DS theory and use a credal partition.
- This credal partition will be based on bootstrapping mixture models.
- As it will be built to approximate some confidence intervals, the resulting credal partition will be frequency-calibrated[8].

[8] T. Denœux and S. Li. Frequency-Calibrated Belief Functions: Review and New Insights. *IJAR*, 92:232–254, 2018.

Model

- We assume that the attributes vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are an iid random sample from a mixture distribution with pdf

$$p(\boldsymbol{x}; \boldsymbol{\theta}) := \sum_{k=1}^{c} \pi_k p_k(\boldsymbol{x}; \boldsymbol{\theta}_k)$$

  where each component in the mixture corresponds to a cluster and $\boldsymbol{\theta}$ is the parameter vector.
- The probability that object i belongs to cluster k is

$$\pi_k(\boldsymbol{x}_i; \boldsymbol{\theta}) = \frac{p_k(\boldsymbol{x}_i; \boldsymbol{\theta}_k)\pi_k}{\sum_{\ell=1}^{c} p_\ell(\boldsymbol{x}_i; \boldsymbol{\theta}_\ell)\pi_\ell}$$

- The probability that two objects i and j belong to the same cluster is

$$P_{ij}(\boldsymbol{\theta}) = \sum_{k=1}^{c} \pi_k(\boldsymbol{x}_i; \boldsymbol{\theta})\, \pi_k(\boldsymbol{x}_j; \boldsymbol{\theta})$$

Estimation

- Given a dataset $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, we can compute the MLE $\widehat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}$ and the corresponding MLEs $\pi_k(\boldsymbol{x}_i; \widehat{\boldsymbol{\theta}})$ and $P_{ij}(\widehat{\boldsymbol{\theta}})$.
- To describe the uncertainty of these estimates, we can use the bootstrap.
- Confidence intervals on the pairwise probabilities $P_{ij}(\boldsymbol{\theta})$ can easily be obtained by the bootstrap percentile method.

Bootstrap confidence intervals on the pairwise probabilities

Require: Dataset $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, model $p(\cdot; \theta)$, number of bootstrap samples B, confidence level $1 - \alpha$

for $b = 1$ **to** B **do**

Draw $\boldsymbol{x}_{b1}, \ldots, \boldsymbol{x}_{bn}$ from $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ with replacement

Compute the MLE $\widehat{\theta}_b$ from $\boldsymbol{x}_{b1}, \ldots, \boldsymbol{x}_{bn}$

for all $i < j$ **do**

Compute $P_{ij}(\widehat{\theta}_b)$

end for

end for

for all $i < j$ **do**

$$P_{ij}^l := \text{Quantile}\left(\left\{P_{ij}(\widehat{\theta}_b)\right\}_{b=1}^{B}; \tfrac{\alpha}{2}\right)$$

$$P_{ij}^u := \text{Quantile}\left(\left\{P_{ij}(\widehat{\theta}_b)\right\}_{b=1}^{B}; 1 - \tfrac{\alpha}{2}\right)$$

end for

Constructing a credal partition

- Given a credal partition $M = (m_1, \ldots, m_n)$, the belief and plausibility that any two objects i and j belong to the same cluster are given by

$$Bel_{ij} = \sum_{k=1}^{c} m_i(\{\omega_k\}) m_j(\{\omega_k\})$$

$$Pl_{ij} = \sum_{A \cap B \neq \emptyset} m_i(A) m_j(B)$$

- Idea: search for a credal partition M such that the belief-plausibility intervals $[Bel_{ij}, Pl_{ij}]$ approximate the confidence intervals $[P_{ij}^l, P_{ij}^u]$.

Optimization problem and frequentist property

- We consider the ptimization problem

$$\min_{M} \sum_{i<j} \left(Bel_{ij} - P_{ij}^{l}\right)^2 + \left(Pl_{ij} - P_{ij}^{u}\right)^2,$$

  which can be solved using a grouped coordinate descent procedure (solving a QP problem at each iteration).
- The solution verifies

$$P\left(Bel_{ij}(\{s_{ij}\}) \leq P_{ij}(\theta) \leq Pl_{ij}(\{s_{ij}\})\right) \approx 1 - \alpha.$$

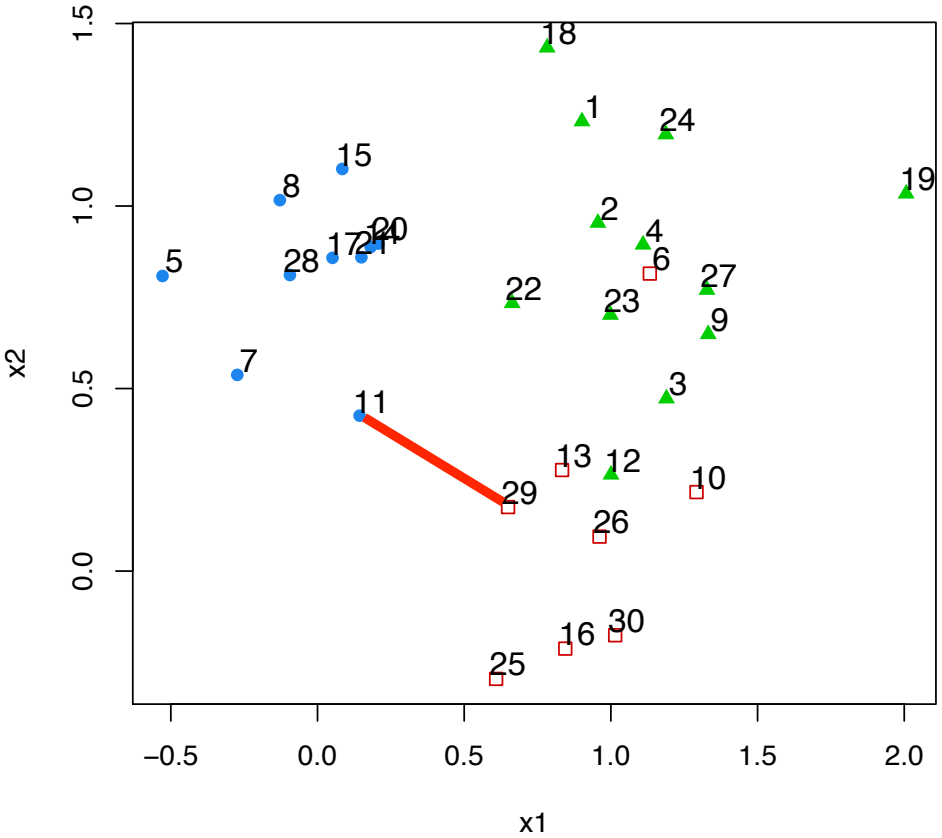
- This corresponds to the definition of a predictive belief function[9] at level $1 - \alpha$, a special kind of frequency-calibrated belief function[10].

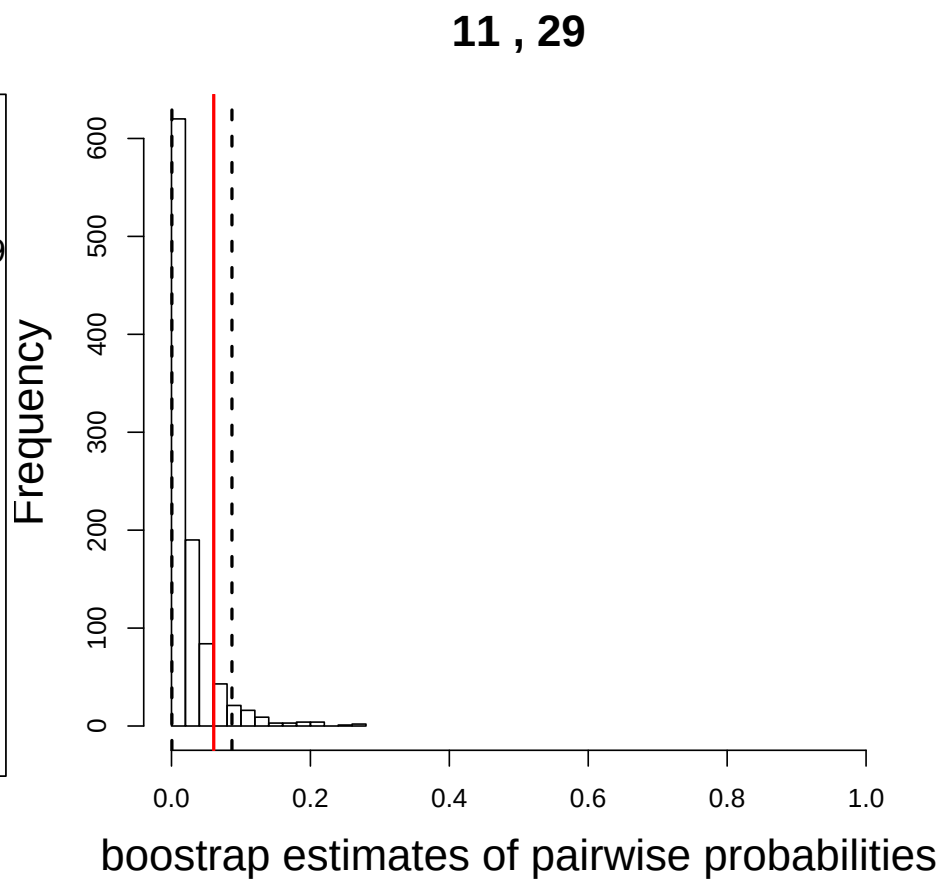
[9] T. Denœux. Constructing Belief Functions from Sample Data Using Multinomial Confidence Regions. *IJAR*, 42(3):228–252, 2006.

[10] T. Denœux and S. Li. Frequency-Calibrated Belief Functions: Review and New Insights. *IJAR*, 92:232–254, 2018.

Example

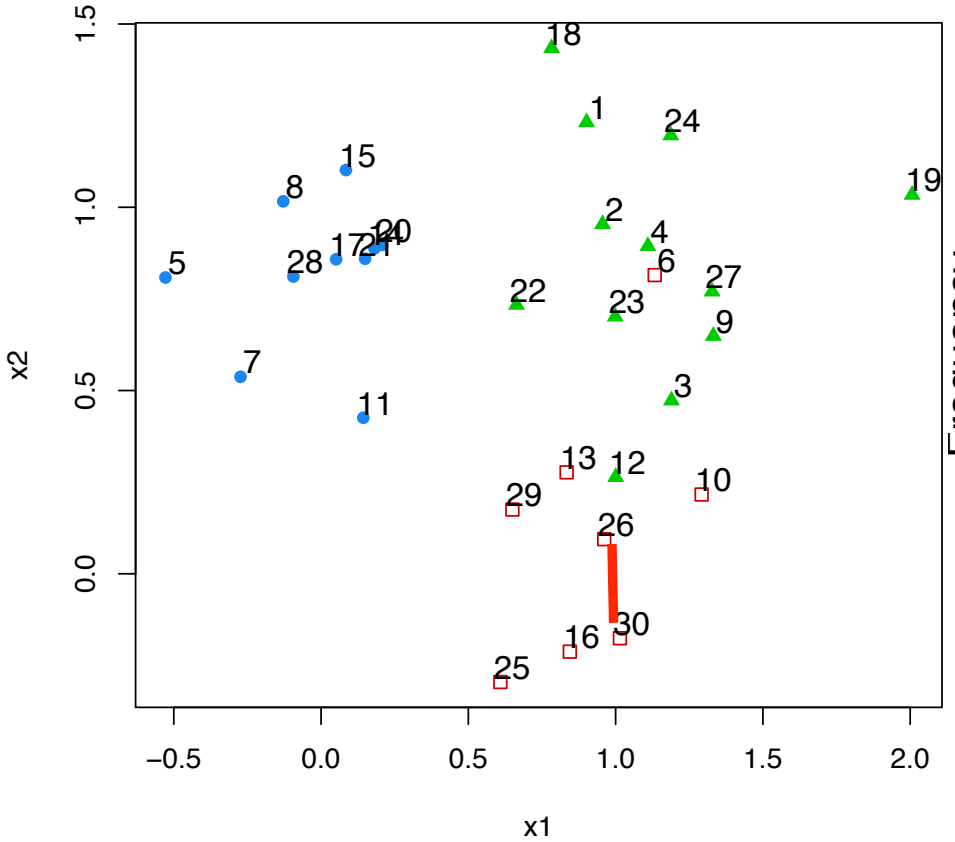
Bootstrap confidence intervals

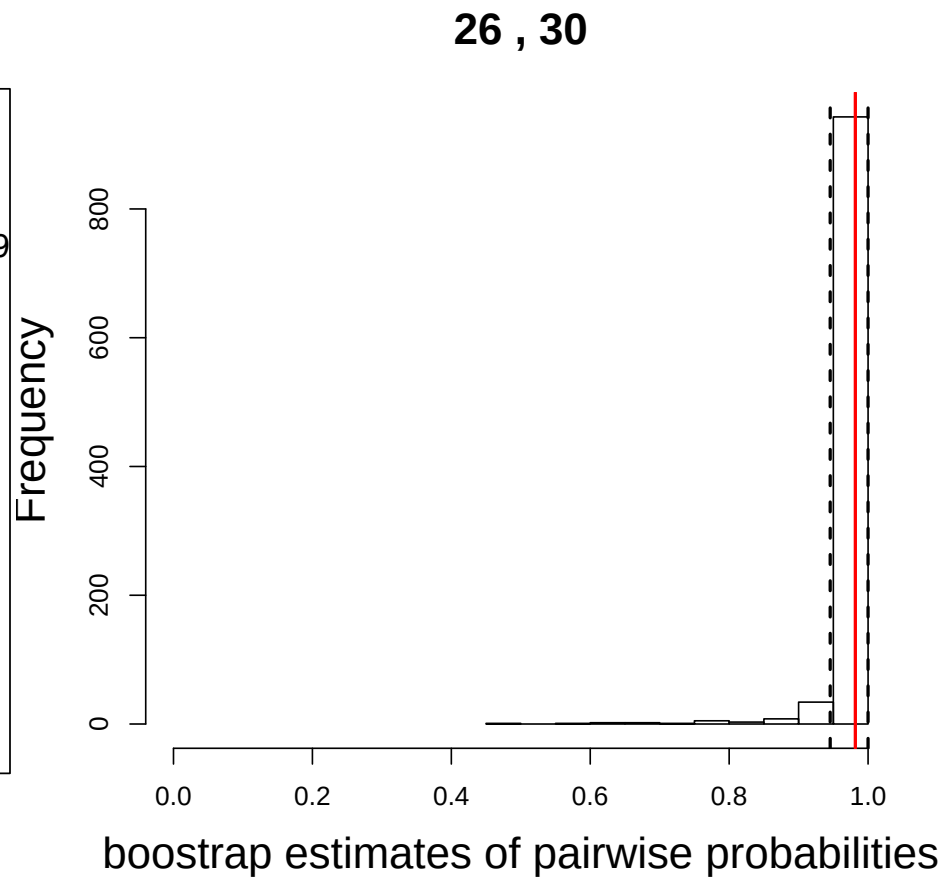




Example

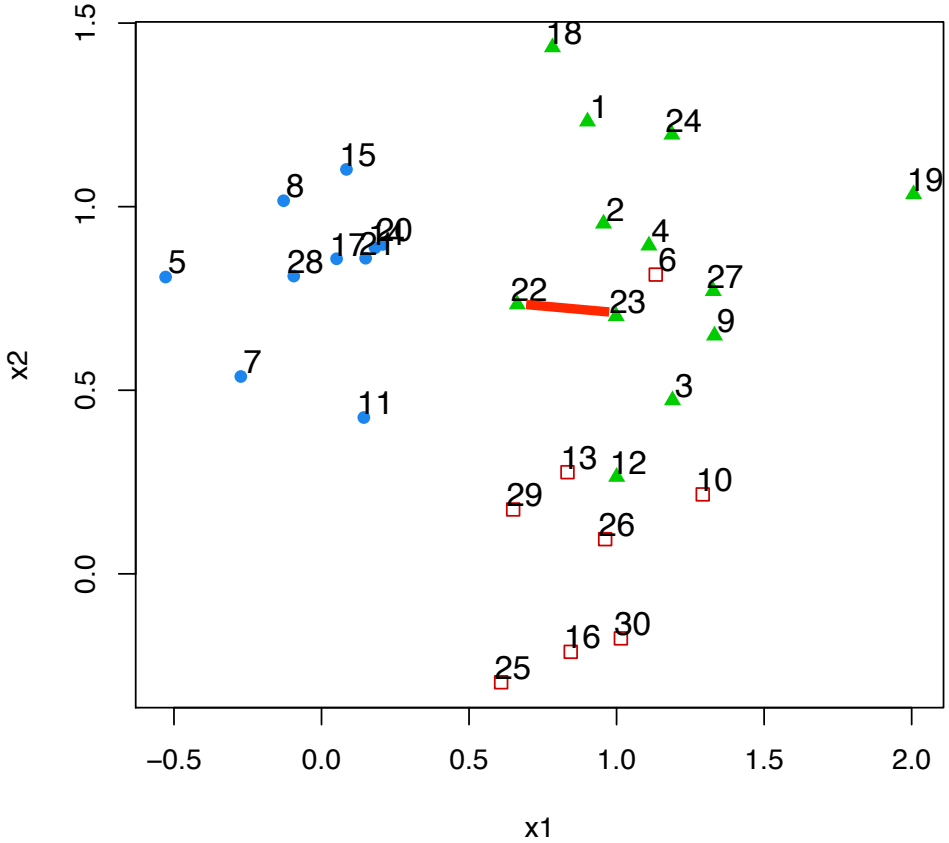
Bootstrap confidence intervals

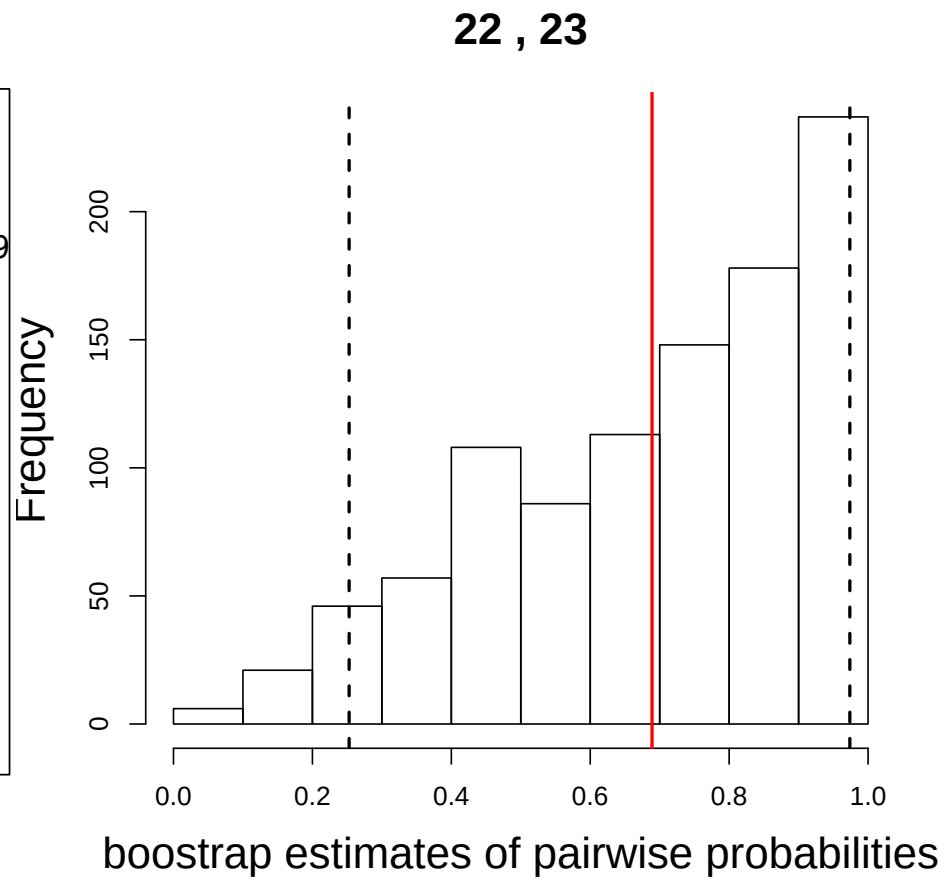




Example

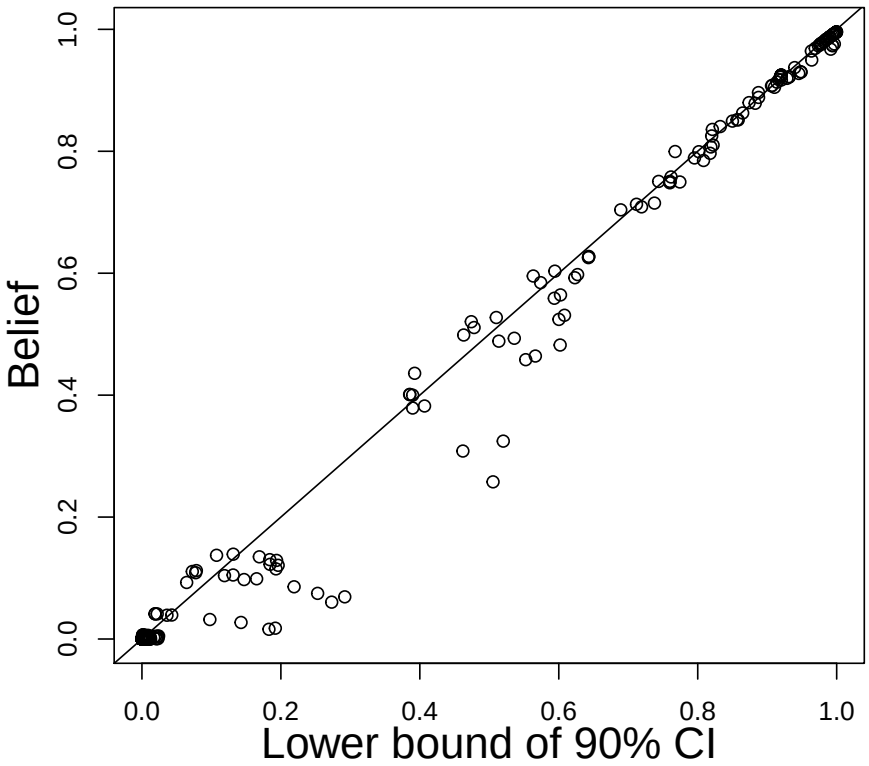
Bootstrap confidence intervals

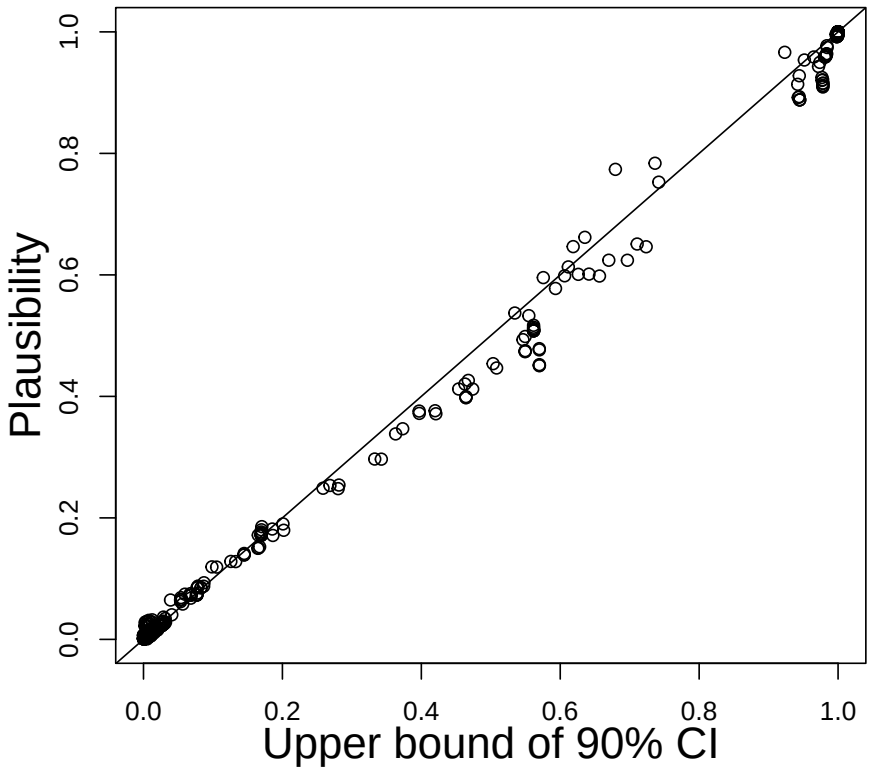




Example

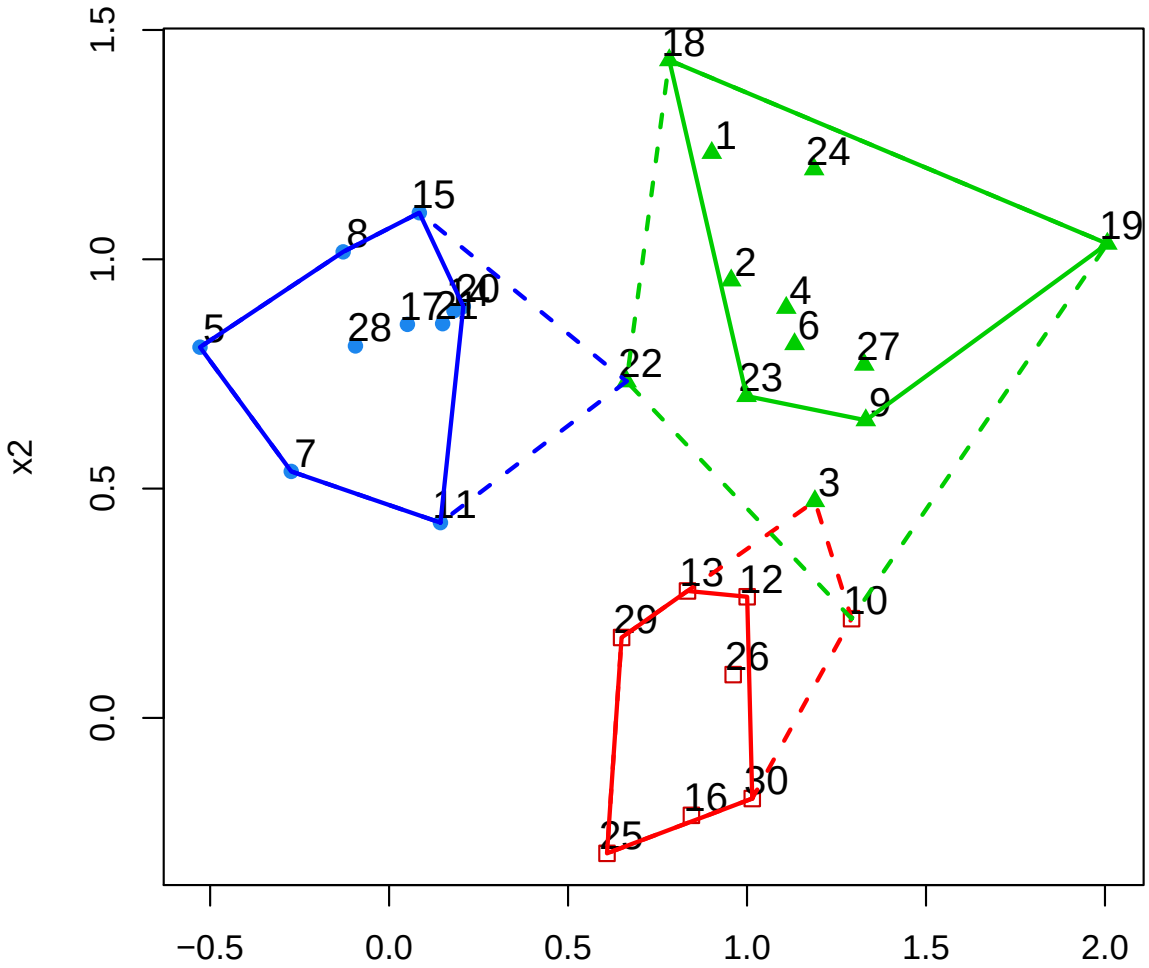
Approximation of confidence intervals by pairwise belief functions





Example

Credal partition



Outline

Classification

- We consider a population of objects partitioned in c groups (classes). Each object is described by a feature vector $\boldsymbol{X} = (X_1, \ldots, X_d) \in \mathcal{X}$ of d features and a class variable $Y \in \Theta = \{\theta_1, \ldots, \theta_c\}$ indicating group membership.
- Problem: given a learning set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ containing observations of X and Y for n objects, build a classifier

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

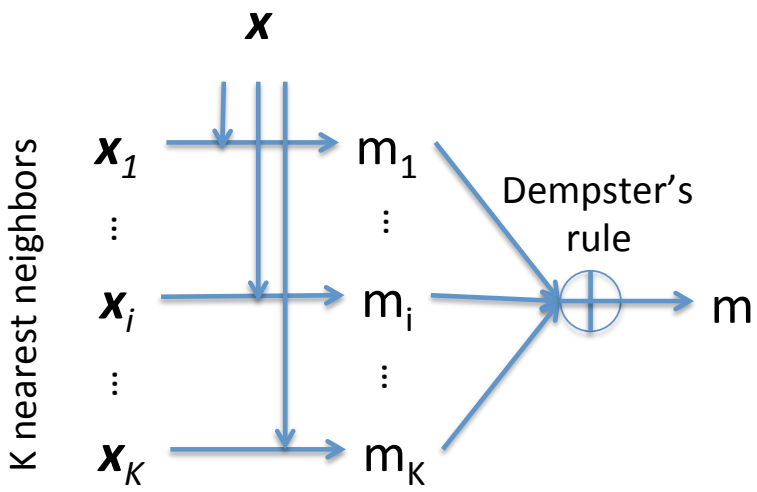
that predicts the value of Y given $\boldsymbol{X}$.

Definition

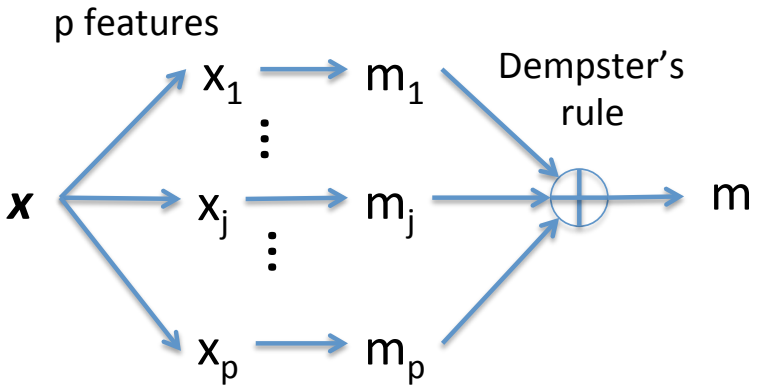
An evidential classifier is a classifier that classifies each $\boldsymbol{x} \in \mathcal{X}$ based on a mass function m on Θ constructed by aggregating evidence about the class of the object.

Two kinds of evidential classifiers

- Distance-based:

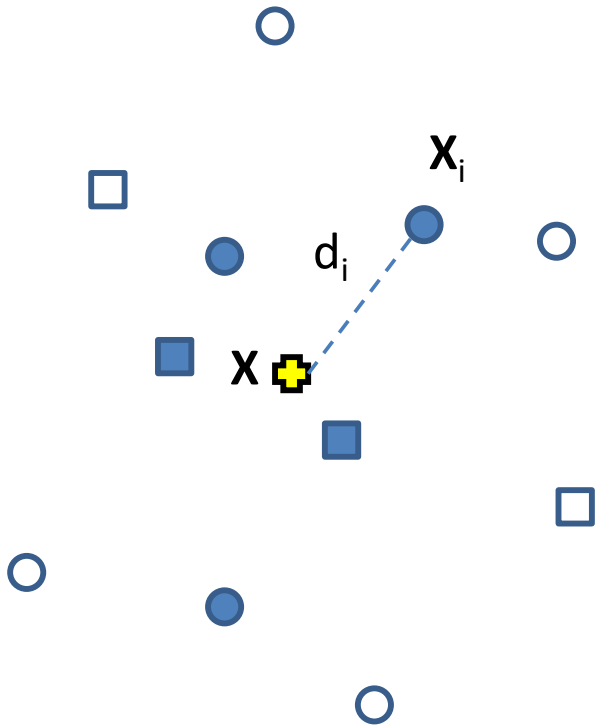


- Feature-based:



Outline

Principle[11]



- Let $N_K(\boldsymbol{x})$ denote the set of the K nearest neighbors of $\boldsymbol{x}$ in the learning set, based on some distance measure
- Each $\boldsymbol{x}_j \in N_K(\boldsymbol{x})$ can be considered as a piece of evidence regarding the class of $\boldsymbol{x}$
- The weight of this evidence decreases with the distance d_j between $\boldsymbol{x}$ and $\boldsymbol{x}_j$

[11] T. Denoeux. A k-nearest neighbor classification rule based on Dempster-Shafer theory. IEEE TSMC, 25(05):804-813, 1995

Definition

- K nearest neighbors of $\boldsymbol{x}$: $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K$, class labels $y_1, \ldots, y_K$.
- The evidence of (x_j, y_j) can be represented by a simple mass function o, $\Theta = \{\theta_1, \ldots, \theta_c\}$:

$$\widehat{m}_j := \bigoplus_{k=1}^{c} \{\theta_k\}^{\varphi_k(d_j) y_{jk}}$$

where φ_k is a decreasing function $\mathbb{R}_+ \longrightarrow \mathbb{R}_+$ and $y_{jk} := I(y_j = \theta_k)$.

- Combined evidence:

$$\widehat{m} = \bigoplus_{j=1}^{K} \widehat{m}_j = \bigoplus_{k=1}^{c} \{\theta_k\}^{w_k}$$

where

$$w_k := \sum_{\{j: y_j = \theta_k\}} \varphi_k(d_j)$$

is the total weight of evidence for class θ_k.

Learning

- Each function φ_k can be parameterized by a parameter γ_k.
- Parameter vector $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_c)$ can be learnt from the data by minimizing a cost function[12] such as:

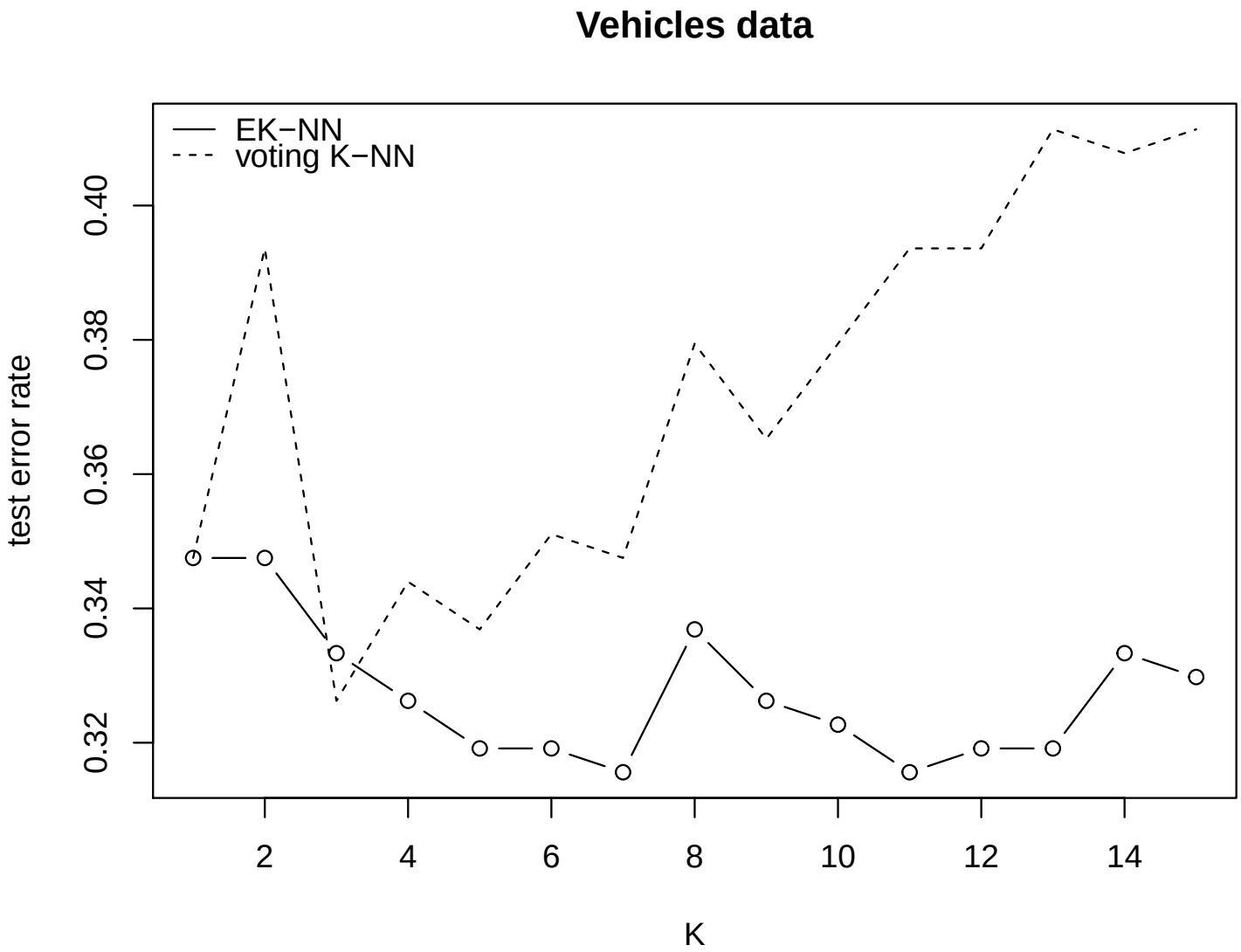
$$C_1(\boldsymbol{\gamma}) := \sum_{i=1}^{n} \sum_{k=1}^{c} (\widehat{pl}_i(\omega_k) - y_{ik})^2$$

where $\widehat{pl}_i$ is the contour function corresponding to $\widehat{m}_i$ computed using the K-NNs of observation $\boldsymbol{x}_i$.

- Implemented in R package `evclas` available at `https://CRAN.R-project.org/package=evclass`.

[12] L. M. Zouhal and T. Denoeux. An evidence-theoretic k-NN rule with parameter optimization. IEEE TSMC C, 28(2):263-271,1998.

Example


Vehicles data
EK−NN
voting K−NN
test error rate
0.32
0.34
0.36
0.38
0.40
2
4
6
8
10
12
14
K

Variants and extensions

- Neural network formulation based on prototypes[13], similar to RBF networks.
- Optimization of the dissimilarity metric[14],
- A recent version allows for fast learning in the case of labeling uncertainty[15], i.e., when each instance $\boldsymbol{x}_i$ has a soft label (a mass function on Θ).

[13] T. Denoeux. A neural network classifier based on Dempster-Shafer theory. IEEE TSMC A, 30(2):131-150, 2000.

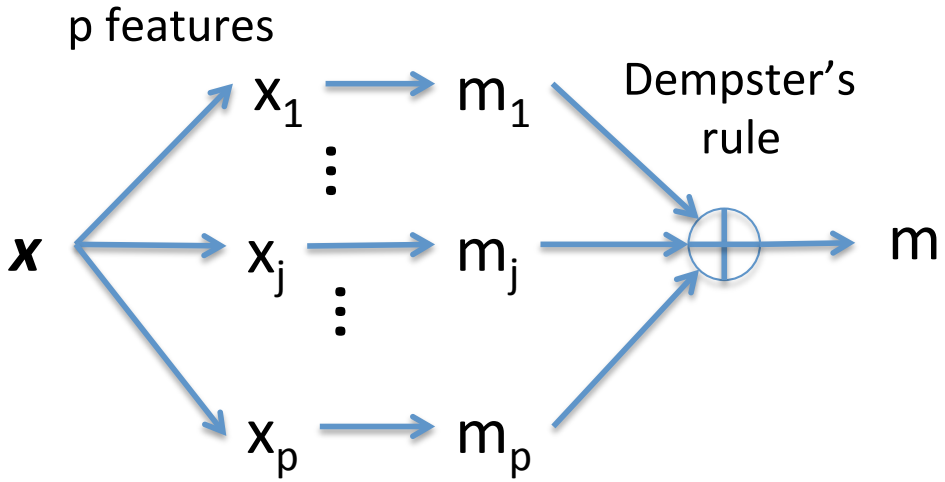
[14] C. Lian *et al.* Dissimilarity metric learning in the belief function framework. IEEE TFS 24(6):1555–1564, 2016

[15] T. Denoeux *et al.*. A New Evidential K-Nearest Neighbor Rule based on Contextual Discounting with Partially Supervised learning. IJAR, 113:287–302, 2019

Outline

Feature-based evidential classification

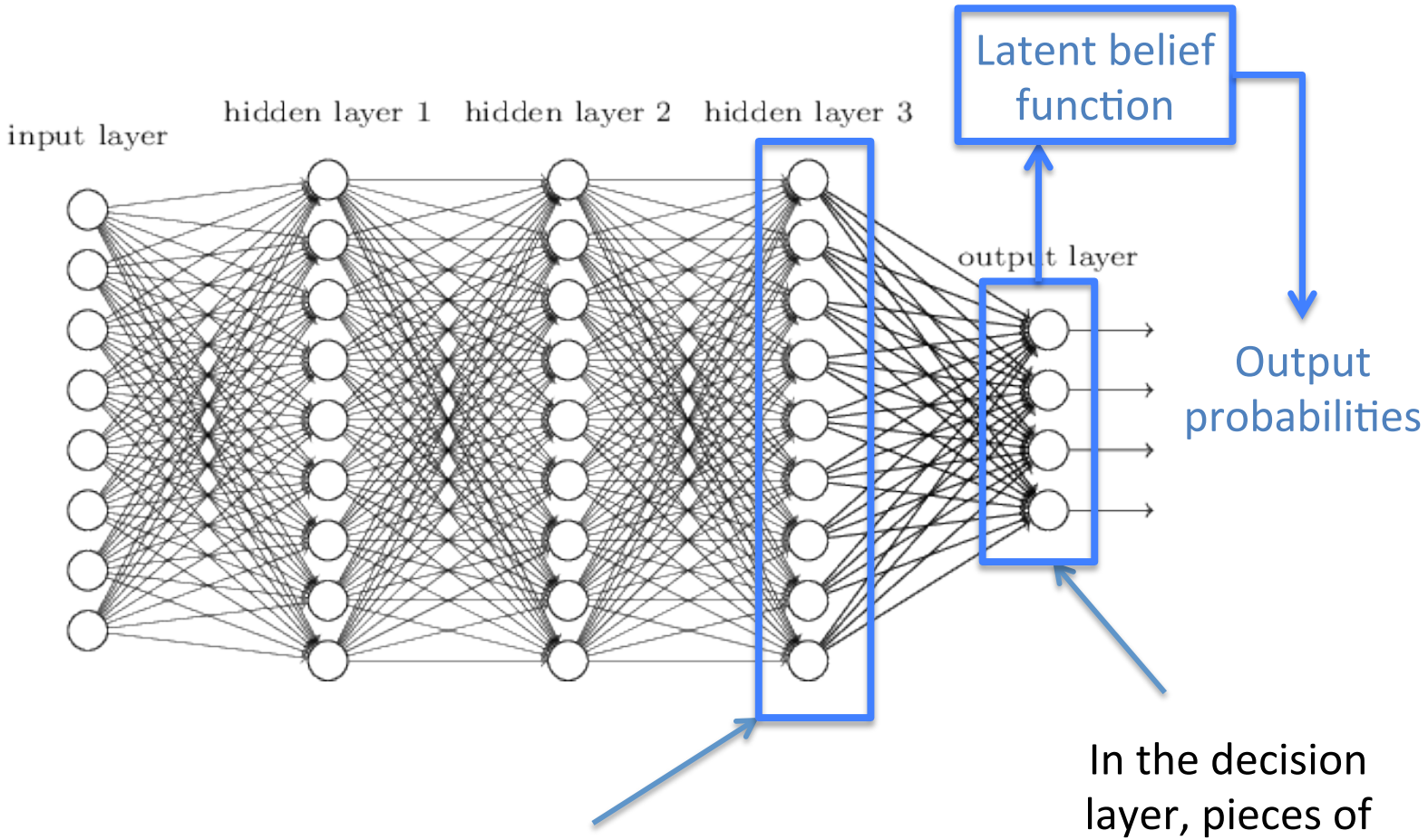
- The idea of feature-based evidential classification is to consider features as independent items of evidence and combine the corresponding mass functions by Dempster's rule:



- This seems unusual, but most ML algorithms, including neural networks can be analyzed in that way[16].

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

Neural nets: DS view



Binomial Logistic regression

- Consider a binary classification problem with d-dimensional feature vector $X = (X_1, \ldots, X_d)$ and class variable $Y \in \Theta = \{\theta_1, \theta_2\}$.
- Let $p(x)$ denote the conditional probability that $Y = \theta_1$ given that $X = x$.

Binomial Logistic Regression (LR) model

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

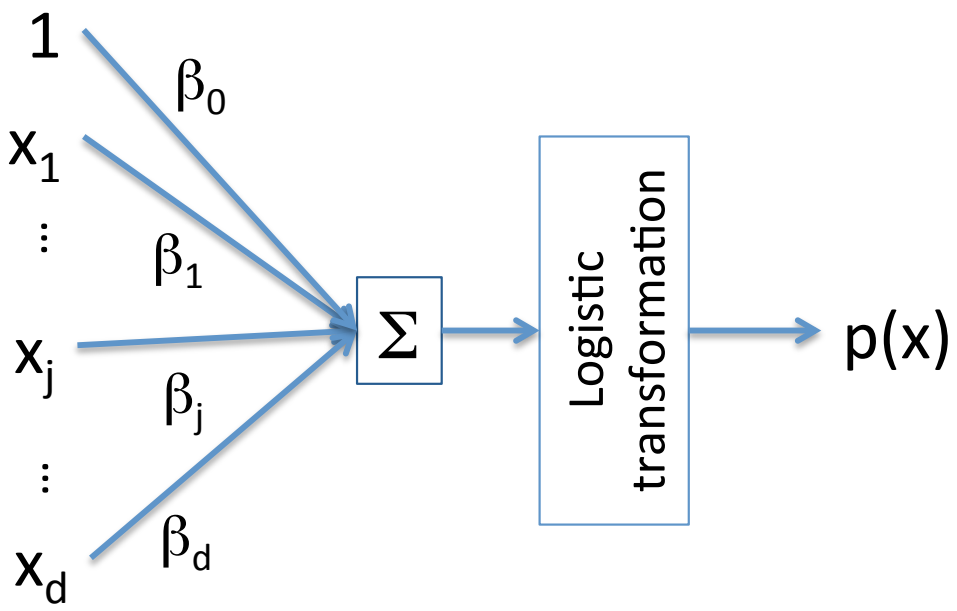
with $\beta \in \mathbb{R}^d$ and $\beta_0 \in \mathbb{R}$.

- Equivalently,

$$p(x) = \frac{1}{1 + \exp\left[-(\beta^T x + \beta_0)\right]} = \Lambda(\beta^T x + \beta_0)$$

where Λ is the logistic function.

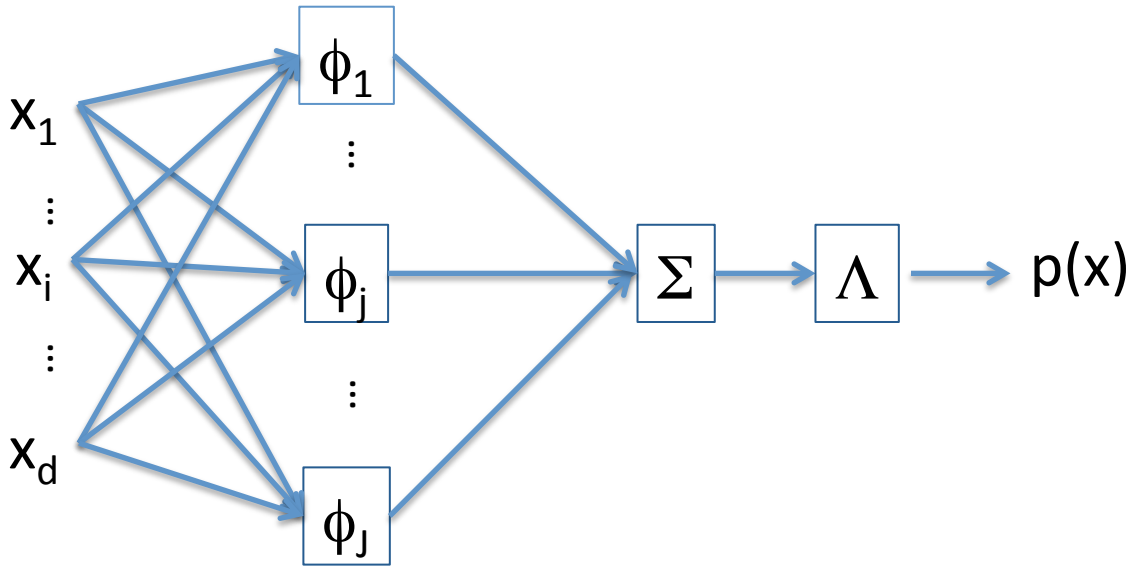
Binomial Logistic Regression (continued)



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:

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

Nonlinear generalized LR classifiers



- 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.
- Popular models based on this principle:
  - Radial basis function networks
  - Support vector machines
  - Multilayer feedforward neural networks (NNs)

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.

Model

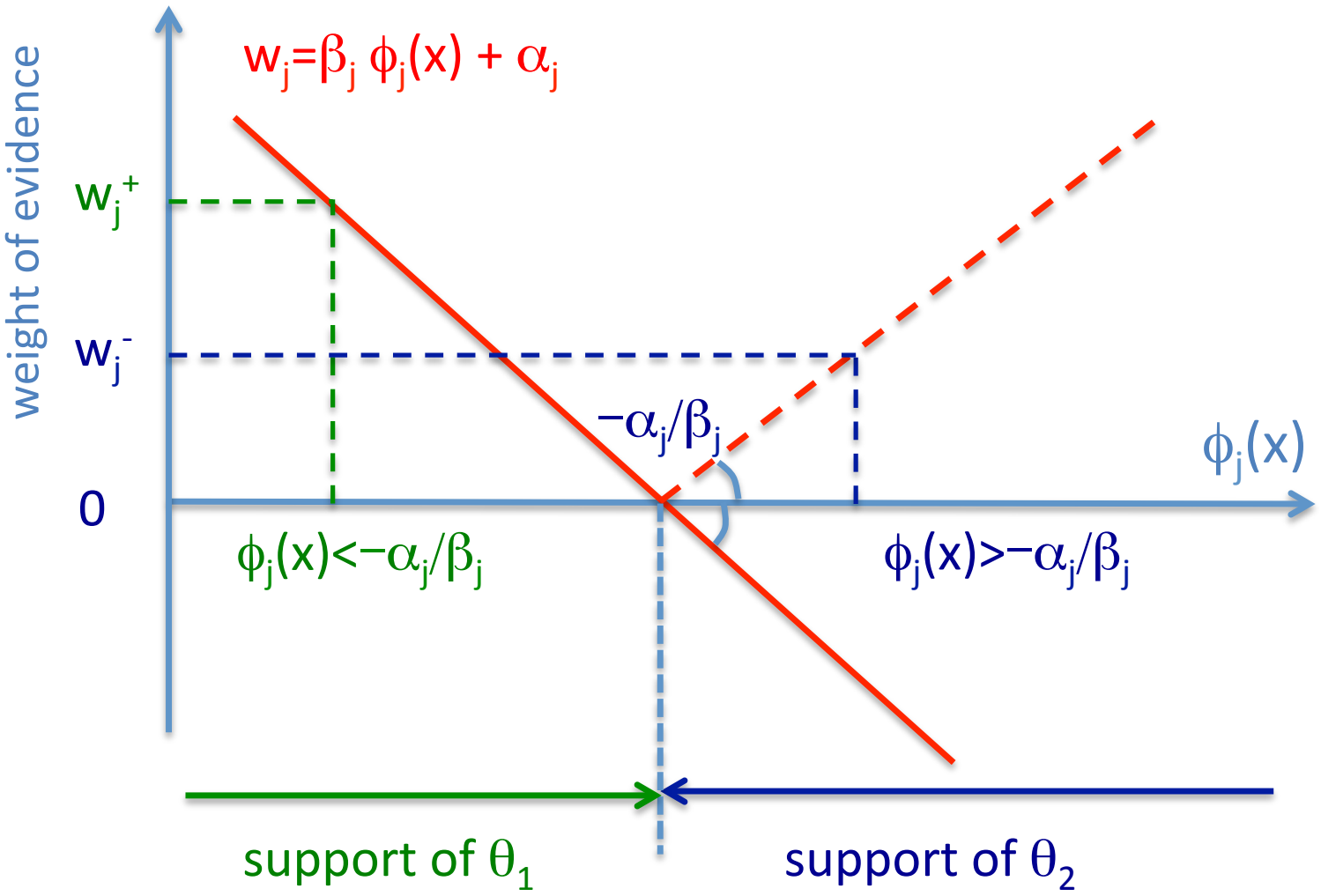
- The value $\phi_j(x)$ taken by feature j is a piece of evidence about the class $Y \in \Theta$ of the instance under consideration.
- This evidence points to θ_1 or θ_2 depending on the sign of

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

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: illustration



Feature-based latent mass function

Under this model, the consideration of feature ϕ_j induces the following 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 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

Normalized plausibilities

The normalized plausibility of class θ_1 as

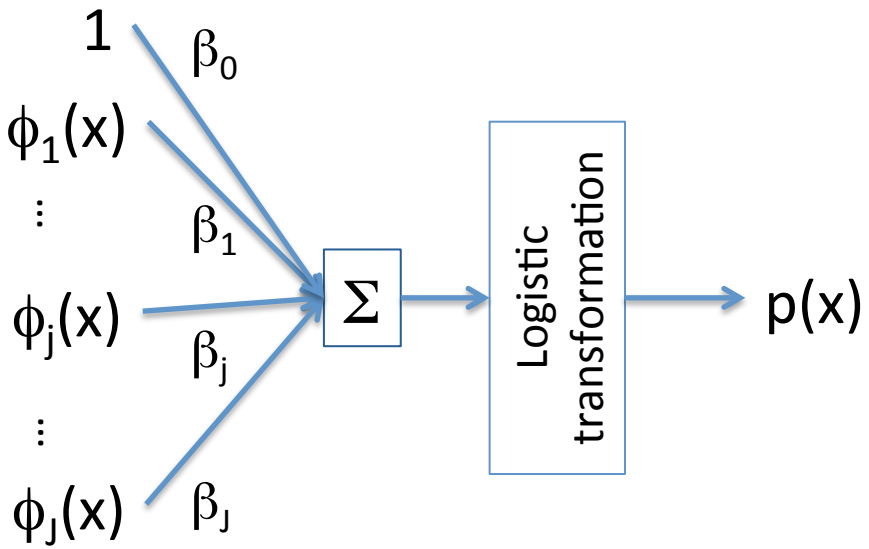
$$\frac{Pl(\{\theta_1\})}{Pl(\{\theta_1\}) + Pl(\{\theta_2\})} = \frac{m(\{\theta_1\}) + m(\Theta)}{m(\{\theta_1\}) + m(\{\theta_2\}) + 2m(\Theta)}$$
$$= \underbrace{\frac{1}{1 + \exp\left[-(\boldsymbol{\beta}^T \phi(x) + \beta_0)\right]}}_{\text{logistic transformation}} = p(x)$$

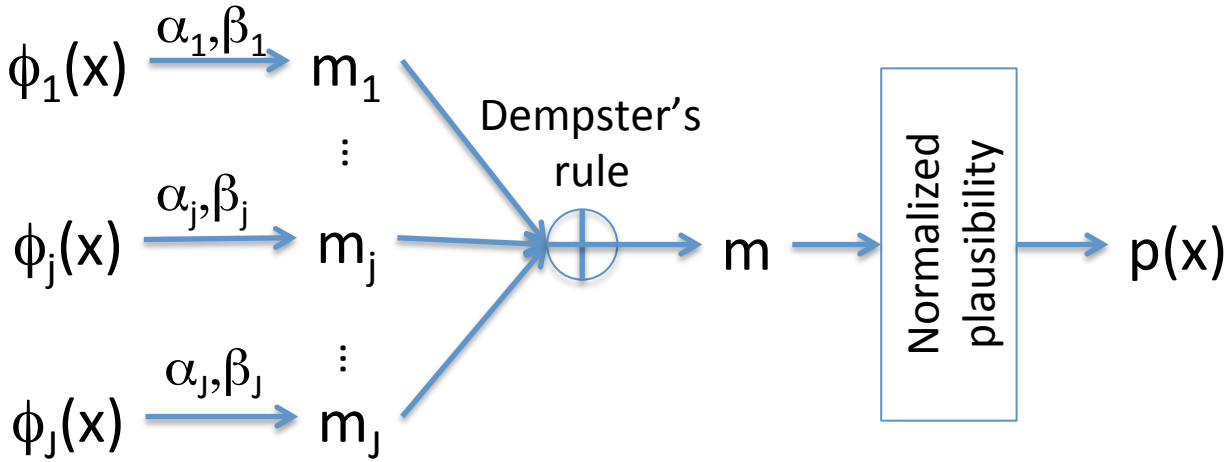
with $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_J)$ and $\beta_0 = \sum_{i=1}^{J} \alpha_j$.

Proposition

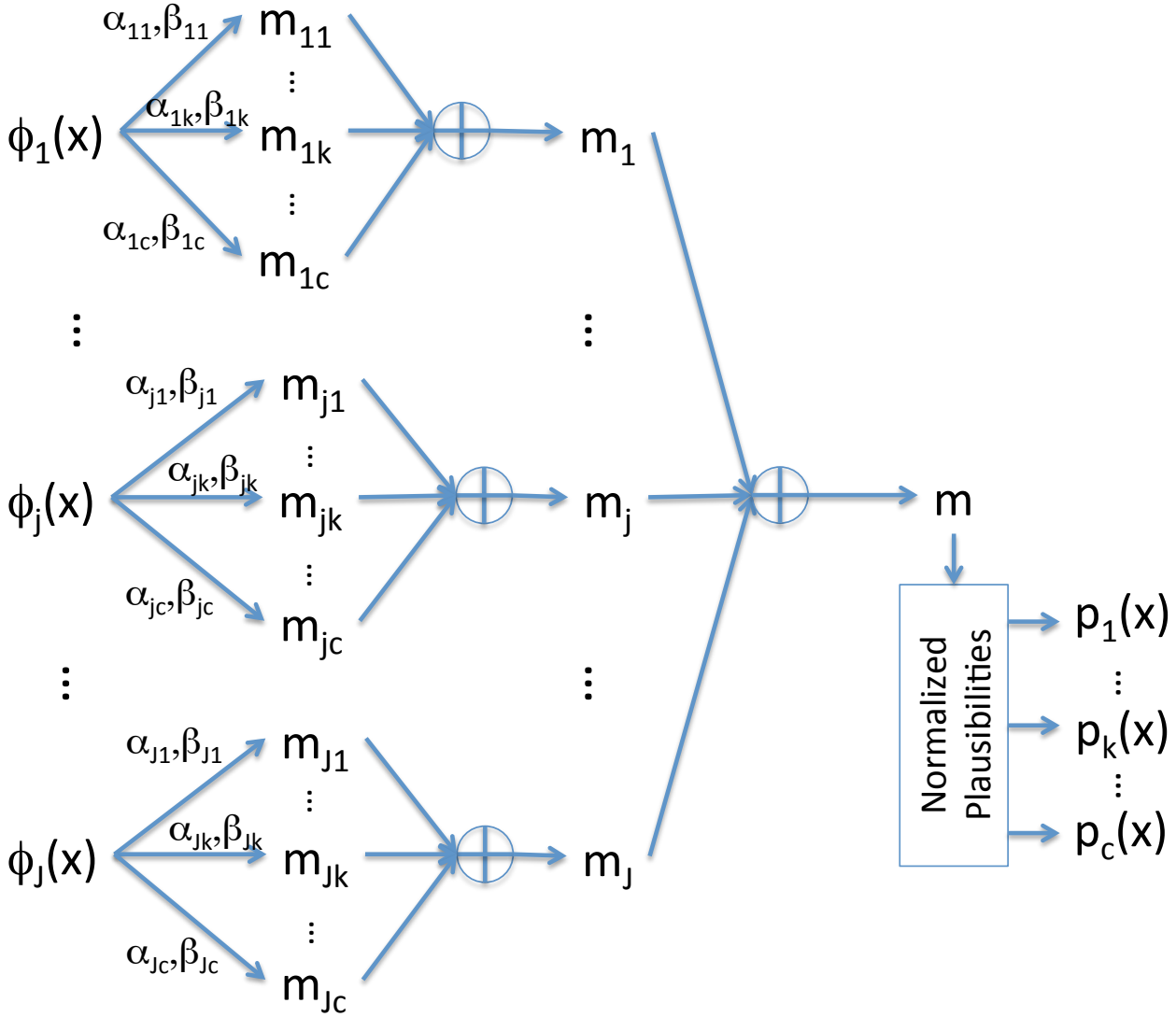
The normalized plausibilities are equal to the conditional class probabilities of the binomial LR model: the two models are equivalent.

Two Views of Binomial Logistic Regression



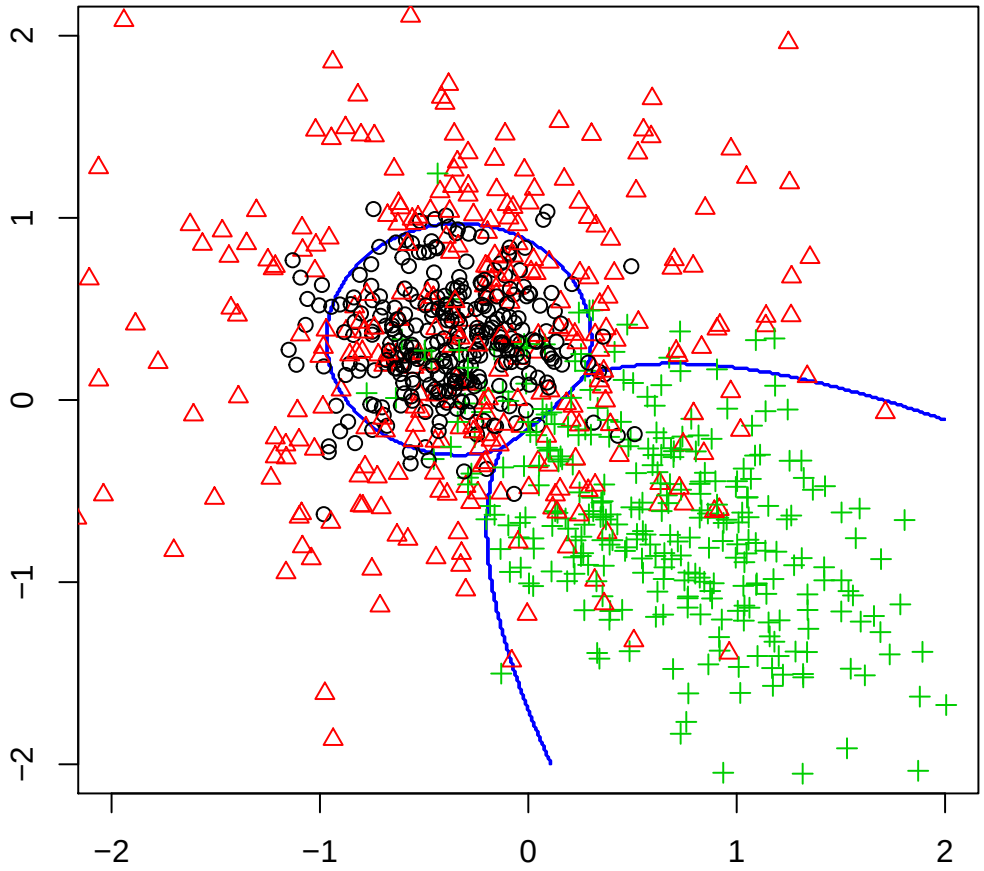


Multinomial Logistic Regression: DS view



Example

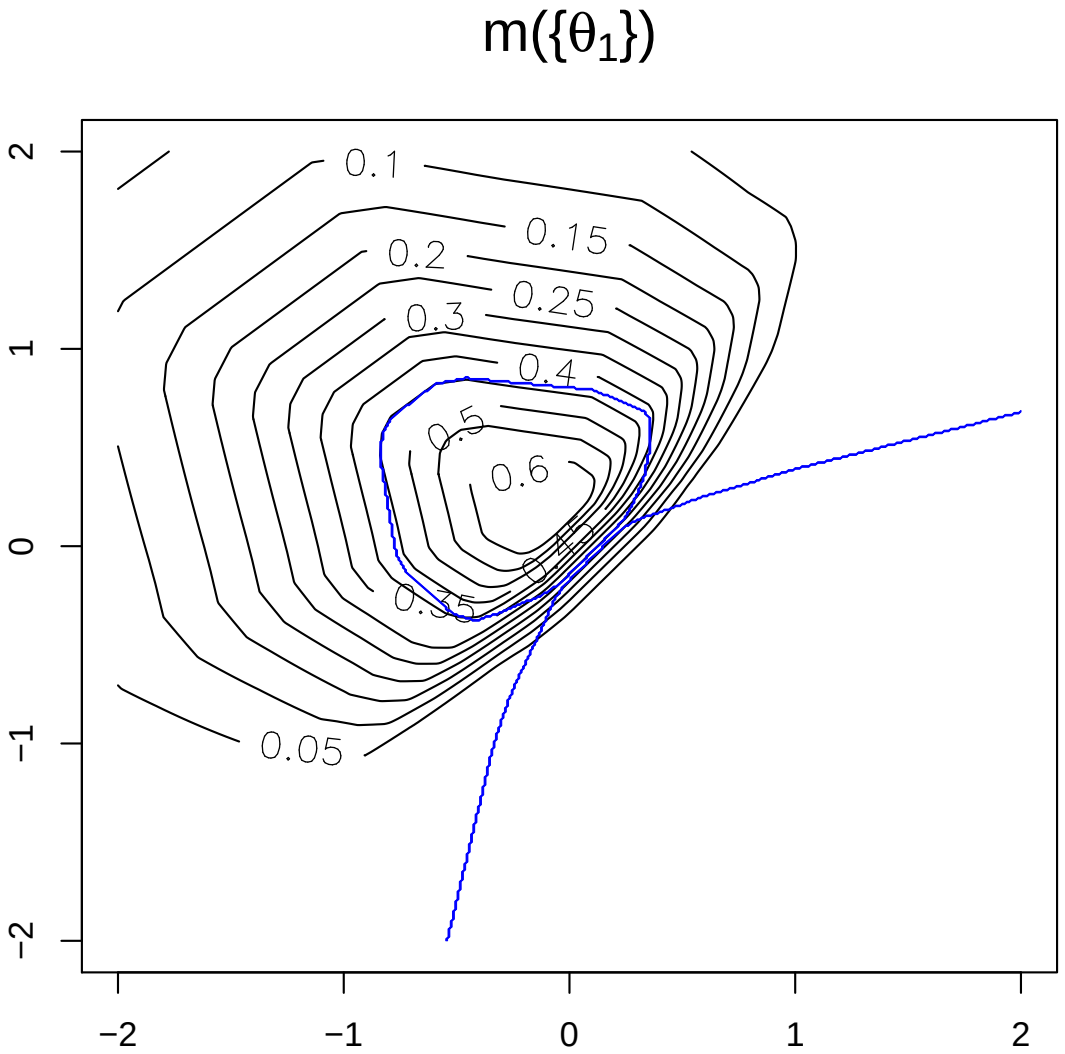
Dataset: 900 instances, 3 equiprobable classes with Gaussian distributions



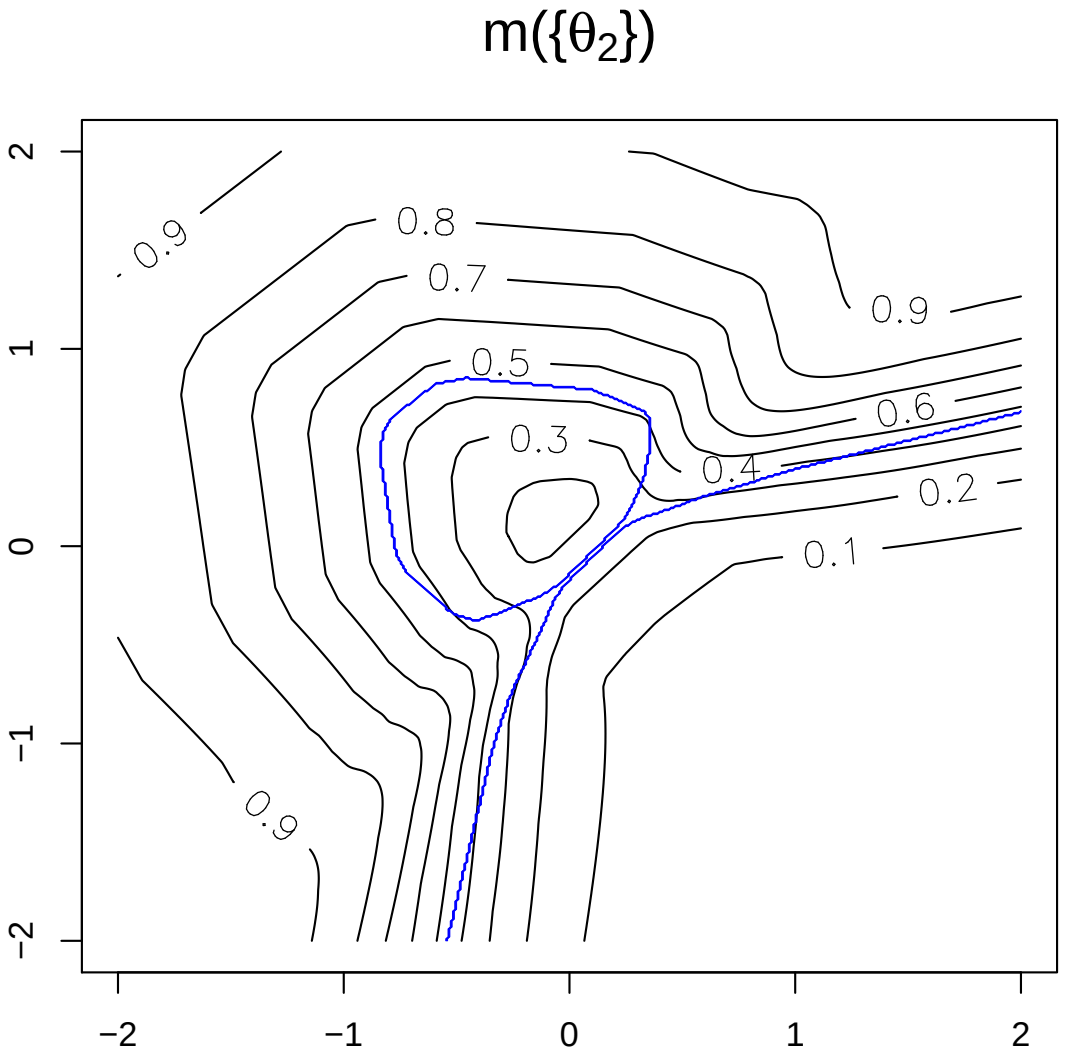
NN model

- 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 ($\lambda = 1$).

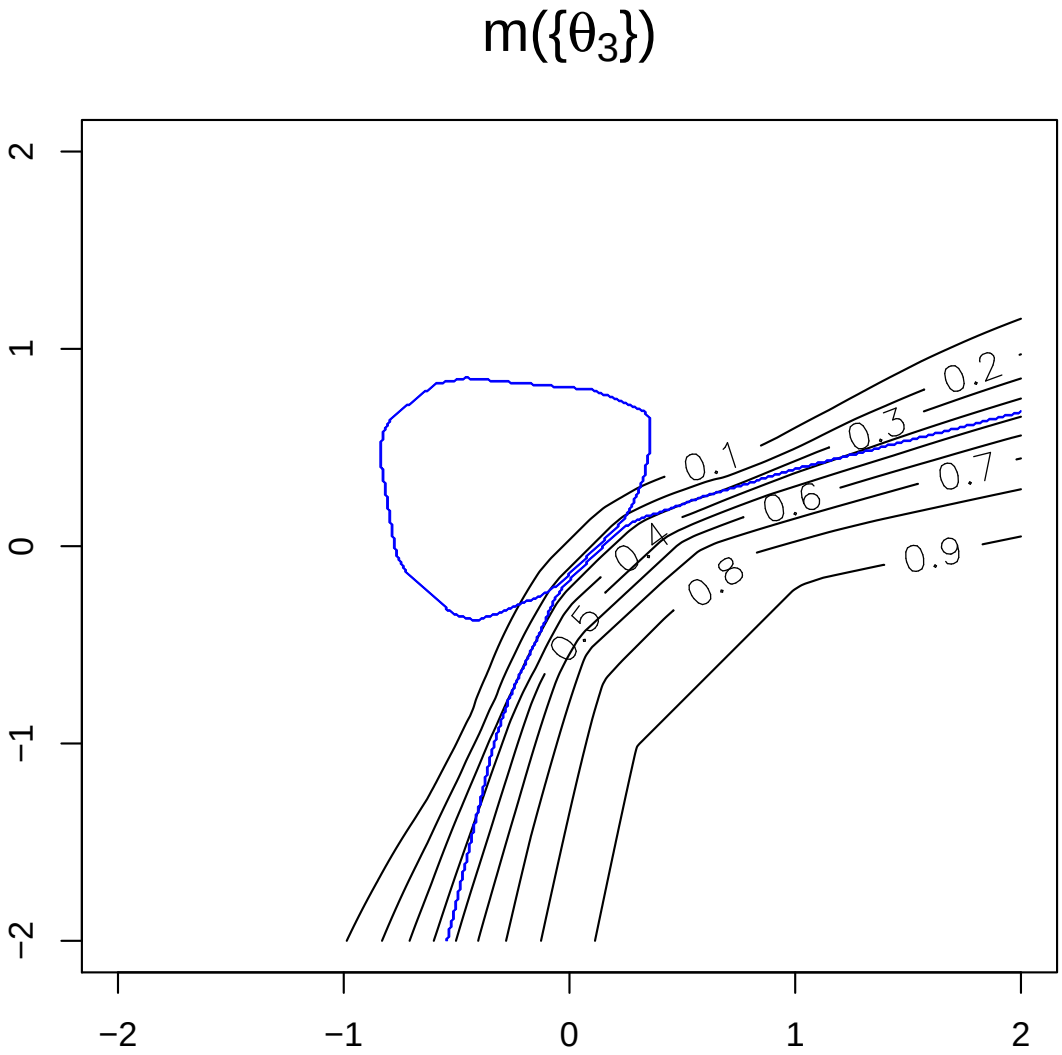
Mass on $\{\theta_1\}$



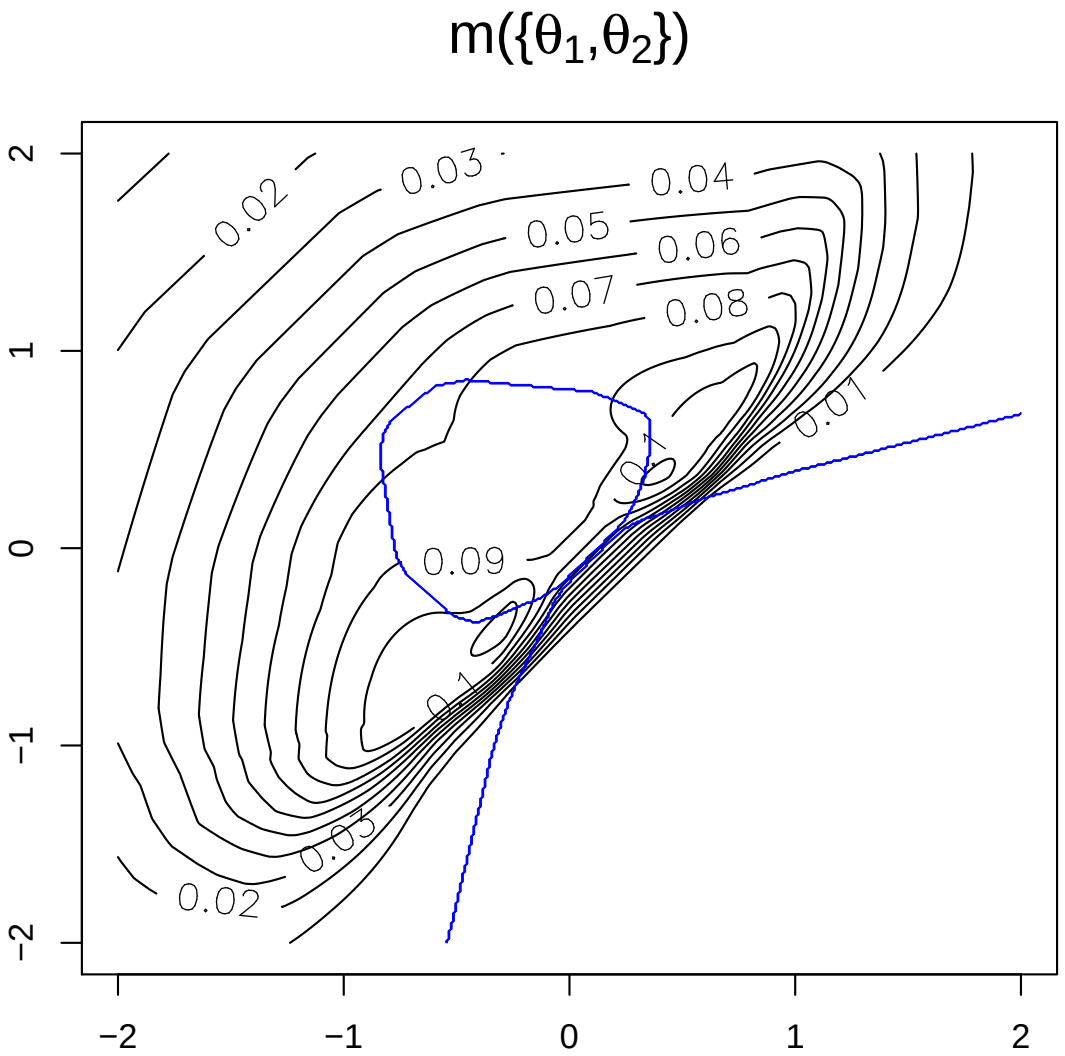
Mass on $\{\theta_2\}$



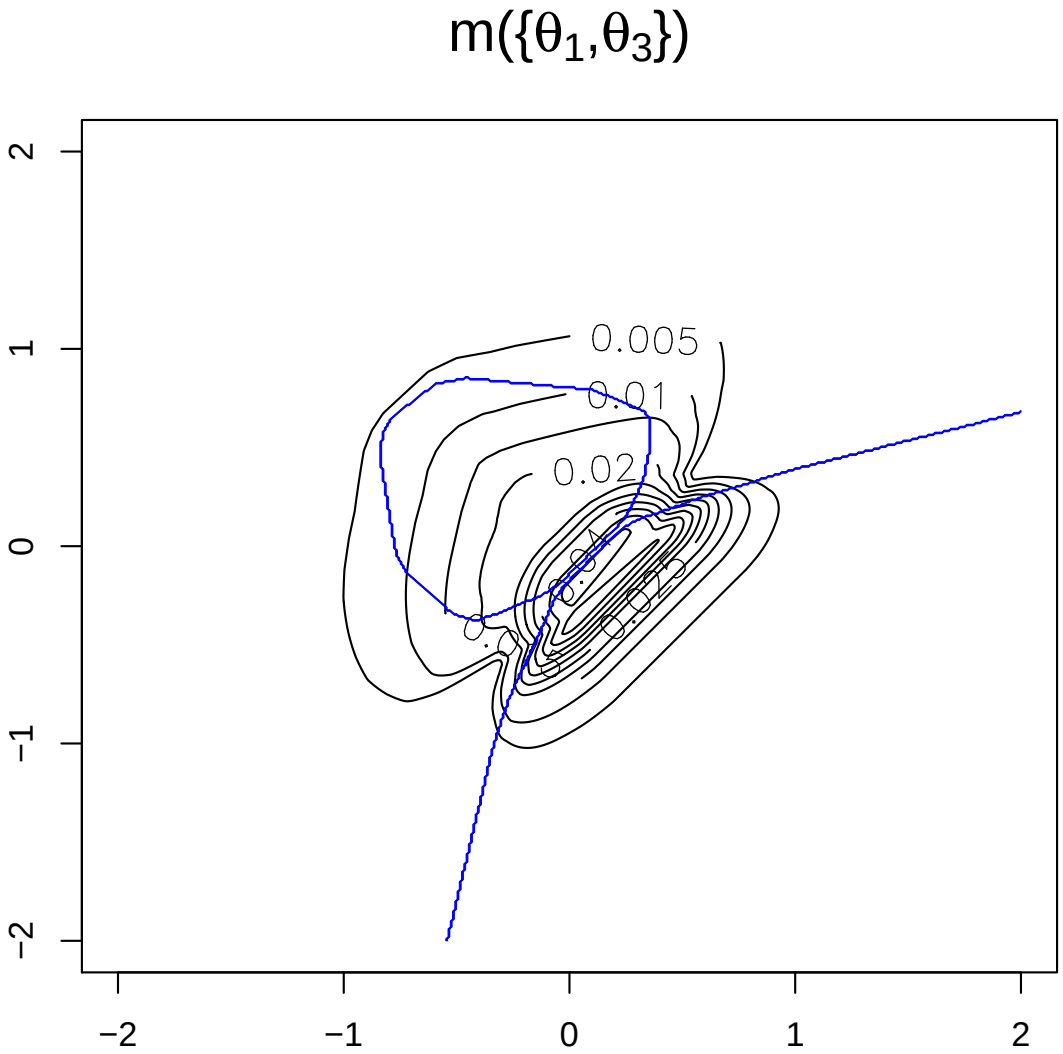
Mass on $\{\theta_3\}$



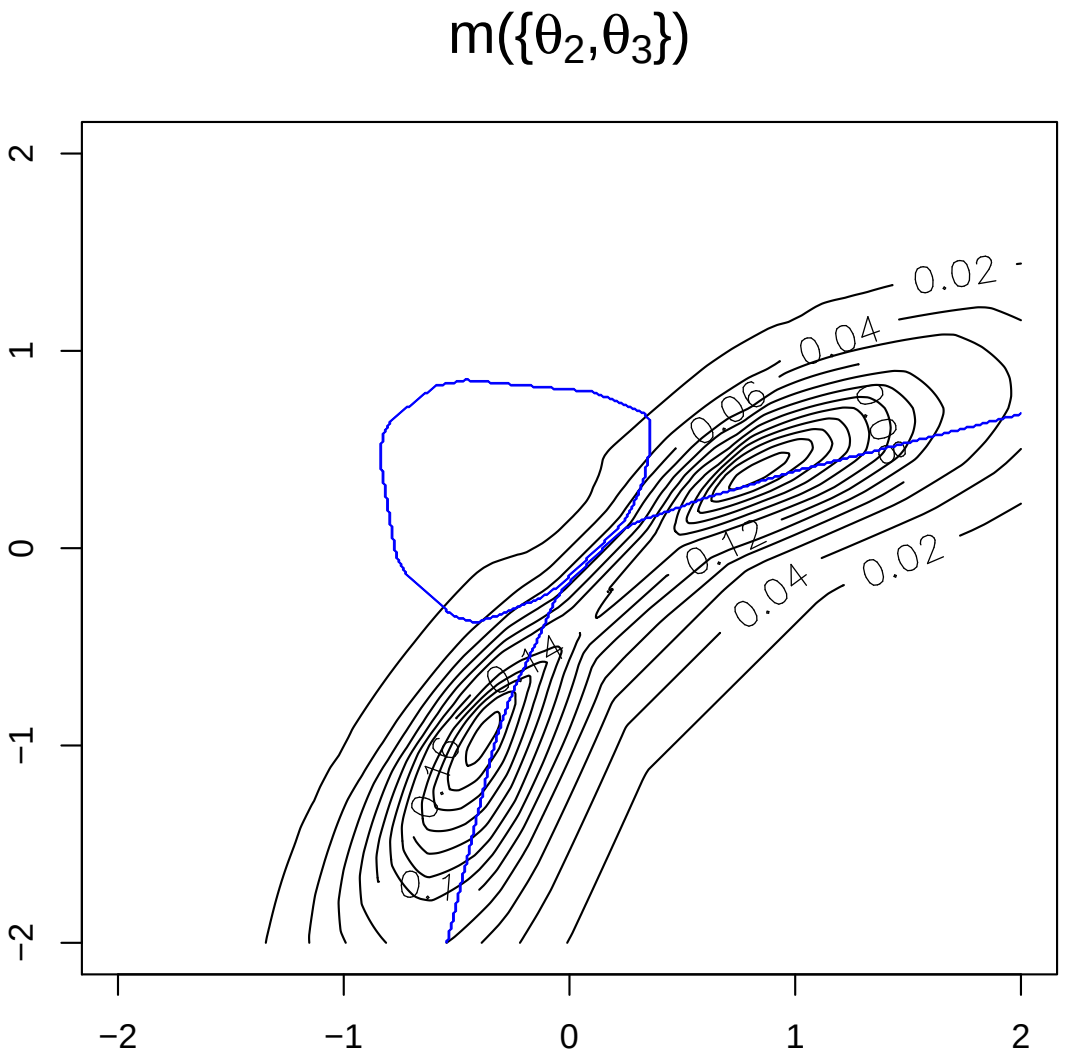
Mass on $\{\theta_1, \theta_2\}$



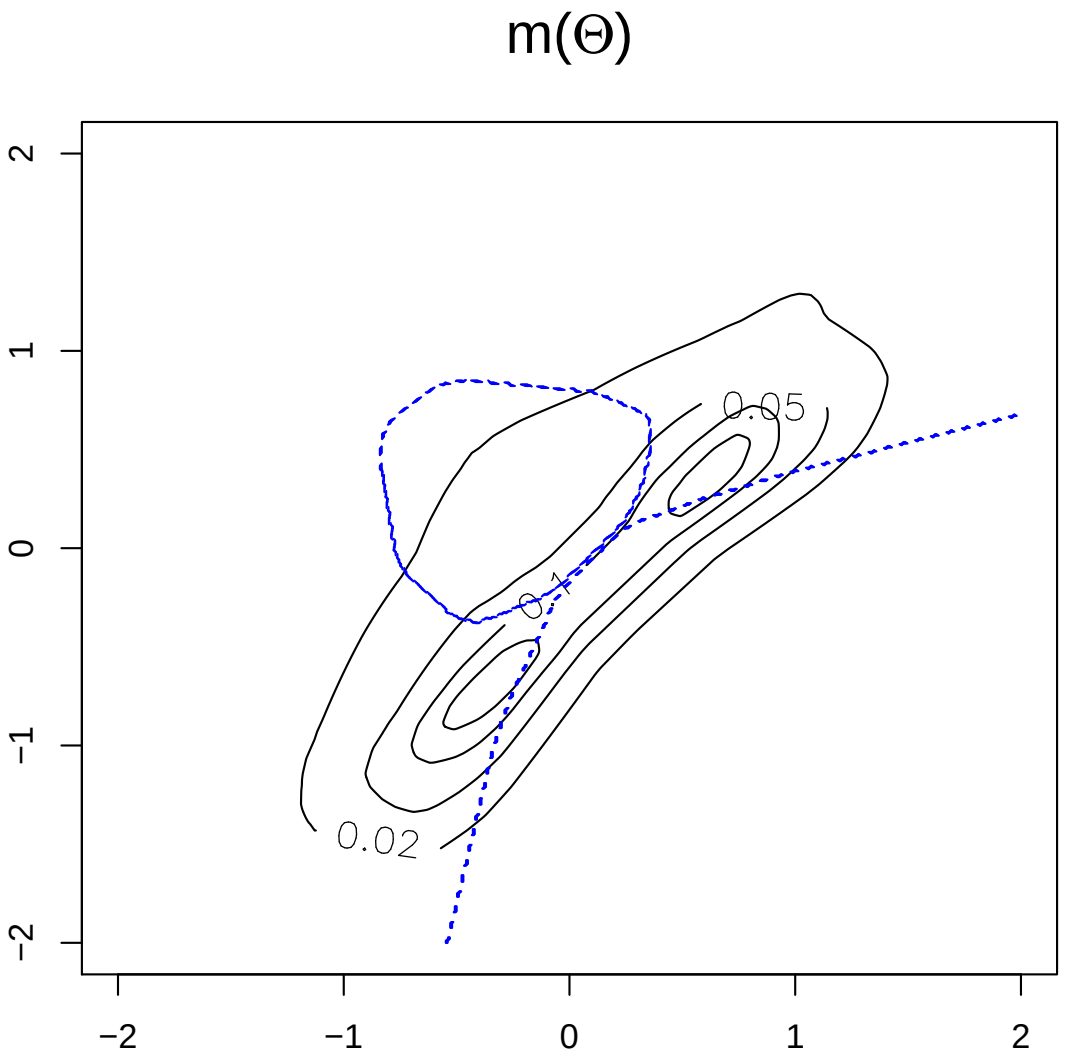
Mass on $\{\theta_1, \theta_3\}$



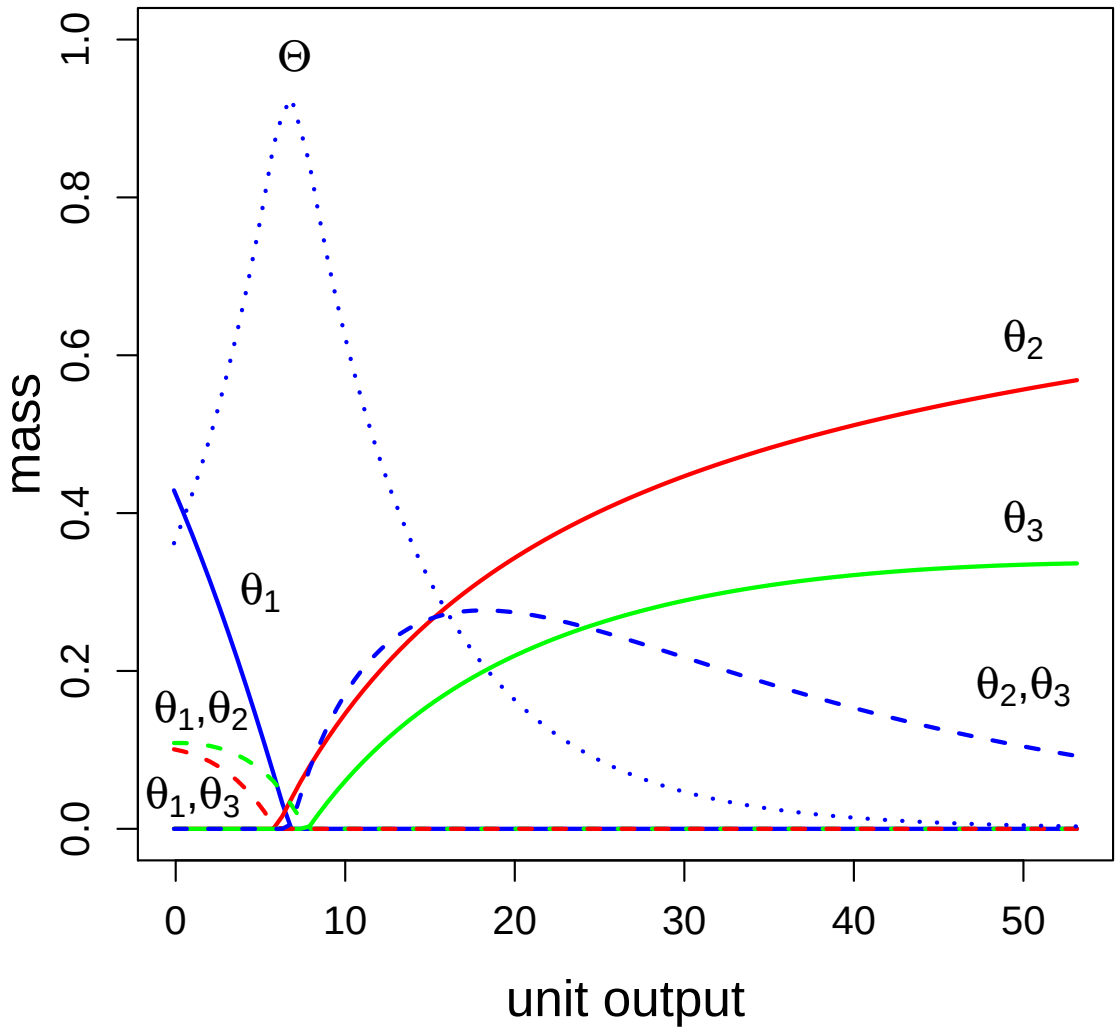
Mass on $\{\theta_2, \theta_3\}$



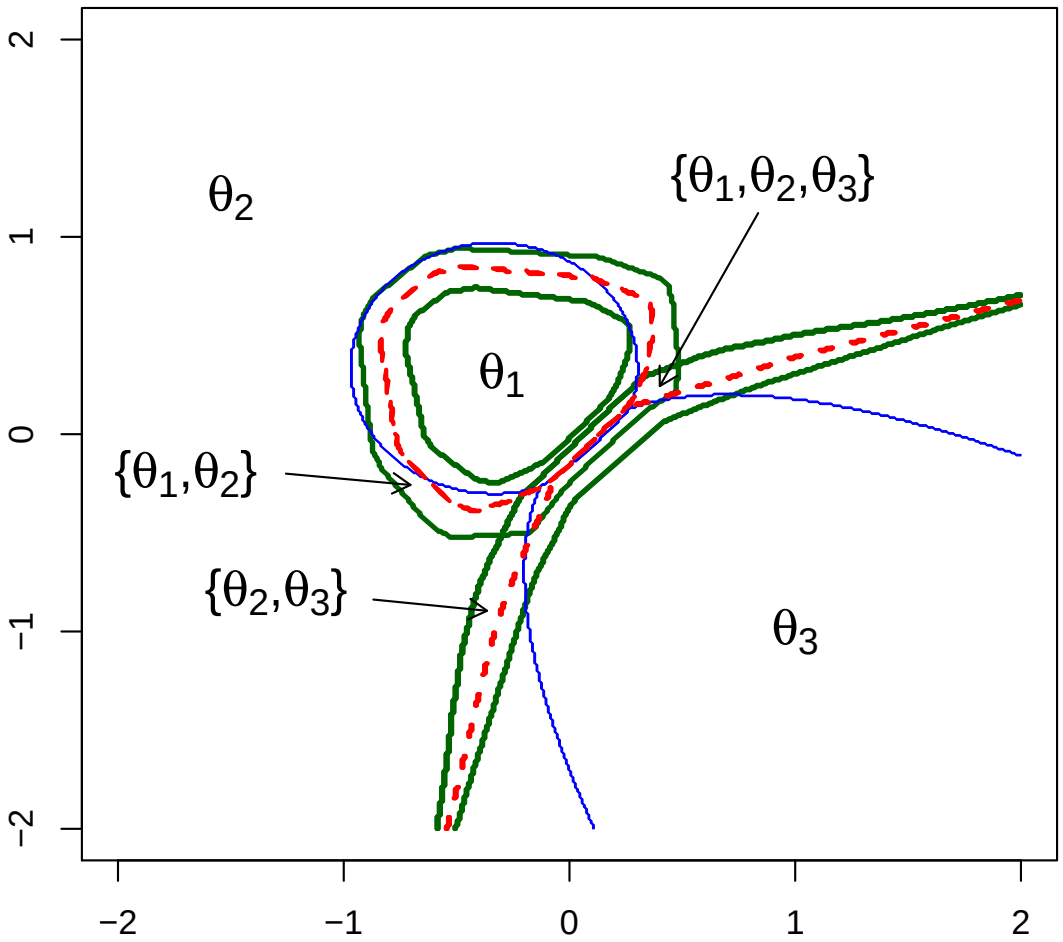
Mass on Θ



Hidden unit 2



Decision regions (Interval Dominance rule)



Summary

- Until recently, ML has been mostly based on probability theory. As a more general model, DS theory offers a radically new and promising approach to uncertainty quantification in ML.
- Other applications of belief functions in ML include
  - Classifier/clusterer ensembles
  - Partially labeled data
  - Constrained clustering
  - Multilabel classification
  - Preference learning, etc.
- Many classical ML techniques can be revisited from a DS perspective, with possible implications in terms of
  - Interpretation
  - Decision strategies
  - Model combination, etc.

References

T. Denœux, D. Dubois and H. Prade.
Representations of Uncertainty in Artificial Intelligence: Beyond Probability and Possibility
In P. Marquis, O. Papini and H. Prade (Eds), A Guided Tour of Artificial Intelligence Research (Chapter 4), Springer Verlag, 2020 (to appear).

Full text of all my papers and other resources at:

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

THANK YOU !