Statistical Analysis of Uncertain Data in the Belief Function Framework

Thierry Denœux

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

Chiang Mai University
August 2016

Outline

Introductory example

- Let us consider a population in which some disease is present in proportion θ.
- n patients have been selected at random from that population. Let $x_i = 1$ if patient i has the disease, $x_i = 0$ otherwise. Each x_i is a realization of $X_i \sim \mathcal{B}(\theta)$.
- We assume that the x_i's are not observed directly. For each patient i, a physician gives a degree of plausibility $pl_i(1)$ that patient i has the disease and a degree of plausibility $pl_i(0)$ that patient i does not have the disease.
- The observations are uncertain data of the form $pl_1, \ldots, pl_n$.
- How to estimate θ?

Aleatory vs. epistemic uncertainty

- In the previous example, uncertainty has two distinct origins:
  1. Before a patient has been drawn at random from the population, uncertainty is due to the variability of the variable of interest in the population. This is aleatory uncertainty.
  2. After the random experiment has been performed, uncertainty is due to lack of knowledge of the state of each particular patient. This is epistemic uncertainty.
- Epistemic uncertainty can be reduced by carrying out further investigations. Aleatory uncertainty cannot.

Approach

- In this lecture, we will consider statistical estimation problems in which both kinds of uncertainty are present: it will be assumed that each data item x
    - has been generated at random from a population (aleatory uncertainty), but
    - it is ill-known because of imperfect measurement or perception (epistemic uncertainty).
- The proposed model treats these two kinds of uncertainty separately:
    - Aleatory uncertainty will be represented by a parametric statistical model;
    - Epistemic uncertainty will be represented using belief functions.

Real world applications

Uncertain data arise in many applications (but epistemic uncertainty is usually neglected). It may be due to:

- Limitations of the underlying measuring equipment (unreliable sensors, indirect measurements), e.g.: biological sensor for toxicity measurement in water.
- Use of imputation, interpolation or extrapolation techniques, e.g.: clustering of moving objects whose position is measured asynchronously by a sensor network,
- Partial or uncertain responses in surveys or subjective data annotation, e.g.: sensory analysis experiments, data labeling by experts, etc.

Data labeling example

Recognition of facial expressions

joy

surprise

sadness

disgust

anger

fear

Recognition of facial expressions

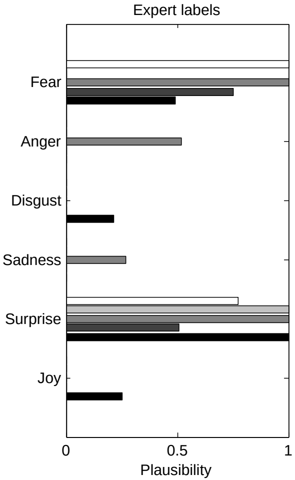
Experiment

- To achieve good performances in such tasks (object classification in images or videos), we need a large number of labeled images.
- However, ground truth is usually not available or difficult to determine with high precision and reliability: it is necessary to have the images subjectively annotated (labeled) by humans.
- How to account for uncertainty in such subjective annotations?
- Experiment:
  - Images were labeled by 5 subjects;
  - For each image, subjects were asked to give a degree of plausibility for each of the 6 basic expressions.

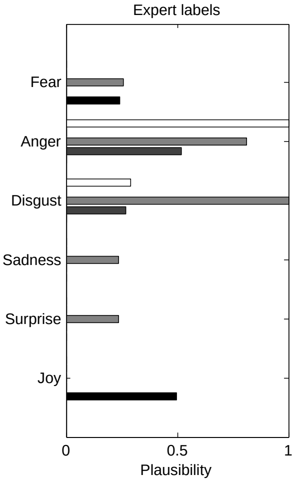
Example 1


Expert labels
Fear
Anger
Disgust
Sadness
Surprise
Joy
0
0.5
1
Plausibility

Example 2


Expert labels
Fear
Anger
Disgust
Sadness
Surprise
Joy
0
0.5
1
Plausibility

Example 3


Expert labels
Fear
Anger
Disgust
Sadness
Surprise
Joy
0
0.5
1
Plausibility

Model

- Complete data: $\boldsymbol{x} = \{(\boldsymbol{w}_i, z_i)\}_{i=1}^n$ with
  - $\boldsymbol{w}_i$: feature vector for image i (pixel gray levels)
  - z_i: class of image i (one the six expressions).
- The feature vectors $\boldsymbol{w}_i$ are perfectly observed but class labels are only partially known through subjective evaluations.
- How to learn a decision rule from such data?

General approach

1. Postulate a parametric statistical model $p_{\boldsymbol{x}}(\boldsymbol{x};\theta)$ for the complete data;
2. Represent epistemic data uncertainty using belief functions (observed data);
3. Estimate θ by minimizing the conflict between the model and the observed data using an extension of the EM algorithm: the evidential EM (E^2M) algorithm.
4. Applications:
    1. Probability estimation (Bernoulli model)
    2. Linear discriminant analysis with uncertain class labels
    3. Linear regression with fuzzy data

Outline

Outline

Model

- Let $\boldsymbol{X}$ be a (discrete) random vector taking values in $\Omega_{\boldsymbol{X}}$, with probability mass function $p_{\boldsymbol{X}}(\cdot; \boldsymbol{\theta})$ depending on an unknown parameter $\boldsymbol{\theta} \in \Theta$.
- Let $\boldsymbol{x}$ be a realization of $\boldsymbol{X}$ (complete data).
- We assume that $\boldsymbol{x}$ is only partially observed, and partial knowledge of $\boldsymbol{x}$ is described by a mass function m on $\Omega_{\boldsymbol{X}}$ (“observed” data).
- Problem: estimate $\boldsymbol{\theta}$.

Likelihood function (reminder)

- Given a parametric model $p_{\boldsymbol{X}}(\cdot;\theta)$ and an observation $\boldsymbol{x}$, the likelihood function is the mapping from Θ to $[0,1]$ defined as

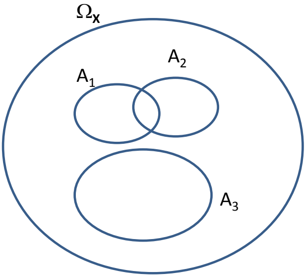
$$\theta \rightarrow L(\theta;\boldsymbol{x}) = p_{\boldsymbol{X}}(\boldsymbol{x};\theta).$$

- It measures the "likelihood" or plausibility of each possible value of the parameter, after the data has been observed.
- If we observe that $\boldsymbol{x} \in A$, then the likelihood function is:

$$L(\theta;A) = \mathbb{P}_{\boldsymbol{X}}(A;\theta) = \sum_{\boldsymbol{x}\in A} p_{\boldsymbol{X}}(\boldsymbol{x};\theta).$$

Generalized Likelihood function

Definition



- Assume that m has focal sets $A_1, \ldots, A_r$.
- If we knew that $\boldsymbol{x} \in A_i$, the likelihood would be

$$L(\theta; A_i) = \mathbb{P}_{\boldsymbol{X}}(A_i; \theta) = \sum_{\boldsymbol{x} \in A_i} p_{\boldsymbol{X}}(\boldsymbol{x}; \theta).$$

- Taking the expectation with respect to m:

$$L(\theta; m) = \sum_{i=1}^{r} m(A_i) L(\theta; A_i)$$

Generalized Likelihood function

Interpretation

- We have

$$
\begin{aligned}
L(\theta; m) &= \sum_{i=1}^{r} m(A_i) \sum_{\boldsymbol{x} \in A_i} p_{\boldsymbol{X}}(\boldsymbol{x}; \theta) \\
&= \sum_{\boldsymbol{x} \in \Omega_{\boldsymbol{X}}} p_{\boldsymbol{X}}(\boldsymbol{x}; \theta) \sum_{A_i \ni \boldsymbol{x}} m(A_i) \\
&= \sum_{\boldsymbol{x} \in \Omega_{\boldsymbol{X}}} p_{\boldsymbol{X}}(\boldsymbol{x}; \theta) pl(\boldsymbol{x}) = 1 - \kappa,
\end{aligned}
$$

  where κ is the degree of conflict between $p_{\boldsymbol{X}}(\cdot; \theta)$ and m.
- Consequently, maximizing $L(\theta; m)$ with respect to θ amounts to minimizing the conflict between the parametric model and the uncertain observations

Generalized Likelihood function

Case of fuzzy data

- We can also write $L(\theta; m)$ as:

$$L(\theta; m) = \sum_{\boldsymbol{x} \in \Omega_{\boldsymbol{X}}} p_{\boldsymbol{X}}(\boldsymbol{x}; \theta) pl(\boldsymbol{x}) = \mathbb{E}_{\theta}\left[pl(\boldsymbol{X})\right]$$

- If m is consonant, pl may be interpreted as the membership function of a fuzzy subset of Ω_X: it can be seen as fuzzy data.
- $L(\theta; m)$ is then the probability of the fuzzy data, according to the definition given by Zadeh (1968).

Independence assumptions

- Let us assume that $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) \in \mathbb{R}^{np}$, where each $\boldsymbol{x}_i$ is a realization from a p-dimensional random vector $\boldsymbol{X}_i$.
- Independence assumptions:
  1. Stochastic independence of $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$:
  $$p_{\boldsymbol{X}}(\boldsymbol{x};\boldsymbol{\theta}) = \prod_{i=1}^{n} p_{\boldsymbol{X}_i}(\boldsymbol{x}_i;\boldsymbol{\theta}), \quad \forall \boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) \in \Omega_{\boldsymbol{X}}$$
  2. Cognitive independence of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ with respect to m:
  $$pl(\boldsymbol{x}) = \prod_{i=1}^{n} pl_i(\boldsymbol{x}_i), \quad \forall \boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) \in \Omega_{\boldsymbol{X}}.$$
- Under these assumptions:
$$\log L(\boldsymbol{\theta}; m) = \sum_{i=1}^{n} \log \mathbb{E}_{\boldsymbol{\theta}} \left[pl_i(\boldsymbol{X}_i) \right].$$

Outline

Description

- The generalized log-likelihood function $\log L(\boldsymbol{\theta}; m)$ can be maximized using an iterative algorithm composed of two steps:

  E-step: Compute the expectation of $\log L(\theta; \boldsymbol{X})$ with respect to $m \oplus p_{\boldsymbol{X}}(\cdot; \boldsymbol{\theta}^{(q)})$:

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)}) = \frac{\sum_{\boldsymbol{x} \in \Omega_X} \log(L(\boldsymbol{\theta}; \boldsymbol{x})) p_{\boldsymbol{X}}(\boldsymbol{x}; \boldsymbol{\theta}^{(q)}) pl(\boldsymbol{x})}{\sum_{\boldsymbol{x} \in \Omega_X} p_{\boldsymbol{X}}(\boldsymbol{x}; \boldsymbol{\theta}^{(q)}) pl(\boldsymbol{x})}.$$

  M-step: Maximize $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)})$ with respect to $\boldsymbol{\theta}$.

- E- and M-steps are iterated until the increase of $\log L(\boldsymbol{\theta}; m)$ becomes smaller than some threshold.

Properties

1. When m is categorical: $m(A) = 1$ for some $A \subseteq \Omega$, then the previous algorithm reduces to the EM algorithm $\rightarrow$ evidential EM (E^{2}M) algorithm.
2. Monotonicity: any sequence $L(\theta^{(q)}; m)$ for $q = 0, 1, 2, \ldots$ of generalized likelihood values obtained using the E^{2}M algorithm is non decreasing, i.e., it verifies
$$L(\theta^{(q+1)}; m) \geq L(\theta^{(q)}; m), \quad \forall q.$$
3. The algorithm only uses the contour function pl, which drastically reduces the complexity of calculations.

Outline

Model and data

- Let us assume that the complete data $\boldsymbol{x} = (x_1, \ldots, x_n)$ is a realization from an i.i.d. sample $X_1, \ldots, X_n$ from $\mathcal{B}(\theta)$ with $\theta \in [0, 1]$.
- We only have partial information about the x_i's in the form: $pl_1, \ldots, pl_n$, where $pl_i(x)$ is the plausibility that $x_i = x$, $x \in \{0, 1\}$.
- Under the cognitive independence assumption:

$$
\begin{aligned}
\log L(\theta; pl_1, \ldots, pl_n) &= \sum_{i=1}^{n} \log \mathbb{E}_{\boldsymbol{\theta}} \left[pl_i(X_i) \right] \\
&= \sum_{i=1}^{n} \log \left[(1-\theta) pl_i(0) + \theta pl_i(1) \right]
\end{aligned}
$$

E- and M-steps

Complete data log-likelihood:

$$\log L(\theta, \boldsymbol{x}) = n \log(1-\theta) + \log\left(\frac{\theta}{1-\theta}\right) \sum_{i=1}^{n} x_i.$$

E-step: compute

$$Q(\theta, \theta^{(q)}) = n \log(1-\theta) + \log\left(\frac{\theta}{1-\theta}\right) \sum_{i=1}^{n} \xi_i^{(q)}, \text{ with}$$

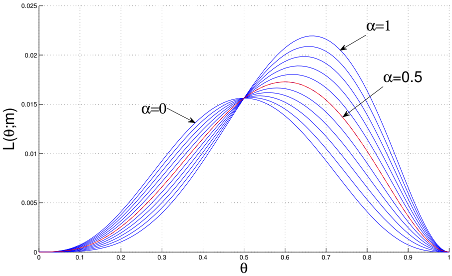
$$\xi_i^{(q)} = \mathbb{E}_{\theta^{(q)}}[X_i | pl_i] = \frac{\theta^{(q)} pl_i(1)}{(1-\theta^{(q)}) pl_i(0) + \theta^{(q)} pl_i(1)}.$$

M-step:

$$\theta^{(q+1)} = \frac{1}{n} \sum_{i=1}^{n} \xi_i^{(q)}.$$

Numerical example

i	1	2	3	4	5	6
$pl_i(0)$	1	1	1	α	0	0
$pl_i(1)$	0	0	0	$1-\alpha$	1	1



$\alpha = 0.5$

q	$\theta^{(q)}$	$L(\theta^{(q)}; pl)$
0	0.3000	6.6150
1	0.5500	16.8455
2	0.5917	17.2676
3	0.5986	17.2797
4	0.5998	17.2800
5	0.6000	17.2800

$\widehat{\theta} = 0.6$

Outline

Outline

Problem statement

- We consider a population of objects partitioned in g classes.
- Each object is described by d continuous features $\boldsymbol{W} = (W^1, \ldots, W^d)$ and a class variable Z.
- The goal of discriminant analysis is to learn a decision rule that classifies any object from its feature vector, based on a learning set.

Learning tasks

- Classically, different learning tasks are considered:
  Supervised learning: $\mathcal{L}_s = \{(\boldsymbol{w}_i, z_i)\}_{i=1}^{n}$;
  Unsupervised learning: $\mathcal{L}_{ns} = \{\boldsymbol{w}_i\}_{i=1}^{n}$;
  Semi-supervised learning: $\mathcal{L}_{ss} = \{(\boldsymbol{w}_i, z_i)\}_{i=1}^{n_s} \cup \{\boldsymbol{w}_i\}_{i=n_s}^{n}$
- Here, we consider partially supervised learning:

$$\mathcal{L}_{ps} = \{(\boldsymbol{w}_i, m_i)\}_{i=1}^{n},$$

  where m_i is a mass function representing partial information about the class of object i.
- This problem can be solved using the E^{2}M algorithm using a suitable parametric model.

Linear discriminant analysis

- Generative model:
  - Complete data: $\boldsymbol{x} = \{(\boldsymbol{w}_i, z_i)\}_{i=1}^n$, assumed to be a realization of an iid random sample $\boldsymbol{X} = \{(\boldsymbol{W}_i, Z_i)\}_{i=1}^n$;
  - Given $Z_i = k$, $\boldsymbol{W}_i$ is multivariate normal with mean $\boldsymbol{\mu}_k$ and common variance matrix Σ.
  - The proportion of class k in the population is π_k.
  - Parameter vector: $\boldsymbol{\theta} = \left(\{\pi_k\}_{k=1}^g, \{\boldsymbol{\mu}_k\}_{k=1}^g, \Sigma\right)$.
- The Bayes rule is approximated by assigning each object to the class k^* that maximizes the estimated posterior probability

$$p(Z = k|\boldsymbol{w}; \widehat{\boldsymbol{\theta}}) = \frac{\phi(\boldsymbol{w}; \widehat{\boldsymbol{\mu}}_k, \widehat{\Sigma})\widehat{\pi}_k}{\sum_\ell \phi(\boldsymbol{w}; \widehat{\boldsymbol{\mu}}_\ell, \widehat{\Sigma})\widehat{\pi}_\ell},$$

where $\widehat{\boldsymbol{\theta}}$ is the MLE of $\boldsymbol{\theta}$.

Observed-data likelihood

- In partially supervised learning, the observed-data log-likelihood has the following expression:

$$\log L(\boldsymbol{\theta}; \mathcal{L}_{ps}) = \sum_{i=1}^{n} \log \left(\sum_{k=1}^{g} pl_{ik} \pi_k \phi(\boldsymbol{w}_i; \boldsymbol{\mu}_k, \Sigma_k) \right),$$

where pl_{ik} is the plausibility that object i belongs to class k.

- Supervised learning is recovered as a special case when:

$$pl_{ik} = z_{ik} = \begin{cases} 1 & \text{if object } i \text{ belongs to class } k; \\ 0 & \text{otherwise.} \end{cases}$$

- Unsupervised learning is recovered when $pl_{ik} = 1$ for all i and k.

E^{2}M algorithm

E-step: Using $p_{\boldsymbol{X}}(\cdot; \boldsymbol{\theta}^{(q)}) \oplus m$, compute

$$t_{ik}^{(q)} = \mathbb{E}(Z_{ik} | m; \boldsymbol{\theta}^{(q)}) = \frac{\pi_k^{(q)} pl_{ik} \phi(\boldsymbol{w}_i; \boldsymbol{\mu}_k^{(q)}, \Sigma^{(q)})}{\sum_\ell \pi_k^{(q)} pl_{i\ell} \phi(\boldsymbol{w}_i; \boldsymbol{\mu}_\ell^{(q)}, \Sigma^{(q)})}$$

M-step: Update parameter estimates

$$\pi_k^{(q+1)} = \frac{1}{n} \sum_{i=1}^n t_{ik}^{(q)}, \qquad \boldsymbol{\mu}_k^{(q+1)} = \frac{\sum_{i=1}^n t_{ik}^{(q)} \boldsymbol{w}_i}{\sum_{i=1}^n t_{ik}^{(q)}}.$$

$$\Sigma^{(q+1)} = \frac{1}{n} \sum_{i,k} t_{ik}^{(q)} (\boldsymbol{w}_i - \boldsymbol{\mu}_k^{(q+1)})(\boldsymbol{w}_i - \boldsymbol{\mu}_k^{(q+1)})'$$

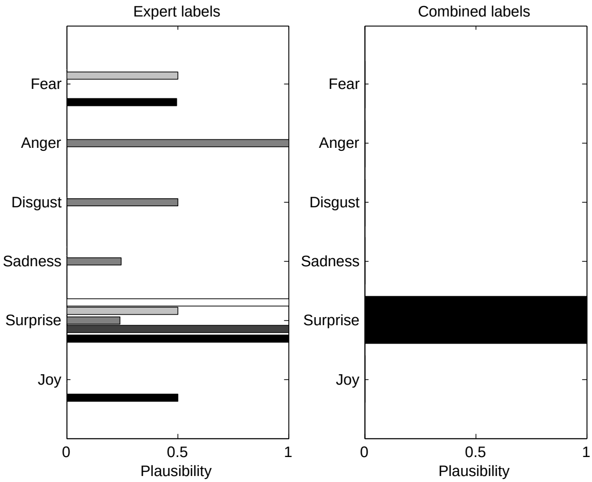
Face recognition problem

Experimental settings

- 216 images of 60 $\times$ 70 pixels, 36 in each class.
- One half for training, the rest for testing.
- A reduced number of features was extracted using Principal component analysis (PCA).
- Each training image was labeled by 5 subjects who gave degrees of plausibility for each image and each class.
- The plausibilities were combined using Dempster's rule (after some discounting to avoid total conflict).

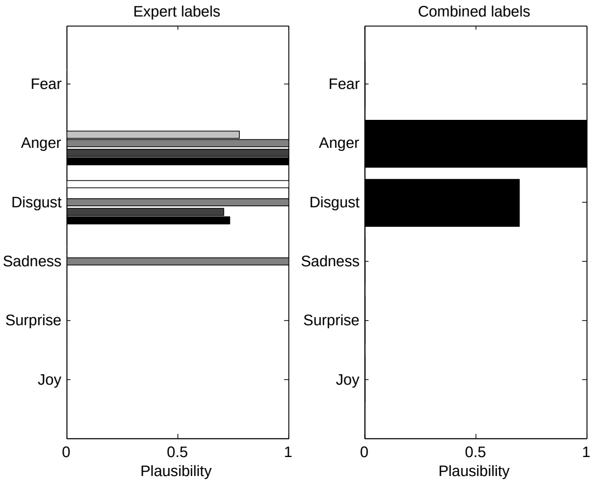
Combined labels

Example 1



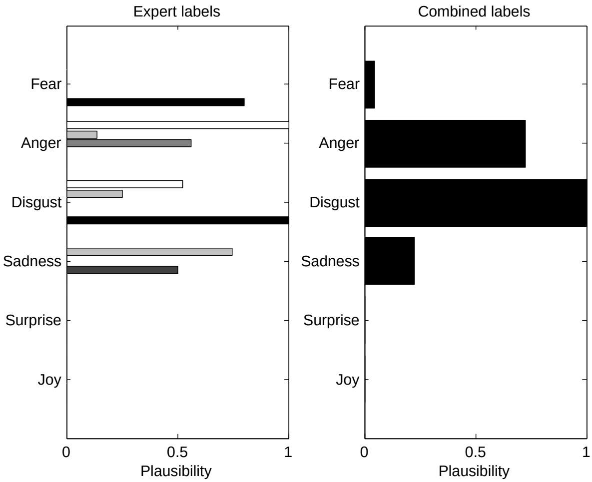
Combined labels

Example 2

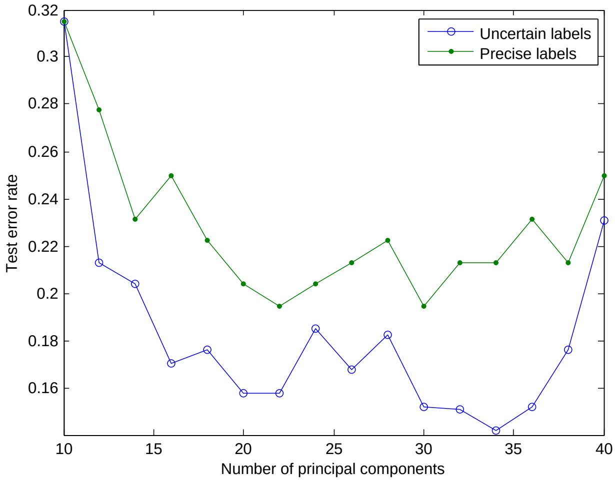


Combined labels

Example 3

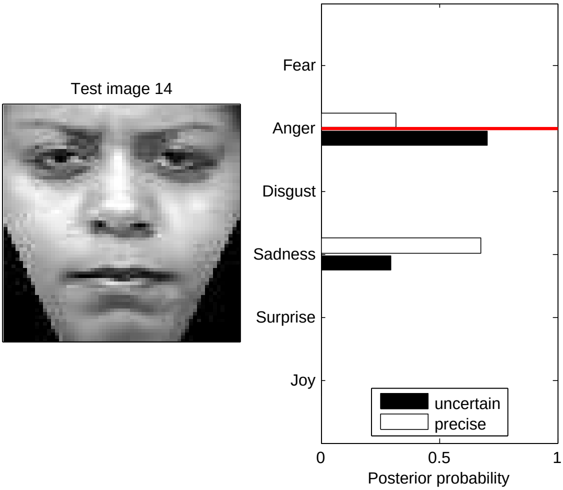


Results


Uncertain labels
Precise labels
Test error rate
0.32
0.3
0.28
0.26
0.24
0.22
0.2
0.18
0.16
10
15
20
25
30
35
40
Number of principal components

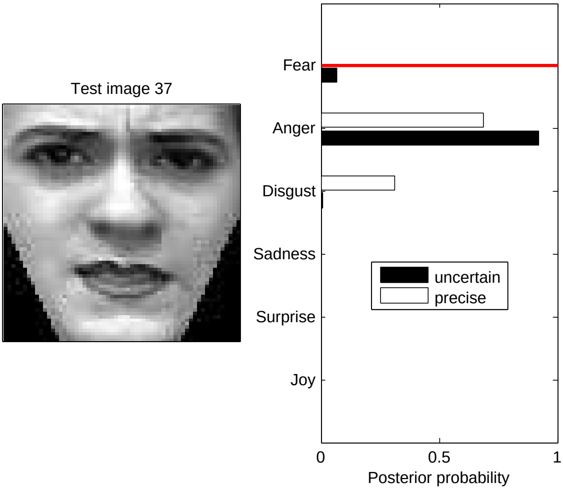
Results

Example 1



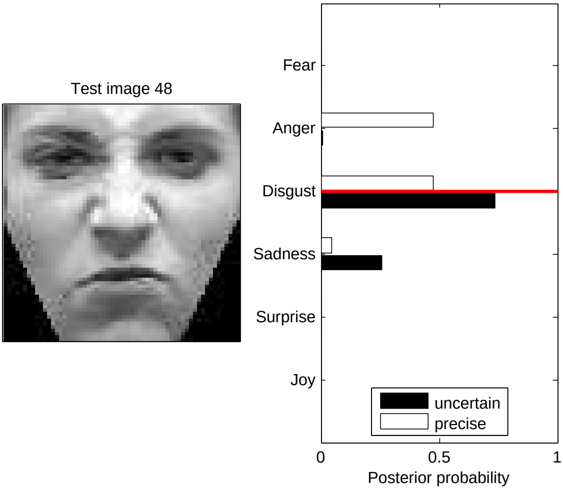
Results

Example 2



Results

Example 3



Outline

Model and data

- The complete data is assumed to be a realization $\boldsymbol{y}$ of an n-dimensional Gaussian random vector $\boldsymbol{Y} \sim \mathcal{N}(\boldsymbol{X}\beta, \sigma^2 I_n)$, where
  - $\boldsymbol{X}$ is a fixed design matrix of size (n, p),
  - I_n is the identity matrix of size n, and
  - $\boldsymbol{\theta} = (\beta, \sigma)^T$ is the parameter vector.
- We further assume that the realizations y_i of the dependent variables are ill-known and described by contour functions pl_i.
- Under the cognitive independence assumption, the joint contour function with respect to $\boldsymbol{y}$ is

$$pl(\boldsymbol{y}) = \prod_{i=1}^{n} pl_i(y_i)$$

Observed and complete-data likelihoods

- The complete data likelihood is

$$L(\boldsymbol{\theta}; y) = \phi(\boldsymbol{y}; \boldsymbol{X}\beta, \sigma^2 I_n) = \prod_{i=1}^{n} \phi(y_i; \boldsymbol{x}_i^T \beta, \sigma^2),$$

where $\boldsymbol{x}_i$ is the vector of input variables for the i-th observation.

- The observed data likelihood is

$$L(\boldsymbol{\theta}; pl) = \int \phi(\boldsymbol{y}; \boldsymbol{X}\beta, \sigma^2 I_n) pl(\boldsymbol{y}) d\boldsymbol{y}$$
$$= \prod_{i=1}^{n} \int \phi(y_i; \boldsymbol{x}_i^T \beta, \sigma^2) pl_i(y_i) dy_i$$

Evidential EM algorithm

- E-step: Taking the expectation of $\log L(\boldsymbol{\theta}; \boldsymbol{Y})$ with respect to $p_{\boldsymbol{Y}}(\cdot; \boldsymbol{\theta}) \oplus pl$ and using the fit $\boldsymbol{\theta}^{(q)}$ of $\boldsymbol{\theta}$ we get

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)}) = -n \log \sigma - \frac{1}{2\sigma^2} \left(\sum_{i=1}^{n} \gamma_i^{(q)} - 2\boldsymbol{\beta}^T \boldsymbol{X}^T \boldsymbol{\xi}^{(q)} + \boldsymbol{\beta}^T \boldsymbol{X}^T \boldsymbol{X} \beta \right) + C,$$

where $\boldsymbol{\xi}^{(q)} = \mathbb{E}_{\boldsymbol{\theta}^{(q)}}(\boldsymbol{Y}|pl)$ and $\gamma_i^{(q)} = \mathbb{E}_{\boldsymbol{\theta}^{(q)}}(Y_i^2|pl_i)$ denote, respectively, the expectations of $\boldsymbol{Y}$ and Y_i^2 with respect to $p_{\boldsymbol{Y}}(\cdot; \boldsymbol{\theta}) \oplus pl$ using the fit $\boldsymbol{\theta}^{(q)}$ of $\boldsymbol{\theta}$.

- M-step: differentiating $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)})$ with respect to β and σ, we get

$$\boldsymbol{\beta}^{(q+1)} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{\xi}^{(q)}$$

$$\sigma^{(q+1)} = \sqrt{\frac{1}{n} \left(\sum_{i=1}^{n} \gamma_i^{(q)} - 2\, \beta^{(q+1)T} \boldsymbol{X}^T \boldsymbol{\xi}^{(q)} + \boldsymbol{\beta}^{(q+1)T} \boldsymbol{X}^T \boldsymbol{X} \boldsymbol{\beta}^{(q+1)} \right)}$$

Case of Gaussian fuzzy numbers

- When the contour functions are normalized Gaussians of the form

$$pl_i(y) = \phi(y; m_i, s_i) s_i \sqrt{2\pi},$$

$p_{\boldsymbol{Y}}(\cdot; \boldsymbol{\theta}) \oplus pl$ is then Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ with $\mu = (\mu_1, \ldots, \mu_n)^T$ and $\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_n)$, where

$$\mu_i = \frac{\boldsymbol{x}_i^T \beta s_i^2 + m_i \sigma^2}{s_i^2 + \sigma^2}$$

and

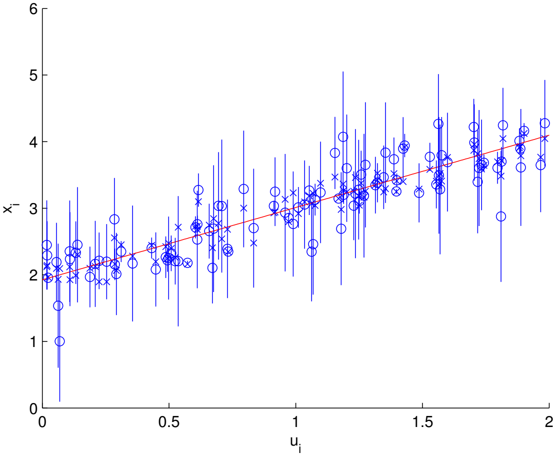
$$\sigma_i = \frac{s_i^2 \sigma^2}{s_i^2 + \sigma^2}.$$

- More complex formula can be found for the case where the contour functiuns are triangular or trapezoidal (see Denoeux, 2011).

Numerical experiment

- To demonstrate the interest of expressing partial information about ill-known data in the form of possibility distributions, we performed the following experiment.
- We generated $n = 100$ values x_i from the uniform distribution in $[0, 2]$, and we generated corresponding values y_i using the linear regression model with $\beta = (2, 1)^T$ and $\sigma = 0.2$.
- To model the situation where only partial knowledge of values $y_1, \ldots, y_n$ is available, contour functions $pl_1, \ldots, pl_n$ were generated as follows:
  - For each i, a "guess" y_i' was randomly generated from a normal distribution with mean y_i and standard deviation σ_i, were σ_i was drawn randomly from a uniform distribution in $[0, 0.5]$;
  - pl_i was defined as the triangular possibility distribution with core y_i' and support $[y_i' - 2\sigma_i, y_i' + 2\sigma_i]$.

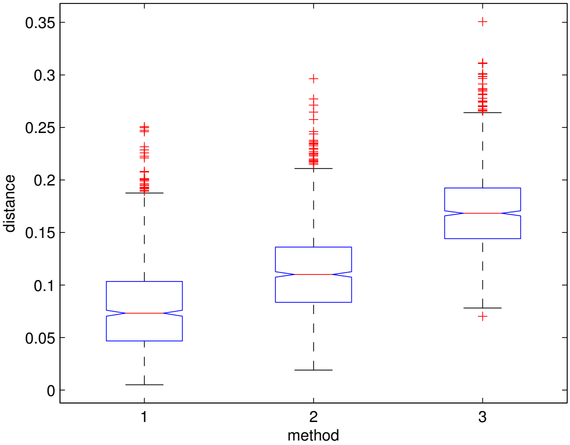
Example of a generated dataset



Numerical experiment (continued)

- Three strategies were compared for estimating the parameter vector $\boldsymbol{\theta} = (\boldsymbol{\beta}, \sigma)^T$:
  1. Using the fuzzy data $pl_1, \ldots, pl_n$ (method 1)
  2. Using only 0.5-cuts of the fuzzy data (method 2)
  3. Using only the crisp guesses $y'_1, \ldots, y'_n$(method 3)
- For each of these three methods, the L_2 distance $\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}\|$ between the true parameter vector and its MLE was computed.
- The whole experiment was repeated 1000 times.

Result



Summary

- The formalism of belief functions provides a very general setting for representing uncertain, ill-known data.
- Maximizing the proposed generalized likelihood criterion amounts to minimizing the conflict between the data and the parametric model.
- This can be achieved using an iterative algorithm (evidential EM algorithm) that reduces to the standard EM algorithm in special cases.
- In classification, the method makes it possible to handle uncertainty on class labels (partially supervised learning). Uncertainty on attributes can be handled as well.

Other applications

- The E^2M algorithm can be applied to any problem involving a parametric statistical model and epistemic uncertainty on observations, e.g.:
  - Independent factor analysis (Cherfi et al., 2011);
  - Clustering of fuzzy data using Gaussian mixture models (Quost and Denoeux, 2016);
  - Hidden Markov models (Ramasso and Denoeux, 2014).
- Open problem: How to elicit subjective evaluations in the Dempster-Shafer framework?

References I

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

T. Denœux.
Maximum likelihood estimation from fuzzy data using the EM algorithm. *Fuzzy Sets and Systems*, 183:72-91, 2011.

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

Z. L. Cherfi, L. Oukhellou, E. Côme, T. Denoeux and P. Aknin.
Partially supervised Independent Factor Analysis using soft labels elicited from multiple experts: Application to railway track circuit diagnosis. *Soft Computing*, Vol. 16, Number 5, pages 741-754, 2012.

References II

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

E. Ramasso and T. Denoeux.
Making use of partial knowledge about hidden states in HMMs: an approach based on belief functions. *IEEE Transactions on Fuzzy Systems*, Vol. 22, Issue 2, pages 395-405, 2014.

B. Quost and T. Denoeux.
Clustering and classification of fuzzy data using the fuzzy EM algorithm. *Fuzzy Sets and Systems*, Vol. 286, pages 134-156, 2016.