Statistics and Machine Learning using belief functions

Lecture 4 – Estimation from Uncertain Data

Thierry Denœux

Université de Technologie de Compiègne
HEUDIASYC (UMR CNRS 6599)

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

Beijing University of Technology
May 2017

Outline

Outline

Uncertain data

- Uncertain data arise in many applications (but it is usually neglected).
- Uncertainty 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.
- How to carry out statistical analysis of uncertain data?

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.
- Application: partially supervised learning.

Outline

1. Introduction
   - Motivations
   - Examples
2. Evidential EM algorithm
   - Evidential Likelihood
   - E^2M algorithm
3. Partially supervised classification
   - Linear discriminant analysis
   - Logistic regression
   - Results

Facial expressions

joy

surprise

sadness

disgust

anger

fear

Recognition of facial expressions

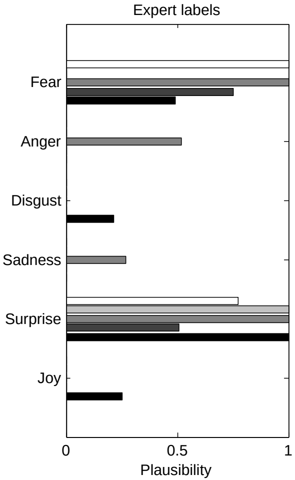
Experiment

- 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.

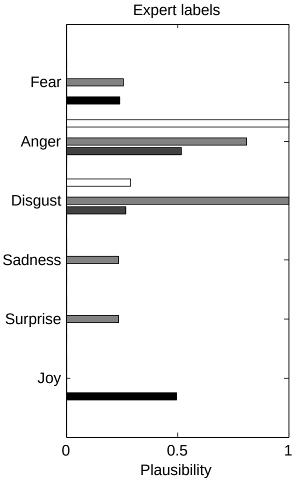
Example 1


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

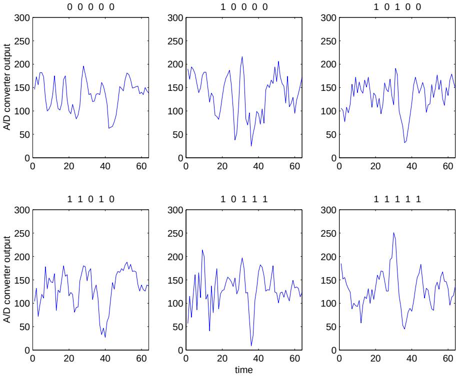
Example 3


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

Detection of K-complexes in EEG signals

- K-complexes: EEG waveforms that occur during stage 2 of non-rapid eye movement sleep . May aid sleep-based memory consolidation.
- Goal: build a system that automatically detects K-complexes in EEG data.
- We need a learning set of EEG signals labeled as positive and negative instances.
- Problem: no ground truth!
- Solution: label data by experts.

K-complex dataset



Each learning instance is composed of a feature vector and an uncertain class label.

Partially supervised learning

- 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.
- Observed data:

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

  where m_i is a mass function representing partial information about the class of object i.
- 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.

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 $\boldsymbol{\theta}$.

Outline

Likelihood function (reminder)

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

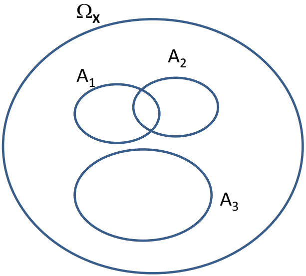
$$\boldsymbol{\theta} \rightarrow L(\boldsymbol{\theta};\boldsymbol{x}) = p_{\boldsymbol{X}}(\boldsymbol{x};\boldsymbol{\theta}).$$

- 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:

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

Evidential 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:

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

Interpretation

- We have

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

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

Case of fuzzy data

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

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

- 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.
- $L(\boldsymbol{\theta}; m)$ 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

Evidential EM algorithm

- The evidential 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^2M) algorithm.
2. Monotonicity: any sequence $L(\theta^{(q)}; m)$ for $q = 0, 1, 2, \ldots$ of evidential likelihood values obtained using the E^2M 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.

Example: uncertain Bernoulli sample

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

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

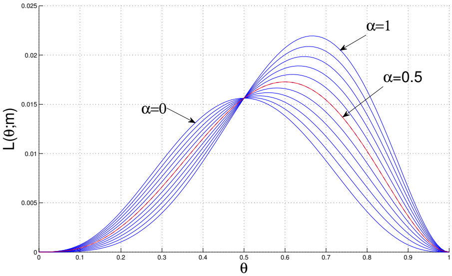
$$\xi_i^{(q)} = \mathbb{E}_{\theta^{(q)}}\left[X_i | pl_i\right] = \frac{\theta^{(q)} pl_i(1)}{(1-\theta^{(q)}) pl_i(0) + \theta^{(q)} pl_i(1)}.$$

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

Classification

- 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 classification is to learn a decision rule that classifies any object from its feature vector, based on a learning set.

Partially supervised learning

- 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.
- In this lecture, I will present two models:
  1. Linear discriminant analysis
  2. Logistic regression

Outline

Model

- 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 $\boldsymbol{\theta}$.

Complete-data likelihood

The complete-data likelihood is

$$
\begin{aligned}
L_c(\theta) &= \prod_{i=1}^{n} p(w_i | Z_i = z_i) p(z_i) && \text{(1a)} \\
&= \prod_{i=1}^{n} \prod_{k=1}^{g} \phi(w_i; \mu_k, \Sigma)^{z_{ik}} \pi_k^{z_{ik}}, && \text{(1b)}
\end{aligned}
$$

where $\phi(\cdot; \mu_k, \Sigma)$ is the multivariate normal density,

$$
\phi(w; \mu_k, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp\left\{ -\frac{1}{2} (w - \mu)^T \Sigma^{-1} (w - \mu) \right\},
$$

and z_{ik} is a binary class indicator variable, such that $z_{ik} = 1$ if $z_i = k$ and $z_{ik} = 0$ otherwise.

Observed-data likelihood

- Under the assumption of cognitive independence, the contour function on Ω_X is $pl(x) = \prod_{i=1}^n pl_i(x_i)$, with

$$pl_i(x_i) = \begin{cases} pl_{ik} & \text{if } x_i = (w_i, k) \text{ for some } k = 1, \ldots, g \\ 0 & \text{otherwise.} \end{cases}$$

- The evidential likelihood is thus

$$L(\theta) = \prod_{i=1}^{n} \sum_{k=1}^{g} pl_{ik} \phi(w_i; \mu_k, \Sigma) \pi_k. \tag{2}$$

Special cases

- When there is no uncertainty, i.e., when $pl_{ik} = z_{ik}$ for all (i,k), we have

$$\sum_{k=1}^{g} pl_{ik}\phi(w_i;\mu_k,\Sigma)\pi_k = \prod_{k=1}^{g} \phi(w_i;\mu_k,\Sigma)^{pl_{ik}}\pi_k^{pl_{ik}},$$

  and the evidential likelihood (2) becomes identical to the complete-data likelihood (1b).

- When uncertainty is maximal, i.e., class labels are completely unknown, then $pl_{ik} = 1$ for all (i,k), and the evidential likelihood (2) becomes

$$L(\theta) = \prod_{i=1}^{n}\sum_{k=1}^{g} \phi(w_i;\mu_k,\Sigma)\pi_k,$$

  which is the likelihood function corresponding to the unsupervised case.

E²M algorithm: E-step

In the E-step of the E²M algorithm for this model, we compute the expectation of the complete-data log-likelihood

$$\ell_c(\theta) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \left[\log \phi(w_i; \mu_k, \Sigma) + \log \pi_k\right]$$

with respect to the combined probability mass function

$$p_X(x|pl; \theta^{(q)}) = \prod_{i=1}^{n} p(x_i|pl_i; \theta^{(q)}),$$

with

$$p(x_i|pl_i; \theta^{(q)}) = \begin{cases} \dfrac{pl_{ik}\pi_k^{(q)}\phi(w_i; \mu_k^{(q)}, \Sigma^{(q)})}{\sum_\ell pl_{i\ell}\pi_\ell^{(q)}\phi(w_i; \mu_\ell^{(q)}, \Sigma^{(q)})} & \text{if } x_i = (w_i, k) \text{ for some } k \\ 0 & \text{otherwise.} \end{cases}$$

E²M algorithm: E-step (continued)

We get

$$Q(\theta, \theta^{(q)}) = \sum_{i=1}^{n} \sum_{k=1}^{g} t_{ik}^{(q)} \left[\log \phi(w_i; \mu_k, \Sigma)\pi_k + \log \pi_k\right], \tag{3}$$

with

$$t_{ik}^{(q)} = \mathbb{E}(Z_{ik}|pl; \theta^{(q)}) = \frac{pl_{ik}\pi_k^{(q)}\phi(w_i; \mu_k^{(q)}, \Sigma^{(q)})}{\sum_\ell pl_{i\ell}\pi_\ell^{(q)}\phi(w_i; \mu_\ell^{(q)}, \Sigma^{(q)})}. \tag{4}$$

E^{2}M algorithm: M-step

- The parameter values maximizing $Q(\theta, \theta^{(q)})$ can be readily obtained as

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

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

- The complexity is the same as that of the EM algorithm with unsupervised data and precise attributes.

Outline

Model

- In contrast with LDA, LR starts with a model of the conditional distribution of Z given $W = w$. The conditional probabilities are

$$p_k(w;\theta) = \frac{\exp(\beta_k^T \widetilde{w})}{1 + \sum_{\ell=1}^{g-1} \exp(\beta_\ell^T \widetilde{w})}, \quad k = 1, \ldots, g-1 \tag{5a}$$

$$p_g(w;\theta) = \frac{1}{1 + \sum_{\ell=1}^{g-1} \exp(\beta_\ell^T \widetilde{w})}, \tag{5b}$$

  where $p_k(w;\theta) = \mathbb{P}(Z = k | W = w;\theta)$, β_k is a $p+1$-dimensional vector of coefficients, $\theta = (\beta_1^T, \ldots, \beta_{g-1}^T)^T$ is the vector of all parameters in the model, and $\widetilde{w} = (1, w^T)^T$ is an extended input vector.
- Logistic regression maximizes the conditional likelihood

$$L_c(\theta) = \prod_{i=1}^n \mathbb{P}(Z_i = z_i | w_i;\theta) = \prod_{i=1}^n \prod_{k=1}^g p_k(w;\theta)^{z_{ik}}, \tag{6}$$

  with $p_k(w;\theta)$ equal to (5).

Evidential likelihood

- Under the cognitive independence assumption, the evidential likelihood is

$$L(\theta) = \prod_{i=1}^{n} \sum_{k=1}^{g} pl_{ik} p_k(w_i; \theta). \tag{7}$$

- We can easily check that $L(\theta) = L_c(\theta)$ whenever $pl_{ik} = z_{ik}$ for all (i, k), i.e., when there is no label uncertainty.
- On the other hand, in case of maximal uncertainty, i.e., when $pl_{ik} = 1$ for all (i, k), we have $L(\theta) = 1$ for all θ, and the model parameters can no longer be estimated.

E^{2}M algorithm: E-step

- In the E-step, we compute the expectation of the complete-data log-likelihood with respect to the combined probability mass function

$$p_Z(z|pl;\theta^{(q)}) = \prod_{i=1}^{n} p_{Z_i}(z_i|pl_i;\theta^{(q)}),$$

with

$$p_{Z_i}(k|pl_i;\theta^{(q)}) = \frac{pl_{ik}p_k(w_i;\theta^{(q)})}{\sum_\ell pl_{i\ell}p_\ell(w_i;\theta^{(q)})}, \quad k = 1,\ldots,g.$$

- We get

$$Q(\theta,\theta^{(q)}) = \sum_{i=1}^{n}\left\{\sum_{k=1}^{g-1} t_{ik}^{(q)}\beta_k^T\widetilde{w}_i - \log\left(1+\sum_{k=1}^{g-1}\beta_k^T\widetilde{w}_i\right)\right\}, \quad (8)$$

with

$$t_{ik}^{(q)} = \mathbb{E}(Z_{ik}|pl;\theta^{(q)}) = \frac{pl_{ik}p_k(w_i;\theta^{(q)})}{\sum_\ell pl_{i\ell}p_\ell(w_i;\theta^{(q)})}. \quad (9)$$

E²M algorithm: M-step

- The maximization of (8) cannot be performed in one step and requires an iterative optimization procedure, such as the Newton-Raphson algorithm.
- It is actually not necessary to maximize function $Q(\theta, \theta^{(q)})$: we may simply make a step uphill, i.e., find some new estimate $\theta^{(q+1)}$ such that $Q(\theta^{(q+1)}, \theta^{(q)}) > Q(\theta^{(q)}, \theta^{(q)})$. Such a procedure is classically called a *Generalized EM algorithm*.
- An uphill step starting from the previous estimate $\theta^{(q)}$ can be made by carrying out one iteration of the Newton-Raphson algorithm with line search, i.e., by using the following update rule,

$$\theta^{(q+1)} = \theta^{(q)} - \eta \left[\frac{\partial^2 Q(\theta, \theta^{(q)})}{\partial\theta\partial\theta^T} \right]^{-1}_{\theta=\theta^{(q)}} \left. \frac{\partial Q(\theta, \theta^{(q)})}{\partial\theta} \right|_{\theta=\theta^{(q)}},$$

where η is the step size.

Outline

Sleep data

- 1178 EEG signals encoded as 64-dimensional patterns.
- Each example (positive or negative) was then assigned a soft label consisting of a Bayesian mass function m_i such that

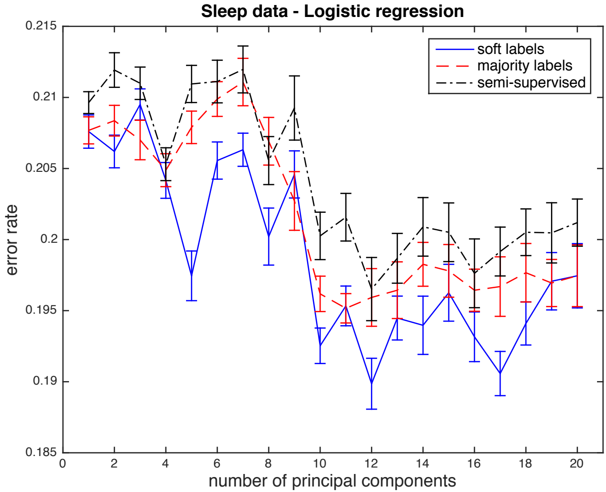
$$m_i(\{1\}) = k_i/5, \quad m_i(\{0\}) = 1 - k_i/5, \tag{10}$$

where 1 and 0 represent, respectively, the positive (K-complex) and negative (delta wave) class, and k_i denotes the number of experts who classified the pattern as positive.

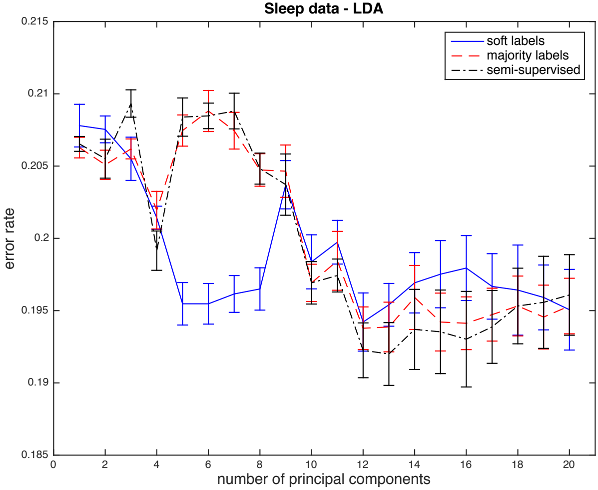
Methodology

- Both LR and LDA were applied to these data. To reduce the input dimension, Principal Component Analysis (PCA) was first used as a preprocessing step, and the number of components was varied between 1 and 20.
- The LR and LDA classifiers were trained using three different sets of labels:
  1. Soft labels (10), taking into account the proportion of experts in favor of each class;
  2. Crisp labels, corresponding to the majority decision;
  3. "Semi-supervised labels": instances classified as positive by two or three experts were considered as ambiguous and were labeled by the vacuous mass function $m_?$; the other instances were labeled unambiguously according to the majority class.
- We used 10-fold cross-validation, repeated 10 times with different random partitions. The mean cross-validation error rates with corresponding 95% confidence intervals are represented as functions of the number of principal components in the following figures.

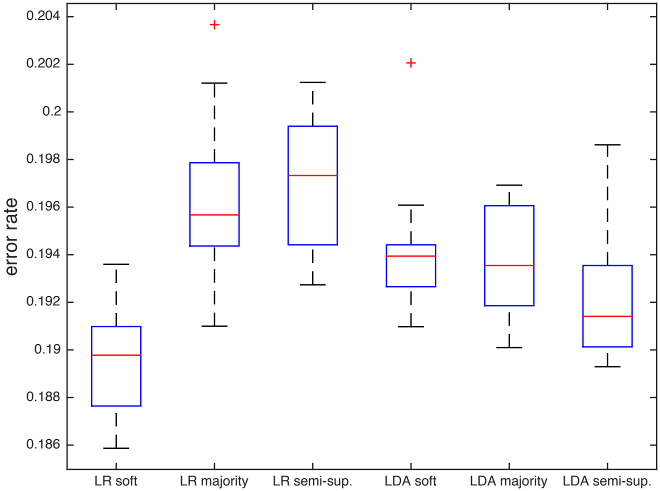
Results: logistic regression



Results: LDA


Sleep data - LDA
soft labels
majority labels
semi-supervised
error rate
number of principal components
0.215
0.21
0.205
0.2
0.195
0.19
0.185
0
2
4
6
8
10
12
14
16
18
20

Results: comparison



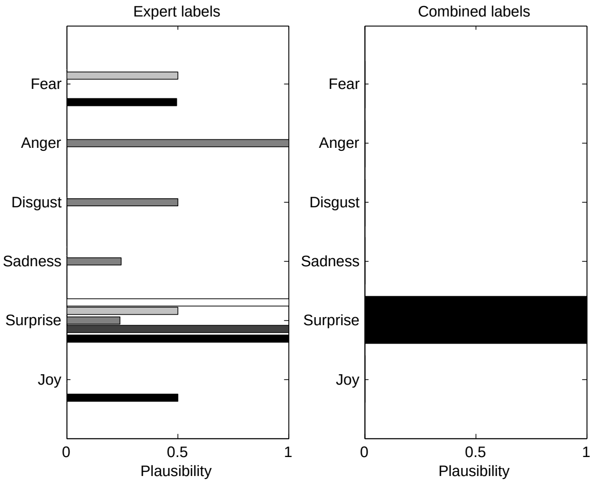
Expression recognition problem

Experimental settings

- 216 images of 60×70 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).

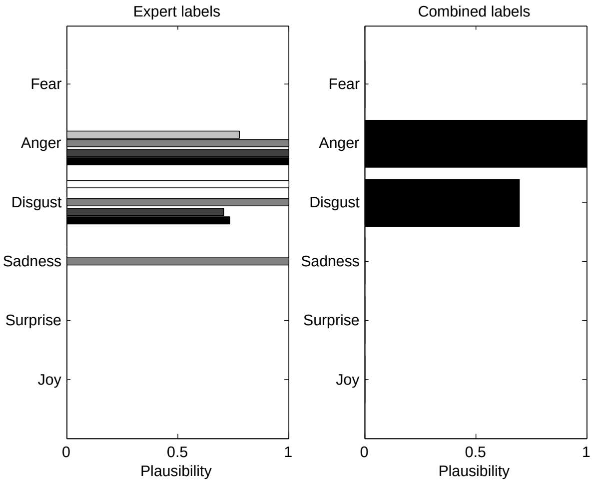
Combined labels

Example 1



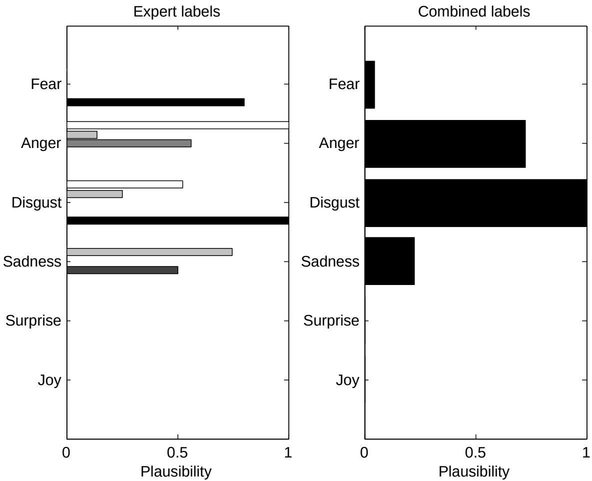
Combined labels

Example 2

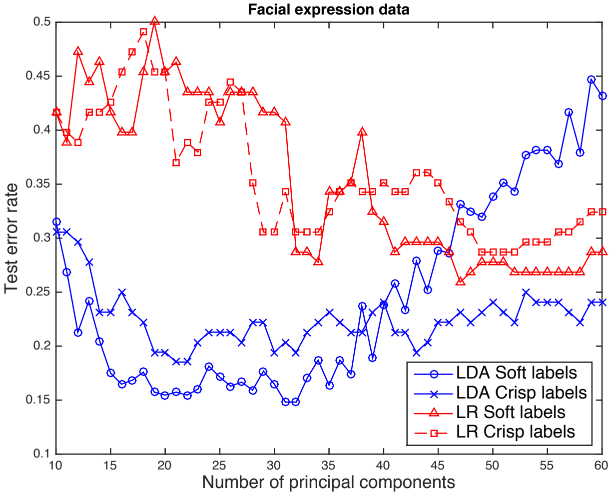


Combined labels

Example 3

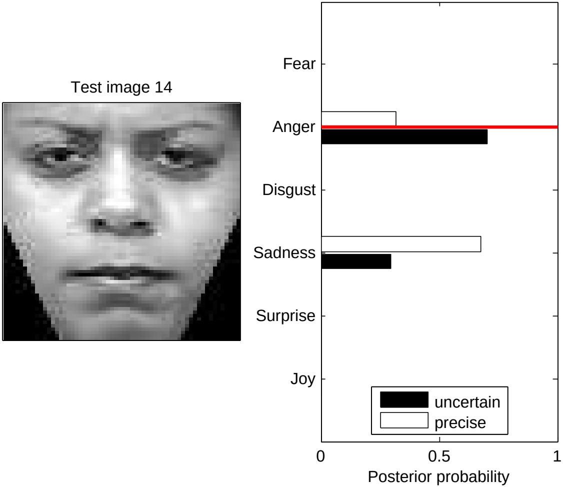


Results


Facial expression data
Test error rate
Number of principal components
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
10
15
20
25
30
35
40
45
50
55
60
LDA Soft labels
LDA Crisp labels
LR Soft labels
LR Crisp labels

Results

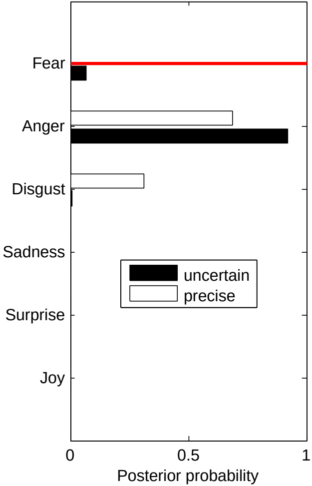
Example 1



Results

Example 2

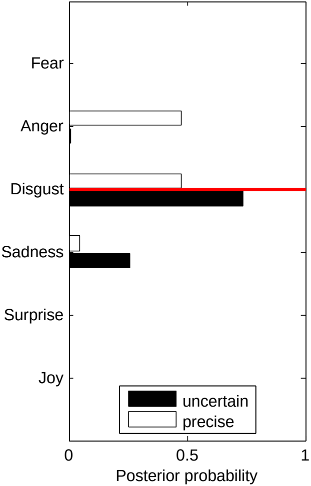
Test image 37



Results

Example 3

Test image 48



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.

B. Quost, T. Denoeux and S. Li.
Parametric classification with soft labels using the Evidential EM algorithm.
Advances in Data Analysis and Classification, submitted, 2017.