Computational Statistics and Data Analysis 71 (2014) 832–848

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Estimating mutual information for feature selection in the presence of label noise

Benoît Frénay *, Gauthier Doquire, Michel Verleysen

Machine Learning Group, ICTEAM Institute, Université catholique de Louvain, Place du Levant 3, BE 1348, Louvain-la-Neuve, Belgium

ARTICLE INFO

Article history:
Received 4 April 2012
Received in revised form 1 May 2013
Accepted 1 May 2013
Available online 9 May 2013

Keywords:
Label noise
Mutual information
Entropy estimation
Feature selection

ABSTRACT

A way to achieve feature selection for classification problems polluted by label noise is proposed. The performances of traditional feature selection algorithms often decrease sharply when some samples are wrongly labelled. A method based on a probabilistic label noise model combined with a nearest neighbours-based entropy estimator is introduced to robustly evaluate the mutual information, a popular relevance criterion for feature selection. A backward greedy search procedure is used in combination with this criterion to find relevant sets of features. Experiments establish that (i) there is a real need to take a possible label noise into account when selecting features and (ii) the proposed methodology is effectively able to reduce the negative impact of the mislabelled data points on the feature selection process.

1. Introduction

Performing feature selection is an essential preprocessing step for many data mining and pattern recognition applications, including classification (Guyon and Elisseeff, 2003; Dash and Liu, 1997). The objective is to determine, among the original set of features of a data set, which are the most relevant ones to achieve a particular task. In practice, the benefits of feature selection are numerous. First, it helps reducing the dimensionality of a data set. This aspect is particularly important when the data are high-dimensional. Indeed, learning in this context is a hard task, due to many difficulties known under the generic term *curse of dimensionality* (Bellman, 1961). In addition, it is likely that for a specific problem, some features are either irrelevant or redundant. Discarding these features generally improves the performances of classification models. Last, feature selection has the advantage over other dimensionality reduction strategies, such as feature extraction (Guyon et al., 2006), that it preserves the original features. This is of crucial importance in many industrial and medical applications, where the interpretation of the models is important.

Among the different possible solutions, filter methods are often preferred to achieve feature selection. Filters are based on the optimisation of a criterion which is independent of any prediction model; in practice, this makes them particularly fast compared to wrapper methods, which directly optimise the performances of a specific prediction model. Moreover, filter methods can be used in combination with any prediction model; for these reasons, they will be considered in this work. As a criterion of relevance, Shannon's mutual information (MI) (Shannon, 1948) is one of the most popular and successful choices for filter-based feature selection. Due to several reasons described in Section 2.1, MI possesses many required qualities for this task and has strong advantages over other well-known criteria such as the correlation coefficient.

The major problem when using MI is that, in general, it cannot be computed analytically but has to be estimated from the available data. Even if estimating MI has been intensively studied for one-dimensional features, estimating the MI between high-dimensional groups of features still remains a challenging task; however, it can prove to be very useful in practice

* Correspondence to: ICTEAM/ELEN, Université catholique de Louvain, Place du Levant 3, BE 1348, Louvain-la-Neuve, Belgium. Tel.: +32 10 478133; fax: +32 10 472598.
E-mail address: benoit.frenay@uclouvain.be (B. Frénay).

0167-9473/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.csda.2013.05.001

for feature selection. Recent works have addressed this problem, by showing the interest of a nearest-neighbours based MI estimator (Kraskov et al., 2004; Gómez-Verdejo et al., 2009).

Even if feature selection for traditional classification problems has been widely studied in the literature, it is somehow surprising that the impact of label noise on this task has not been investigated yet. To our knowledge, problems with feature selection were only mentioned by Zhang et al. (2006) and Shanab et al. (2012). In the particular context of gene selection, they show that only a few mislabelled samples cause a large percentage of the most discriminative genes to be not identified and that label noise decreases the stability of feature rankings. It is quite common when working with real-world datasets that some of the class labels are wrong (Brodley and Friedl, 1999). This can be due to the fact that, for many applications, human expertise is needed to assign class labels. Moreover, some errors can be made when labels are encoded in a data set. As label noise is known to have a negative impact on the performances of supervised classification algorithms, it is reasonable to assume that it will also degrade the performances of supervised feature selection algorithms. In this case, a label noise-tolerant feature selection algorithm would undoubtedly be of great interest.

First, the impact of label noise on a traditional MI-based filter feature selection algorithm is analysed, which shows how the performances of such an algorithm can decrease when the label noise increases. A solution to make a nearest neighbours based entropy estimator less sensitive to errors in the class labels is then proposed; the solution is based on a statistical model of the label noise and an expectation-maximisation algorithm.

The rest of the paper is organised as follows. Section 2 briefly reviews basic notions about MI-based feature selection and about the label noise problem; the impact of label noise on feature selection is also illustrated. Section 3 introduces a label noise-tolerant entropy estimator, assuming the true class memberships are known. An expectation-maximisation algorithm to estimate these memberships is derived in Section 4. The complete label noise-tolerant feature selection procedure is introduced in Section 5 and its interest is experimentally illustrated in Section 6. Section 7 concludes the paper.

2. Imprecise labels and feature selection

This section reviews basic concepts about mutual information (MI)-based feature selection and methods to handle label noise. The impact of the label noise on the performances of a classical MI-based supervised feature selection algorithm is eventually illustrated in an example.

2.1. Mutual information: definitions and interest for feature selection

Filter-based feature selection requires the use of a statistical criterion, measuring the relevance of a feature set for predicting the class labels. In this work, the mutual information (MI) (Shannon, 1948) criterion is considered. Let X denote a (group of) real-valued random variable(s) on domain $\mathcal{X}$ and Y a discrete random variable on domain $\mathcal{Y}$. In a feature selection context, X is a (group of) feature(s) and Y the associated class label. The MI between X and Y is defined as

$$I(X;Y) = H(X) - H(X|Y), \tag{1}$$

where $H(X)$ is called the entropy of X. The entropy is

$$H(X) = -\int_{\mathcal{X}} p_X(x) \log p_X(x) dx, \tag{2}$$

p_X being the probability density function of X. In Eq. (1), $H(X|Y)$ is the conditional entropy of X given Y:

$$H(X|Y) = \sum_{y \in \mathcal{Y}} p_Y(y) H(X|Y = y), \tag{3}$$

where p_Y is the probability mass function of Y. In the last equation, $H(X|Y = y)$ is the classical entropy of X, but limited to the points whose class label is y.

The MI criterion has many desirable properties for feature selection. First it has a natural interpretation in terms of uncertainty reduction. Indeed, it is symmetric and Eq. (1) can be equivalently rewritten as

$$I(X;Y) = H(Y) - H(Y|X). \tag{4}$$

Since the entropy measures the uncertainty on the observed values of a random variable, the MI can be seen as the reduction of uncertainty on the class labels once a (group of) feature(s) is known. This is obviously a sound criterion to assess the interest of a subset of features. Moreover, the MI has the advantage over other well-known criteria (such as the popular correlation coefficient, see e.g. Yu and Liu (2003)) that it is able to detect non-linear relationships between variables; it is thus more powerful in practice. Eventually, the MI can be naturally defined for multidimensional variables, which again is not the case for other popular criteria. This property can be particularly helpful for feature selection, since some features are often only relevant or redundant when considered together.

2.2. Search procedures

The objective of the feature selection method that is considered in this paper is to find the subset of the original features which together maximise the MI with the output Y. The most straightforward strategy is to try all possible feature subsets. However, such an exhaustive search is intractable in practice as the number of features gets large.

An efficient alternative often encountered in the literature is to use greedy procedures, whose most popular ones are the forward and the backward searches (Caruana and Freitag, 1994). A forward search begins with an empty set of features. Then, at each step, the feature whose addition to the set of already selected features leads to the highest MI with the output is selected. On the contrary, a backward search starts with all features. At each step of the procedure, the feature whose removal leads to the subset with the highest MI with the output vector is eliminated.

2.3. Mutual information estimation

As can be understood from Eq. (3), the MI cannot be directly computed for real-world problems, as the probability density function (PDF) of X is not known in practice. The MI has thus to be estimated from the dataset. Traditional approaches to MI estimation first start by estimating the PDF, in order to get an approximation of the entropy of X. Eqs. (1) and (3) can then be used to estimate the MI. Estimating a PDF for one-dimensional variables is a widely studied task for which many satisfactory solutions exist, e.g. the kernel-based density estimation (Steuer et al., 2002), the B-splines approach (Daub et al., 2004) or even the basic histogram (Battiti, 1994). When the dimension of the data increases, however, these methods are likely to fail, since they are strongly affected by the curse of dimensionality (Bellman, 1961). As a consequence, they cannot be used in combination with the multivariate greedy search procedures described above.

A possible alternative is to consider nearest neighbours based entropy estimators, such as the one proposed by Kozachenko and Leonenko (1987). Indeed, such estimators are expected to be less sensitive to the dimensionality of the data (Rossi et al., 2006). For this reason, the results in Kozachenko and Leonenko (1987) have been extended to the estimation of the MI for both continuous (Kraskov et al., 2004) and categorical (Gómez-Verdejo et al., 2009) output vectors. Feature selection results obtained through these estimators are particularly encouraging (see e.g. Rossi et al., 2006). The entropy estimator proposed in Kozachenko and Leonenko (1987) is

$$\hat{H}(X) = -\psi(k) + \psi(n) + \log c_d + \frac{d}{n}\sum_{i=1}^{n} \log \epsilon_k(i), \tag{5}$$

where k, the only parameter of the estimator, is the number of nearest neighbours considered, n is the total number of samples of X, d is the dimensionality of the samples (or equivalently the number of features), $c_d = \frac{2\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2})}$ is the volume of the unitary ball of dimension d (where Γ is the Gamma function) and $\epsilon_k(i)$ is twice the distance from the ith sample in X to its kth nearest neighbour (in terms of the Euclidean distance). Notice that $\epsilon_k(i)$ can also be interpreted as the diameter of the hypersphere containing the k-nearest neighbours of the ith sample. Eventually, ψ is the digamma function.

Combining Eqs. (1), (3) and (5), Gómez-Verdejo et al. (2009) derived the estimator

$$\begin{aligned}\hat{I}(X;Y) &= \hat{H}(X) - \sum_{y\in\mathcal{Y}} p_Y(y)\hat{H}(X|Y=y) \\ &= \psi(n) - \frac{1}{n}\sum_{y\in\mathcal{Y}} n_y\psi(n_y) + \frac{d}{n}\left[\sum_{i=1}^{n}\log\epsilon_k(i) - \sum_{y\in\mathcal{Y}}\sum_{i|y_i=y}\log\epsilon_k(i|y)\right],\end{aligned} \tag{6}$$

with n_y being the number of samples whose observed label is y and $\epsilon_k(i|y)$ being defined similarly as $\epsilon_k(i)$ but considering only the points belonging to class y. Eq. (6) assumes that $p_Y(y)$ can be adequately estimated by $\frac{n_y}{n}$.

2.4. Label noise

In the literature, three main approaches can be distinguished to deal with label noise in the context of classification problems.

First, some authors have proposed to filter the noisy data, in order to detect the wrongly labelled samples. Those samples are then removed or their label is corrected. Different criteria indicating the existence of mislabelled data points can be thought of. As a few examples, Guyon et al. (1996) considers the information gain, while Barandela and Gasca (2000) rather makes use of the labels of the neighbouring samples. Eventually, in Brodley and Friedl (1999), the authors use disagreement in ensemble methods. In these examples, the *decontamination* of the data set is made prior to any further classification step.

Second, a quite different strategy is the model-based approach, where an explicit label noise model is considered. The first work to propose such a strategy was Lawrence and Schölkopf (2001), where a probabilistic noise model is combined with a Fisher Kernel discriminant to perform binary classification. This model has been extended by getting rid of the limiting assumption of Gaussian distribution (Li et al., 2007) and adapted to multi-class problems (Bootkrajang and Kaban, 2011). Other models have also been proposed as e.g. Paulino et al. (2005) and Bouveyron and Girard (2009).

Eventually, different but related works address problems where uncertainties over labels are assumed to be readily available (Côme et al., 2008, 2009).

This work focuses on model-based approaches and develops a noise-tolerant MI estimator to achieve feature selection. Indeed, model-based approaches are theoretically sound and have the advantage of not discarding any sample containing potentially valuable information.

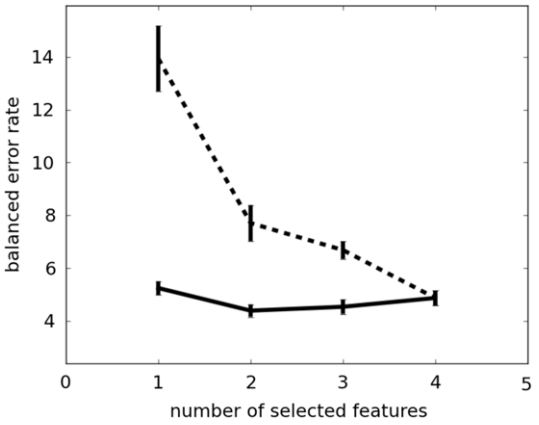


Fig. 1. Balanced classification error rate of a k-nearest neighbours classifier for the Iris dataset as a function of the number of selected features for noise-free (plain line) and noisy (dashed line) data. The error bars correspond to 95% confidence intervals.

2.5. Impact of label noise on feature selection

It is well known that label noise has a negative impact on the performances of supervised classification models. Nevertheless, there is to the best of our knowledge no evidence that label noise also degrades the performances of feature selection algorithms, except Zhang et al. (2006) which shows in the context of gene selection that only a few mislabelled samples cause a large percentage of the most discriminative genes to be not identified and Shanab et al. (2012) which shows that label noise decreases the stability of feature rankings. Notice that gene selection is particular, since there are only a few tens of training samples, what may make this application especially sensitive to label noise. The goal of this section is to show that the performances of feature selection are actually degraded by label noise and that taking label noise into account when performing feature selection can be important and useful in practice.

To this end, a greedy backward feature selection algorithm based on the MI criterion, estimated as detailed in Section 2.3, was run on the well-known Iris dataset from the UCI repository (Frank and Asuncion, 2010). The same experiment was also carried out, but after 20% of the class labels have randomly been switched to another class, chosen equiprobable among the other classes. To measure the quality of feature selection, the performances of a k-nearest neighbours (kNN) classifier are observed for each obtained feature subset, since the kNN classifier is known to be very sensitive to the presence or irrelevant features. It is important to notice that only the feature selection is made with noisy data, while the training, the validation and the prediction steps of the kNN are made using the noise-free data. This allows us to compare the results on the actual problem of interest, feature selection. The experiment was repeated 100 times and samples are split into training and test sets (70%–30%). The optimal value of the meta-parameter k was chosen using ten-fold cross-validation. All the technical details are discussed in Section 6.

Fig. 1 shows that even for the simple Iris dataset, the performances of the kNN classifier are considerably affected by the presence of label noise. Indeed, the balanced classification error rate (the average of the classification error rates obtained for each class) is largely and significantly higher in the noisy case, whatever the number of selected features is. Moreover, the stability of the feature selection is also degraded by the presence of label noise, as shown by the much larger confidence intervals.

3. Label noise-tolerant entropy estimation

As discussed in the previous section, feature selection in classification can be affected by label noise. Indeed, the Gomez MI estimator (Gómez-Verdejo et al., 2009) uses the observed labels, which may be incorrect. In turn, the incorrect MI values disrupt feature selection, which may lead to the selection of less informative feature subsets. This section proposes a label noise-tolerant entropy estimator. Since the next three sections are highly interdependent, a short introduction is given to help the reader.

3.1. Short summary of the three next sections

In this section and the two following ones, a methodology is proposed to deal with label noise for feature selection in classification. Concepts and algorithms are progressively introduced in these three sections to make developments easier to follow.

Section 3 shows that the Kozachenko–Leonenko estimator is affected by label noise. Consequently, a label noise-tolerant estimator of the entropy is proposed by relaxing the observed labels. This estimator requires the knowledge of the true memberships of each instance to the different classes, which are of course usually unknown in practice but can be estimated

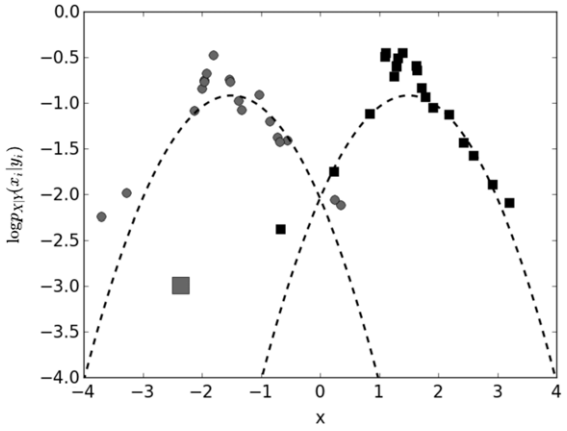


Fig. 2. Estimates of the logarithm of the conditional probability for a binary classification problem. Each class has a Gaussian distribution (dashed lines) and 40 samples are shown (grey circles belong to class 0, black squares belong to class 1); one sample is mislabelled (large grey square).

as seen in Section 4. This allows to estimate mutual information in a label noise-tolerant way to perform feature selection, as seen in Section 5.

The true memberships which are necessary in Section 3 are obtained in Section 4. A statistical model of label noise introduced in Lawrence and Schölkopf (2001) is used, whose parameters are optimised using a new expectation-maximisation algorithm. At each iteration of this EM algorithm, the class memberships are estimated using the current label noise model parameters, which are in turn updated. The resulting values can be used to evaluate the entropy estimator proposed in Section 3.

Eventually, a backward search algorithm is proposed in Section 5 to perform feature selection in the presence of label noise. This feature selection algorithm uses the results of Section 4 to estimate the class memberships and the results of Section 3 to estimate mutual information.

3.2. Effects of label noise on the Kozachenko–Leonenko estimator

As explained in Section 2, the MI between the features X and the target Y can estimated for classification problems using

$$\hat{I}(X;Y) = \hat{H}(X) - \sum_{y\in\mathcal{Y}} \hat{p}_Y(y)\hat{H}(X|Y=y) \tag{7}$$

where probabilities $\hat{p}_Y(y)$ are estimated from data. One has to estimate the entropy $H(X)$ of the features and their partial conditional entropy $H(X|Y=y)$ for each class $y \in \mathcal{Y}$. In Gómez-Verdejo et al. (2009), the Kozachenko–Leonenko estimator of entropy (5) is used to obtain the Gomez estimator of MI for classification (6), which implements Eq. (7). When instances are mislabelled, two different, yet related problems occur with both Eq. (7) and the Gomez estimator (6).

Firstly, since each term in the sum of Eq. (7) requires an empirical estimator of the form

$$\hat{H}(X|Y=y) = -\frac{1}{n_y}\sum_{i|y_i=y} \log \hat{p}_{X|Y}(x_i|y), \tag{8}$$

where the sum is taken on all instances with observed label y, the instances with incorrect labels are used to estimate the wrong quantity. For example, if x belongs to class 0 but is labelled in class 1, it is used to estimate $H(X|Y=1)$ instead of $H(X|Y=0)$. In that case, both estimates are altered.

Secondly, and consequently, the Kozachenko–Leonenko estimates of the partial conditional entropies $H(X|Y=y)$ are biased. Indeed, since the label noise results in the removal and addition of neighbours with label y, it can modify the diameter $\epsilon_k(i|y)$ of the hypersphere containing the k nearest neighbours of x_i with label y. Depending on the altered labels, the diameter can increase or decrease. Moreover, and more importantly, the diameter of the hypersphere may get large for mislabelled instances. Indeed, if x_i is incorrectly labelled in class 1 while it is surrounded by instances from its true class 0, the k nearest neighbours of x_i in class 1 will probably be far away. In such a case, the estimate of $p_{X|Y}(x_i|y)$ is almost zero and a large negative value occurs in Eq. (8).

The problem of large negative values is illustrated in Fig. 2. In that experiment, 40 samples are generated from two classes with Gaussian conditional probability distributions $\mathcal{N}(\mu_0 = -1.5, \sigma_0 = 1)$ and $\mathcal{N}(\mu_1 = 1.5, \sigma_1 = 1)$ and identical priors $p_Y(0) = p_Y(1) = \frac{1}{2}$. Moreover, one of the samples with label 0 is incorrectly assigned the label 1. Fig. 2 shows estimates of the $\log p_{X|Y}(x_i|y_i)$ terms in Eq. (8), which are computed using the Kozachenko–Leonenko approach with $k = 8$. The values reflect the conditional probability of each sample, except for the only mislabelled sample which corresponds to a large negative value. Consequently, the resulting estimate of MI is only $\hat{I}(X;Y) = 0.58$. By way of comparison, the estimate for

a clean version of this dataset (with no mislabelling) is $\hat{I}(X;Y) = 0.63$. Hence, a single mislabelled instance can already influence the estimation of MI, even for a simple problem.

3.3. True class memberships and label noise modelling

In order to address the two above problems, it is proposed to associate each sample to a true class which may be different from the observed label. In other words, each observed label is a noisy copy of a true hidden label. For each instance, one can therefore estimate the membership $p_{S|X,Y}(s|x,y)$ of an instance x to the true class s, if the observed label is y. If there is no label noise, then one simply obtains

$$p_{S|X,Y}(s|x,y) = \begin{cases} 1 & \text{if } s = y \\ 0 & \text{if } s \neq y. \end{cases} \tag{9}$$

In other cases, it is necessary (i) to choose a model of the label noise and (ii) to estimate the most probable true class memberships, given the observed data. For example, in the situation where the classes correspond to separate clusters with no overlap, it is reasonable to assume that an instance which is obviously in the wrong cluster is mislabelled. The estimated memberships depend on the label noise model, which itself expresses hypotheses on the nature of the label noise.

The rest of this section assumes that true class memberships are available. Of course, true class memberships are usually unknown in practice. Hence, an approach is proposed to estimate these quantities in Section 4. In order to simplify mathematical notations, the following notation is introduced:

$$\gamma(s|i) = p_{S|X,Y}(s|x_i, y_i). \tag{10}$$

3.4. An entropy estimator based on the true class memberships

The Kozachenko–Leonenko estimator is based on the hypothesis that $p_{Y|X}$ remains constant in a small hypersphere with diameter $\epsilon_k(i|y)$ containing exactly the k nearest neighbours of the ith sample. Using this assumption, Kozachenko and Leonenko obtain the following estimate

$$\log \hat{p}_{X|Y}(x_i|y_i) = \psi(k) - \psi(n_y) - \log c_d - d \log \epsilon_k(i|y). \tag{11}$$

Hence, the partial conditional entropy in Eq. (8) can be estimated by

$$\hat{H}(X|Y=y) = -\psi(k) + \psi(n_y) + \log c_d + \frac{d}{n_y} \sum_{i|y_i=y} \log \epsilon_k(i|y). \tag{12}$$

However, as explained in Section 3.2, problems can occur when observed labels are polluted by label noise. Instead of $H(X|Y=y)$, one would rather prefer to estimate $H(X|S=s)$. In other words, one is interested in the entropy of X given the true class S, rather than the entropy of X given the observed label Y which is potentially incorrect. In this paper, the proposed solution consists in using the hypersphere which contains approximately an expected number of k instances really belonging to the target class s. Since true class memberships are assumed to be available, they can be used to determine the new hypersphere diameter. For each class $s \in \mathcal{Y}$ and each sample x_i, one can pick the hypersphere which contains enough neighbours of x_i so that the sum of their memberships to class s is approximately equal to k. The resulting algorithm is shown in Algorithm 1.

For each class $s \in \mathcal{Y}$ and each sample x_i, Algorithm 1 starts with the standard hypersphere with diameter $\epsilon_k(i)$ which contains the k nearest neighbours of x_i. Notice that the observed labels are not taken into account. The initial hypersphere contains an expected number

$$\sum_{j=1}^{k} \gamma(s|i_j) \tag{13}$$

of instances of the target class s, where i_j is the index of the jth neighbour of the ith sample. For correctly labelled samples, this quantity is expected to be (i) close to k for their observed class and (ii) close to zero for other classes, except near the classification boundary where the expected number of instances for each class is proportional to the class prior. For mislabelled examples, the initial hypersphere is expected to contain almost no samples of their observed class, since they stand in a region where that class should not be observed.

Thereafter, the hypersphere diameter $\epsilon_{k,\gamma}(i|s)$ is increased until the sum of the memberships

$$\Gamma(s|i) = \sum_{j=1}^{k(s|i)} \gamma(s|i_j) \tag{14}$$

becomes at least equal to k, where $k(s|i)$ is the number of actually considered neighbours. The sum $\Gamma(s|i)$ of memberships estimates the actual number of neighbours of the ith sample which really belong to class s. The resulting hypersphere with diameter $\epsilon_{k,\gamma}(i|s)$ contains an expected number of $\Gamma(s|i) \approx k$ instances of the target class s.

Algorithm 1 Label noise-tolerant estimation of hypersphere diameters

Input: set of samples $\{x_i\}_{i\in 1\ldots n}$ and memberships $\{\gamma(s|i)\}_{i\in 1\ldots n, s\in\mathcal{Y}}$
Output: hypersphere diameters $\epsilon_{k,\gamma}(i|s)$ and memberships sums $\Gamma(s|i)$

```
for all class s ∈ 𝒴 do
  for all sample x_i do
    compute the ordering i_1 ... i_n of samples w.r.t. x_i

    k(s|i) ← k
    Γ(s|i) ← Σ_{j=1}^{k} γ(s|i_j)

    while Γ(s|i) < k do
      k(s|i) ← k(s|i) + 1
      Γ(s|i) ← Γ(s|i) + γ(s|i_{k(s|i)})
    end while

    ε_{k,γ}(i|s) ← 2 ‖x_{i_{k(s|i)}} − x_i‖_2
  end for
end for
```

The hypersphere diameter $\epsilon_{k,\gamma}(i|y)$ associated with the observed label y is expected to be much larger for mislabelled samples than for correctly labelled samples. Indeed, the hypersphere has to grow much in order to comprehend a region where samples have sufficient memberships to the class y. In contrast, the hypersphere diameter $\epsilon_{k,\gamma}(i|y)$ associated with the true class of mislabelled samples should be close to the diameter for correctly labelled samples of the same class. Hence, hypersphere diameters can be used to deal with noisy instances, as shown below.

With the new robust estimation of the hypersphere diameters, one obtains the following label noise-tolerant estimate

$$\log \hat{p}_{X|S}(x_i|s_i) = \psi\left(\Gamma(s|i)\right) - \psi\left(\Gamma(s)\right) - \log c_d - d\log \epsilon_{k,\gamma}(i|s) \tag{15}$$

where

$$\Gamma(s) = \sum_{i=1}^{n} \gamma(s|i) \tag{16}$$

can be interpreted as the expected number of samples which really belong to class s. Notice that the instance x_i itself is included in sums in both Eqs. (14) and (16), so that $0 \leq \Gamma(s|i) \leq \Gamma(s) \leq n$ for each ith instance and class s. Also, Γ is not the Gamma function used in Section 2.3 to compute the volume of the unitary ball of dimension d. In the rest of this paper, the notation Γ always refers to Eqs. (14) and (16).

Since (i) each sample x_i belongs to class s with probability $\gamma(s|i)$ and (ii) $\Gamma(s)$ is the estimated number of samples in class s, Eq. (8) becomes

$$\hat{H}(X|S=s) = -\frac{1}{\Gamma(s)} \sum_{i=1}^{n} \gamma(s|i) \log \hat{p}_{X|S}(x_i|s_i). \tag{17}$$

Using Eq. (15) together with Eq. (17), one eventually obtains the following label noise-tolerant estimate of the partial conditional entropy

$$\hat{H}(X|S=s) = -\frac{1}{\Gamma(s)} \sum_{i=1}^{n} \gamma(s|i)\psi\left(\Gamma(s|i)\right) + \psi\left(\Gamma(s)\right) + \log c_d + \frac{d}{\Gamma(s)} \sum_{i=1}^{n} \gamma(s|i) \log \epsilon_{k,\gamma}(i|s). \tag{18}$$

3.5. A label noise-tolerant estimator for mutual information

For feature selection, the actual quantity of interest is the mutual information. Using the above results, it is possible to derive a label noise-tolerant estimate of this quantity. Indeed, Eq. (18) allows us to estimate the partial conditional entropy $H(X|S=s)$ in a label noise-tolerant way for each class $s \in \mathcal{Y}$. Since the estimation of the entropy $H(X)$ is not affected by label noise, because labels are not taken into account, one can replace the Gomez estimator (6) by the following new label noise-tolerant estimator of the MI

$$\hat{I}(X;S) = \hat{H}(X) - \sum_{s\in\mathcal{S}} \hat{p}_S(s)\hat{H}(X|S=s) \tag{19}$$

where the partial conditional entropies $H(X|S=s)$ are estimated using Eq. (18) and $\hat{p}_S(s) = \Gamma(s)/n$. This new MI estimator measures the relationship between X and the true class S, whereas Eq. (7) estimates the relationship between X and the

observed label Y. Hence, Eq. (19) is more reliable for feature selection in the context of label noise, which is shown through experiments in Section 6.

4. True class memberships estimation

In the above developments, true class memberships are assumed to be known. However, in practice, it is seldom the case. Hence, this section proposes a new expectation maximisation (EM) algorithm to estimate the true class memberships.

4.1. Label noise modelling

In order to estimate the true class memberships, it is necessary to model the label noise. This paper uses the model introduced in Lawrence and Schölkopf (2001) which assumes that the observed label Y is a noisy copy of the true class S. Under the hypothesis that (i) an instance with true class s has a probability $p_e(s)$ to be mislabelled and that (ii) in that case the incorrect label is randomly chosen amongst the remaining labels $\mathcal{Y} \setminus \{s\}$, one obtains

$$p_{Y|S}(y|s) = \begin{cases} 1 - p_e(s) & \text{if } s = y \\ \dfrac{p_e(s)}{|\mathcal{Y}| - 1} & \text{if } s \neq y. \end{cases} \tag{20}$$

In the rest of this section, it is shown how to estimate the error probabilities $p_e(s)$ and the true class memberships through a probabilistic approach.

4.2. Derivation of an objective criterion for label noise estimation

Label noise tends to increase the estimate of the partial conditional entropy

$$\hat{H}(X|Y = y) = -\frac{1}{n_y} \sum_{i|y_i=y} \log \hat{p}_{X|Y}(x_i|y). \tag{21}$$

Indeed, as illustrated in Section 3.2, the most important effect of label noise is to introduce large negative values in the sum in Eq. (21). Eventually, this increase of the partial conditional entropies decreases the estimate of the MI.

The large negative values in Eq. (21) are due to the difficulty of standard models to explain misclassified instances. Indeed, such observations typically arise in regions where instances of the corresponding class should not exist. Consequently, misclassified instances are seen by standard methods as outliers that (i) bias the estimated class distributions whose tails get heavier and (ii) have a very small estimated conditional probability $\hat{p}_{X|Y}(x|y)$.

Label noise modelling allows increasing $\hat{p}_{X|Y}(x|y)$ for misclassified instances, since the label noise model can help to explain such noisy observations. Eventually, it reduces the decrease of the estimated MI value. It is proposed here to take advantage of this behaviour to define an objective criterion for selecting p_e. More precisely, one should seek the model which maximises the estimated MI, i.e.

$$\hat{p}_e = \arg\max_{p_e} \hat{I}(X; Y). \tag{22}$$

In such a case, the label noise model is used to reduce the uncertainty coming from the label noise itself. Using Eqs. (7) and (21), one can show that the above formulation is equivalent to

$$\hat{p}_e = \arg\max_{p_e} \sum_{i=1}^{n} \log \hat{p}_{X|Y}(x_i|y_i). \tag{23}$$

This formulation is similar to objective function considered in Lawrence and Schölkopf (2001). However, there are two main differences: (i) the justification of Eq. (23) is based on MI maximisation, whereas the formulation in Lawrence and Schölkopf (2001) is based on maximum likelihood and (ii) only the label noise model is optimised in our case, whereas a classification model is also optimised in Lawrence and Schölkopf (2001). Indeed, Lawrence and Schölkopf (2001) is about label noise-tolerant classification, whereas this paper is about MI estimation.

4.3. An expectation maximisation algorithm for label noise estimation

There exists no closed-form solution to Eq. (23). Indeed, one can write

$$\sum_{i=1}^{n} \log \hat{p}_{X|Y}(x_i|y_i) = \sum_{i=1}^{n} \log \sum_{s \in \mathcal{Y}} \hat{p}_{X,S|Y}(x_i, s|y_i) \tag{24}$$

where the second sum spans all possible true classes. Quantity (24) is called the incomplete log-likelihood, because the true classes are unknown. Because of the occurrence of logarithms in the sum, the function is non-convex and multiple local maxima may exist. As a closed-form expression for the global maximum of Eq. (24) does not exist, one can rather use an expectation maximisation (EM) algorithm (Dempster et al., 1977) which considers S as a latent random variable in order (i) to build successive approximations of the incomplete log-likelihood and (ii) to use them to maintain an estimate of the error probabilities $p_e(s)$. For Eq. (24), the EM algorithm alternatively (i) estimates the functional

$$Q(p_e, p_e^{\text{old}}) = \sum_{i=1}^{n} \sum_{s \in \mathcal{Y}} \gamma(s|i) \log \hat{p}_{X,S|Y}(x_i, s|y_i) \tag{25}$$

using the current estimate p_e^{old} (E step) and (ii) maximises $Q(p_e, p_e^{\text{old}})$ with respect to p_e in order to update its estimate (M step). See e.g. Sections 9.3 and 9.4 of Bishop (2007) for details about the EM algorithm and the link between Eqs. (24) and (25). At the beginning of the EM algorithm, the error probabilities $p_e(s)$ are randomly initialised and the true class priors are equal to the label priors, i.e. $p_Y(s) = \frac{n_y}{n}$.

The E step consists in estimating the true class memberships using the current label noise model, i.e.

$$\gamma(s|i) = \frac{p_{X|S}(x_i|s) p_{Y|S}(y_i|s) p_S(s)}{\sum_{s \in \mathcal{S}} p_{X|S}(x_i|s) p_{Y|S}(y_i|s) p_S(s)}. \tag{26}$$

The labelling probabilities $p_{Y|S}(y_i|s)$ are provided by Eq. (20), whereas the likelihoods $p_{X|S}(x_i|s)$ can be obtained from Eq. (15) as

$$\hat{p}_{X|S}(x_i|s_i) = \exp\big(\psi(\Gamma(s|i)) - \psi(\Gamma(s)) - \log c_d - d \log \epsilon_{k,\gamma}(i|s)\big). \tag{27}$$

During the M step, the error probabilities are updated (see Appendix) as

$$p_e(s) = \frac{1}{\Gamma(s)} \sum_{i|y_i \neq s} \gamma(s|i) \tag{28}$$

whereas the true class priors become

$$p_S(s) = \frac{1}{n} \sum_{i=1}^{n} \gamma(s|i). \tag{29}$$

Notice that EM is an iterative algorithm that may converge to local maxima. Hence, the E step and M step must be repeated until convergence and the whole EM must be repeated several times with random initial values for p_e and $p_S(s)$. The best solution is then selected by evaluating the incomplete log-likelihood with Eq. (24), which is the objective function. This quantity can also be used to stop the EM, e.g. by detecting that a (local) maximum has been reached.

4.4. Time complexity comparison of the entropy estimation algorithms

The results derived in Sections 3 and 4 allow estimating mutual information in the presence of label noise. Let us now compare the proposed method and the standard method in terms of time complexity. In order to simplify the discussion, the estimation of $H(X)$ can be ignored since it is identically performed in both methods. Similarly, both methods require an efficient data structure to sort instances, whose computational cost of creation can be ignored. In fact, the neighbours search is the more computationally demanding operation in both cases. The exact cost of the neighbour search depends on the implementation (using e.g. k-d trees introduced in Bentley, 1975; Friedman et al., 1977), but it is possible to perform a simple comparative analysis.

For the Gomez estimator (6), a kth neighbour search has to be done n times. In the proposed methodology, neighbours are searched in Algorithm 1: for each class $s \in \mathcal{Y}$ and each sample x_i, the hypersphere diameter $\epsilon_{k,\gamma}(i|s)$ is increased until $\Gamma(s|i)$ becomes at least equal to k. This process is equivalent to the kth neighbour search for a correctly labelled instance, whereas it takes in the worst case n iterations for a mislabelled instance. Hopefully, the number of mislabelled instances which fall in the *worst case* category is likely to remain small. Since Algorithm 1 is used in each E step of the EM algorithm, the average computation cost of the proposed method is $r|\mathcal{Y}|$ times larger than the cost of the Gómez estimator, where r is the number of EM iterations.

In practice, memberships stabilise after only a few iterations of the EM algorithm. For a small number of classes $|\mathcal{Y}|$ and a reasonable number of mislabelled instances, the proposed methodology is only a few times slower than the standard approach.

4.5. Illustration

Let us consider again the problem of large negative values discussed in Section 3.2 and illustrated for a simple binary classification problem in Fig. 2. With the above procedure for estimating true class memberships, Fig. 3 shows the estimates

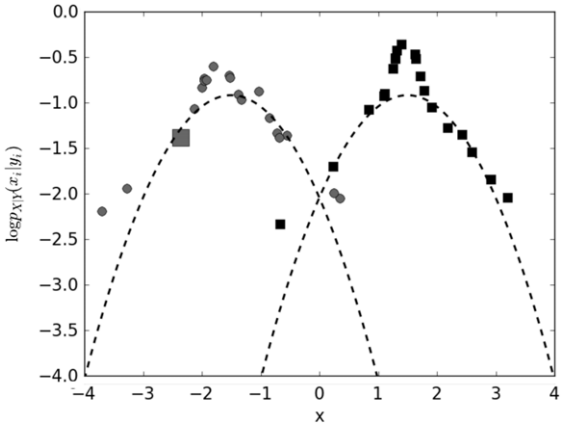


Fig. 3. Estimates of the logarithm of the conditional probability for a binary classification problem. Each class has a Gaussian distribution (dashed lines) and 40 samples are shown (grey circles belong to class 0, black squares belong to class 1), but one sample is mislabelled (large grey square).

of the terms

$$\log p_{X|Y}(x_i|y_i) = \log \sum_{s \in \mathcal{Y}} p_{X,S|Y}(x_i, s|y_i). \tag{30}$$

It can be seen that the conditional probability of the mislabelled sample has significantly increased and is now at the same level as the samples of its true class. Moreover, the estimated error probabilities are $p_e(0) = 0.02$ and $p_e(1) = 0.00$, which is close to the true error frequencies in the dataset. Eventually, the new MI estimate obtained using Eq. (19) is $\hat{I}(X; Y) = 0.61$, which is better than the MI estimate obtained by the standard estimator in Section 3.2.

5. Label noise-tolerant backward search for feature selection

In Sections 3 and 4, tools were developed for estimating MI in the presence of label noise. This section proposes a simple variant of the backward search, which is a well-known feature selection algorithm. The choice of this algorithm is motivated by the fact that backward search starts with large groups of features. In that case, the available information is higher at the beginning and both feature selection and label noise modelling should be easier. On the contrary, forward search has the drawback that it is largely influenced by the choice of the first feature. Indeed, different choices for the first feature generally lead to very different feature subsets. However, the MI between different features and the output vector is not always significantly different, which means that the choice of the first feature to be added is difficult. The backward search traditionally leads to a greater stability regarding the selected features. Notice that backward search may however not be suitable for problems with very large dimensionality, since (i) the number of necessary MI estimations grows in $\mathcal{O}(d^2)$ and (ii) the MI estimation itself becomes less reliable in such cases.

Algorithm 2 shows the proposed label noise-tolerant backward search. At each iteration, the algorithm firstly estimates a model of the label noise and, thereafter, chooses the feature to be removed. Hence, the label noise model is identical for all features within an iteration. There are two advantages to this approach, compared to estimating a new label noise model for each tested feature. Firstly, the computational load is much smaller. Indeed, $\mathcal{O}(d)$ label noise models are estimated in the former case instead of $\mathcal{O}(d^2)$ label noise models in the latter case. Secondly, the same label noise model is used to compare all features which can still be removed. Notice that $\mathcal{S}_i$ is the subset of feature indices of size $i \in 1 \ldots d$, which is obtained at the step $d - i$ of the algorithm.

6. Experiments

Let us recall the two questions raised in the introduction which are addressed in this paper: (i) what are the effects of label noise on feature selection and (ii) is it possible to reduce these effects? In Sections 2.5 and 3.2, it has been shown that label noise adversely impacts feature selection, because the MI estimation itself is affected. In Section 5, a label-noise tolerant algorithm has been derived using a new label noise-tolerant entropy estimator, which is shown in Section 4.5 to be less sensitive to label noise.

The goal of this section is not to compare the proposed feature selection method with all existing feature selection methods. Instead, this section addresses the two above questions for several real-world datasets, in order to confirm the results obtained in previous sections for simple examples. Hence, only two feature selection algorithms are compared: (i) backward search with standard MI (BW) and (ii) the proposed label noise-tolerant backward search (LNT-BW). Since both algorithms are identical, except for the evaluation of MI, it allows focusing the analysis on the two above questions about feature selection.

Algorithm 2 Label noise-tolerant backward search for feature selection

Input: set of samples $\{x_i\}_{i\in 1\dots n}$
Output: subsets of feature indices $\{\mathcal{S}_i\}_{i\in 1\dots d}$

$\mathcal{S}_d \leftarrow \{1, \dots, d\}$

for all number of features $i \in d-1\dots 1$ **do**
  estimate true class memberships γ with the EM proposed in Section 4

  for all remaining feature with index $j \in \mathcal{S}_{i+1}$ **do**
    compute $\hat{I}_j = \hat{I}(X_{\mathcal{S}_{i+1}\setminus\{j\}}; S)$ with the label noise-tolerant estimator (19)
  end for

  $\mathcal{S}_i \leftarrow \mathcal{S}_{i+1} \setminus \left\{\arg\max_j \hat{I}_j\right\}$
end for

6.1. Experimental setting

In the following experiments, the k-nearest neighbours (kNN) classifier (Kononenko and Kukar, 2007) is used to compare selected subsets of features. Given a new sample, this classifier predicts the majority class in its k nearest neighbours. Despite its simplicity, this classifier usually achieves excellent results (Cover and Hart, 1967). Moreover, it is fast and only one meta-parameter has to be tuned: the number k of neighbours. Since the kNN classifier uses all features to compute distances in order to locate nearest neighbours, it is particularly well suited to assess the two questions recalled at the beginning of this section. Indeed, it is theoretically unable to perform any embedded feature selection and is sensitive to the presence of irrelevant features, which alter distances. If an irrelevant feature is selected, pairwise distances between instances become noisy. In turn, this distance noise causes the neighbourhood of an instance to be less appropriate to predict its class and leads to a decrease in performances.

The experiments are performed on the eleven UCI datasets (Frank and Asuncion, 2010) described in Table 1, where the last column gives the following measure of class imbalance

$$\sqrt{\frac{1}{|\mathcal{Y}|}\sum_{y\in\mathcal{Y}}\left(\frac{n_y}{n}-\frac{1}{|\mathcal{Y}|}\right)^2}, \tag{31}$$

which is inspired from the measure introduced in Pinto et al. (2009). The fourth feature of the Ecoli dataset and the third feature of the Segment dataset are constant and have been removed, since they have no predictive power and can be ignored beforehand. Eq. (31) measures the difference between the observed frequency of each class and the frequency $\frac{1}{|\mathcal{Y}|}$ which corresponds to the perfectly balanced case. Balanced datasets obtain a zero value, whereas imbalanced datasets correspond to larger values. The Glass and Yeast datasets originally contain some very small classes which are not used, since they do not even consist of enough instances to estimate the standard MI. Classes {1, 2, 7} and classes {1, 2, 3, 4} were kept for the Glass and Yeast datasets, respectively. Notice that very high-dimensional datasets are not considered in our experiments, like e.g. micro-array datasets which contain only a few tens of instances with hundreds of thousands of features. In such cases, specific feature selection methods have to be used. For example, in the micro-array case, linear models provide good results (Guyon et al., 2002; Mukherjee, 2003; Carin et al., 2004; Lee et al., 2005; Xu et al., 2009) since they are about the more complex models which can be used, because of the very small number of instances. Other methods based on correlation, statistical tests or regularisation are commonly used (Guyon et al., 2002; Li and Cheng, 2004; Helleputte and Dupont, 2009; Wang et al., 2010; Hall and Xue, 2014). Investigating our method for very high-dimensional datasets is thus left for further work and could be the topic of a specific paper, due to the special characteristics of such problems.

For each dataset, all features are normalised. Then, samples are split into training and test sets (70%–30%). Training labels are then polluted by a random label noise, i.e. a given percentage of labels are flipped. For each flipped label, the new label is chosen among the other possible classes with uniform probability. The training set is used to perform feature selection in three different ways: (i) backward search with standard MI estimated on the clean labels (BW-C), (ii) backward search with standard MI estimated on the noisy labels (BW-N) and (iii) the proposed label noise-tolerant backward search (LNT-BW) with label noise-tolerant MI estimated on the noisy labels.

For each feature subset obtained by the three above feature selection algorithms, the training set with clean labels is also used to obtain a kNN classifier. This step consists in (i) selecting the optimal value of the meta-parameter k using ten-fold cross-validation and (ii) building the kNN classifier. Eventually, the performances of each kNN are assessed on the test set, which is not polluted by label noise. Both validation and test errors are measured by the balanced error rate criterion, which is the average of the percentages of misclassification for each class. This criterion avoids situations where a classifier could assign all instances to the majority class and yet obtain a good score. Indeed, some of the datasets in Table 1 are imbalanced and measures like e.g. the error rate are not appropriate, since they are relevant for balanced classification problems only.

Table 1
Detailed list of datasets used for experiments, ordered by name.

Name	Size	Dimensionality	nb. of classes	Imbalance
Ecoli	327	6	5	0.13
Glass	175	9	3	0.12
Iris	150	4	3	0.00
Page	5473	10	5	0.35
Segment	2310	18	7	0.00
Vehicle	752	18	4	0.01
Vertebral Column	310	6	3	0.12
Wall Robot	5456	24	4	0.15
Waveform	5000	40	3	0.00
Wine	178	13	3	0.05
Yeast	1296	8	4	0.10

In order to achieve statistically significant conclusions, experiments are repeated 100 times and the dataset is shuffled before each run. The mean and standard deviation of the balanced error rate are computed. The levels of label noise are 10% and 20% of flipped labels, respectively. The parameter for the MI estimator is $k = 8$, which is standard in MI estimation (Stögbauer et al., 2004). In theory, the parameter k should be optimised using e.g. cross-validation (François et al., 2007; Verleysen et al., 2009), but such methods are computationally consuming (Gómez-Verdejo et al., 2009). This could greatly limit the interest of MI estimators in filter-based feature selection. Instead, it is advised in the MI estimation literature to use small values for k (Kraskov et al., 2004; Stögbauer et al., 2004; Rossi et al., 2006), which has been shown to give reliable MI estimates (Doquire and Verleysen, 2012). Similarly, the parameter for the label noise modelling is $k = 3$, which constrains the label noise to be estimated locally. Indeed, it was noticed during preliminary experiments that minority classes tend to be integrated in majority classes with larger values of k during the estimation of the true class memberships. The tested values for the meta-parameter k of kNN classifiers are considered incrementally between 1 neighbour and 50 neighbours, with increasing step size. Between 1 and 10 neighbours, the step size is 1, whereas it is 2 and 5 between 10 and 20 neighbours and between 20 and 50 neighbours, respectively.

Notice that the Kraskov estimator suffers from numerical problems when all neighbours of a sample stand at zero distance from it. Indeed, in that case, the terms $\log \epsilon_k(i|s)$ and $\log \epsilon_{k,\gamma}(i|s)$ are infinite in Eq. (12) and Eq. (18), respectively. This situation can occur when one selects only a few features which contain many repeated values. In order to avoid this situation, a small Gaussian noise with zero mean and standard deviation $\sigma = 10^{-3}$ is added to each feature before (and only for) the feature selection step, so that zero distances no longer occur.

6.2. Results on real datasets

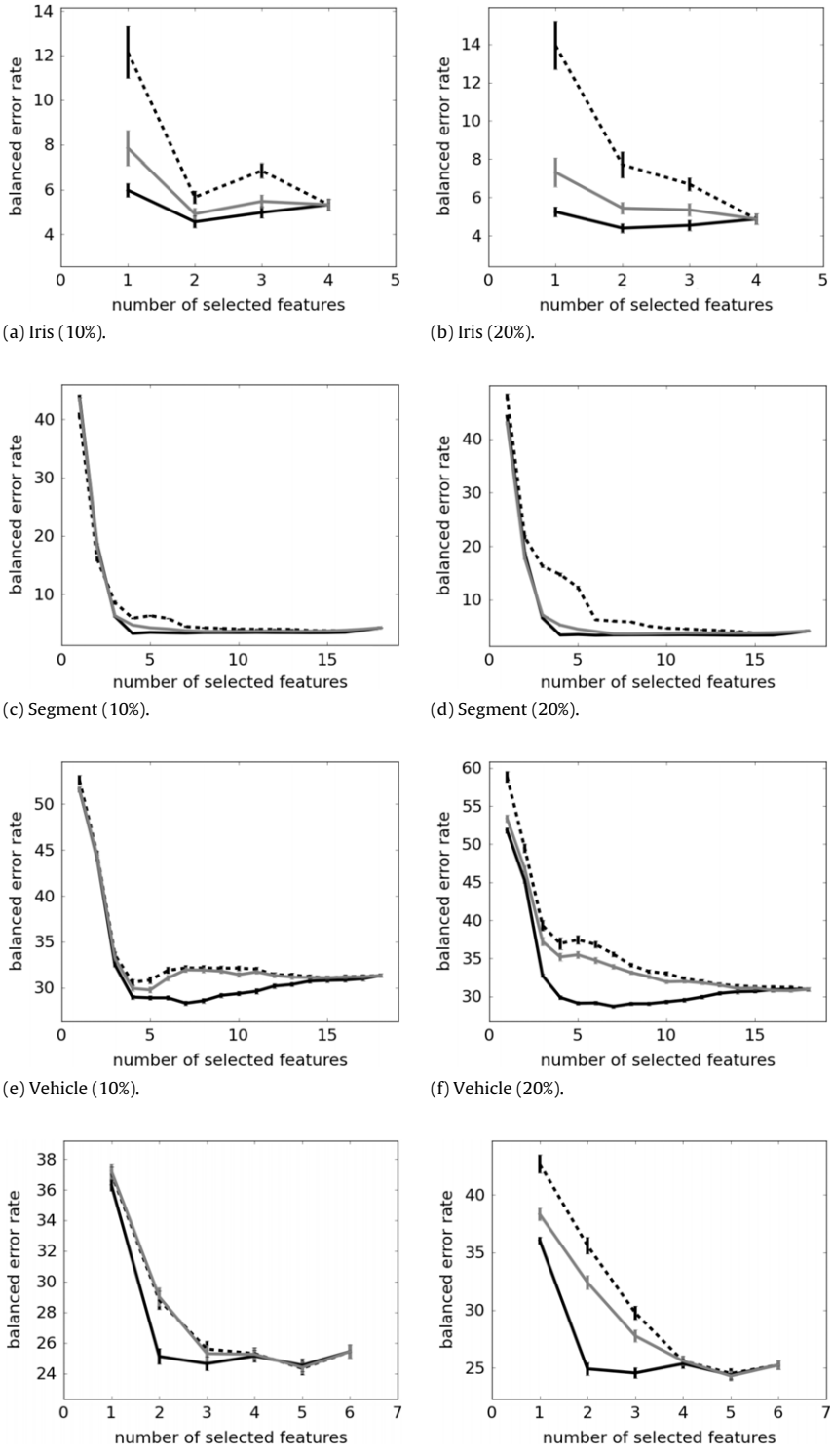
Figs. 4–6 show the results for each dataset in Table 1. In each figure, 10% of the labels have been flipped to simulate label noise in the left column, whereas this percentage increases to 20% in the right column. The curves show the mean over 100 runs of the balanced error rate in terms of the feature subset size. Error bars show the 95% confidence interval for the means. The training sets used in the case of BW-C are not polluted by artificial label noise, contrarily to the training sets for BW-N and LNT-BW. Moreover, the training sets used to build kNN classifiers and the test sets are not polluted by artificial label noise.

In order to facilitate the discussion of the experimental results, datasets are split into two groups: (i) Figs. 4 and 5 show those for which the proposed feature selection method produces better feature subsets than BW-N, whereas (ii) Fig. 6 shows those for which the balanced error rate is not significantly different between BW-N and LNT-BW.

6.3. Discussion

The results in Figs. 4–6 show that the label noise can have a significant impact on feature selection. Indeed, the performances obtained by the kNN classifier are almost always significantly worst when feature subsets are obtained using BW-N than with BW-C. This can be explained by the fact that the artificial label noise affects the MI estimation, which in turns leads the backward search in a wrong direction. The final consequence is a decrease of classification performances, which can be important in practice. In particular, large differences in balanced error rates between BW-C and BW-N are shown in Figs. 4(f), (h), 5(d) and (f). The only exceptions are the Ecoli and Yeast datasets for which the difference is either small or non-existent.

For the datasets illustrated in Figs. 4 and 5, the proposed feature selection method is able to improve the classification performances. When 20% of the labels are flipped, LNT-BW is always significantly better than BW-N, in terms of the balanced error rate. Notice that this is only true for intermediate feature subset sizes, i.e. $1 \ll |\mathcal{S}| \ll d$. Indeed, when $|\mathcal{S}| = d$, all features are being selected, whatever the feature selection algorithm, which means that their respective performances must be identical. This also explains that the performances of the three feature selection algorithms become more and more similar as the feature subset size tends towards d. When $|\mathcal{S}| = 1$, only one feature is selected and the amount of available



(a) Iris (10%). (b) Iris (20%).

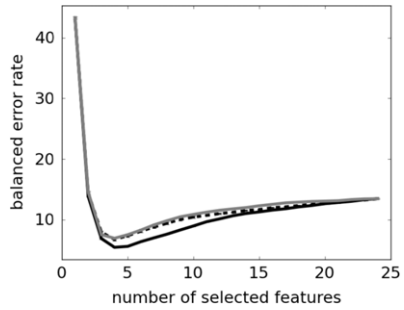
(c) Segment (10%). (d) Segment (20%).

(e) Vehicle (10%). (f) Vehicle (20%).

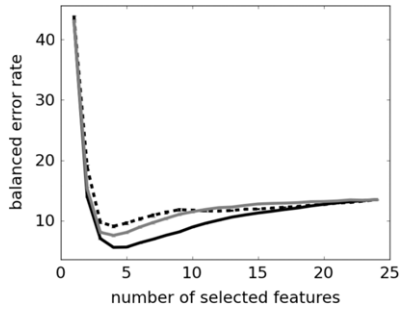
(g) Vertebral Column (10%). (h) Vertebral Column (20%).

Fig. 4. Results for the (a–b) Iris, (c–d) Segment, (e–f) Vehicle and (g–h) Vertebral Column datasets. Balanced error rates in percentages are shown in terms of the feature subset size for BW-C (plain black line), BW-N (dashed black line) and LNT-BW (plain grey line). The levels of label noise are 10% and 20% of flipped labels in the left and right columns, respectively.

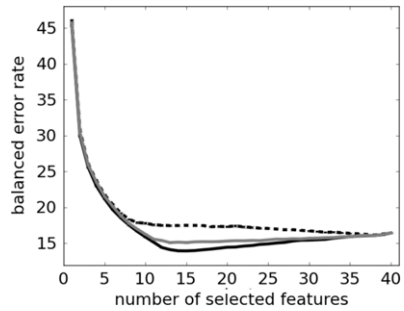
information is too low to achieve satisfying classification performances. Therefore, models achieve bad performances with any feature selection when the number of selected features becomes too small.



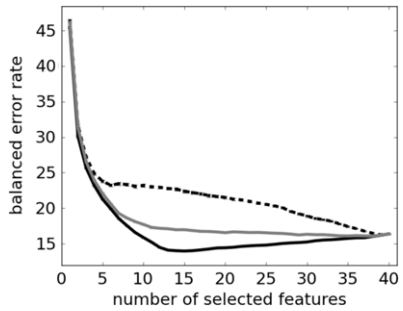
(a) Wall Robot (10%).



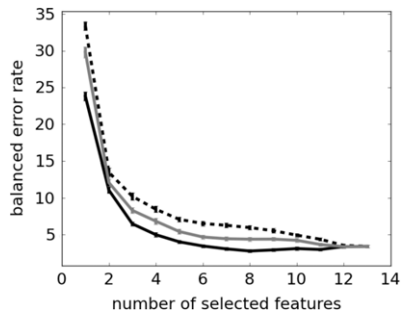
(b) Wall Robot (20%).



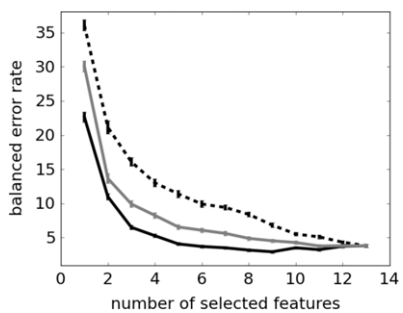
(c) Waveform (10%).



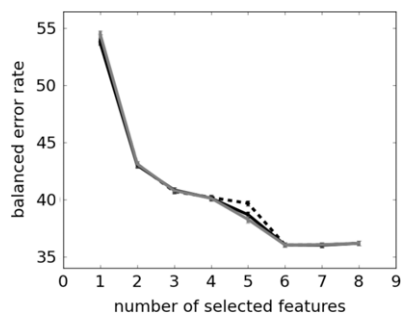
(d) Waveform (20%).



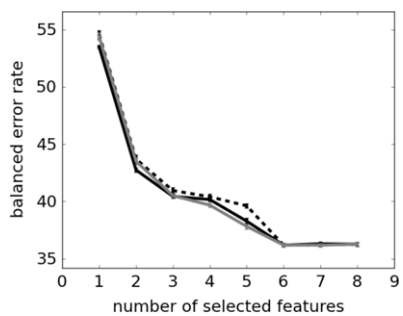
(e) Wine (10%).



(f) Wine (20%).



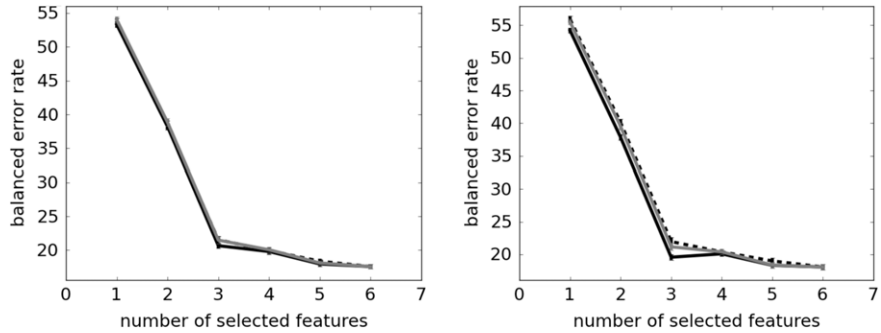
(g) Yeast (10%).



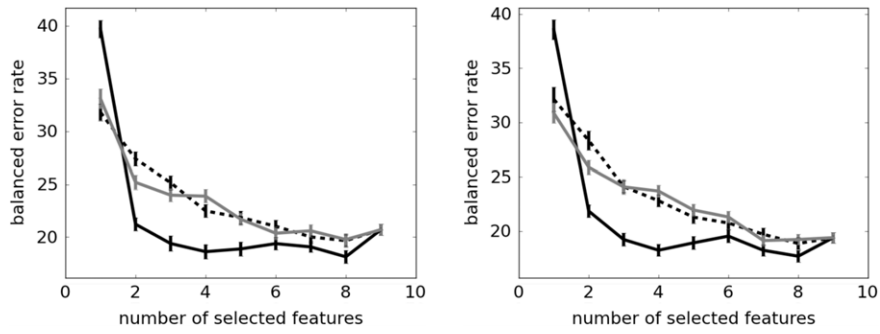
(h) Yeast (20%).

Fig. 5. Results for the (a–b) Wall Robot, (c–d) Waveform, (e–f) Wine datasets and (g–h) Yeast datasets. Balanced error rates in percentages are shown in terms of the feature subset size for BW-C (plain black line), BW-N (dashed black line) and LNT-BW (plain grey line). The levels of label noise are 10% and 20% of flipped labels in the left and right columns, respectively.

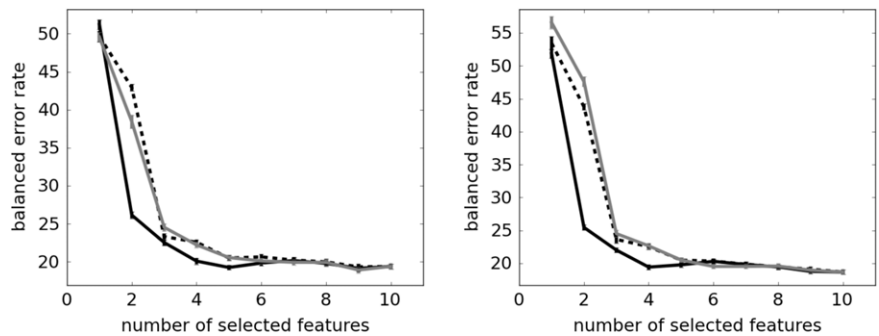
Notice that BW-C is better than LNT-BW in most cases, which means that it could be possible to further reduce the impact of label noise on feature selection. When 10% of the labels are flipped, LNT-BW remains better than BW-N, except in the case of the Vertebral Column and Wall Robot datasets.



(a) Ecoli (10%). (b) Ecoli (20%).



(c) Glass (10%). (d) Glass (20%).



(e) Page (10%). (f) Page (20%).

Fig. 6. Results for the (a–b) Ecoli, (c–d) Glass and (e–f) Page. Balanced error rates in percentages are shown in terms of the feature subset size for BW-C (plain black line), BW-N (dashed black line) and LNT-BW (plain grey line). The levels of label noise are 10% and 20% of flipped labels in the left and right columns, respectively.

For the datasets illustrated in Fig. 6, it is impossible to distinguish the performances of the LNT-BW and BW-N methods. For the Ecoli dataset, the balanced error rates are identical for the three methods. In other words, the artificial label noise has no impact on feature selection for this dataset. For the Glass and Page datasets, both LNT-BW and BW-N are affected by label noise, but neither of them is significantly better over a large range of feature subset sizes.

6.4. Summary of results

According to the above results and discussion, it is possible to answer the two questions asked in the beginning of this experimental section. Firstly, the label noise adversely impacts feature selection. By degrading the MI estimation, it causes backward search to take wrong decisions when the MI estimate does not take label noise into account. In turn, this leads to less informative feature subsets and the classification performances are eventually decreased. The resulting decrease in performances can be important. Secondly, it has been shown that the proposed approach can improve feature selection results. In most cases, it significantly improves the classification performances. At the very worst, performances are not degraded when the method does not improve the feature selection.

7. Conclusion

A new entropy estimator is proposed to be used in the context of label noise-tolerant feature selection. Indeed, experiments show that label noise often has a significant negative influence on the quality of selected features for MI-based algorithms, because the MI estimators themselves are altered by label noise. The proposed entropy estimator is used to derive a new label noise-tolerant MI estimator. In turn, it is shown how to use this estimator to perform label noise-tolerant feature selection. Experimental results show that the proposed label noise-tolerant feature selection algorithm obtains feature subsets which allow improving the performances of classification models.

Acknowledgements

Gauthier Doquire is funded by a Belgian F.R.I.A grant. Computational resources have been provided by the supercomputing facilities of the Université catholique de Louvain (CISM/UCL) and the Consortium des Equipements de Calcul Intensif en Fédération Wallonie Bruxelles (CECI) funded by the Fond de la Recherche Scientifique de Belgique (FRS-FNRS).

Appendix

Proof of the update rule for error probabilities

The update rule for error probabilities (28) can be obtained by maximising

$$Q(p_e, p_e^{\text{old}}) = \sum_{i=1}^{n} \sum_{s \in \mathcal{Y}} \gamma(s|i) \log p_{X,S|Y}(x_i, s|y_i) \tag{32}$$

with respect to $p_e(s)$ for each $s \in \mathcal{Y}$. Since

$$p_{X,S|Y}(x_i, s|y_i) = \frac{p_{X|S}(x_i|s) p_{Y|S}(y_i|s) p_S(s)}{p_Y(y_i)} \tag{33}$$

where only $p_{Y|S}(y_i|s)$ depends on $p_e(s)$, this is equivalent to maximising

$$\sum_{i=1}^{n} \sum_{s \in \mathcal{Y}} \gamma(s|i) \log p_{Y|S}(y_i|s) = \sum_{i=1}^{n} \left[\gamma(y_i|i) \log(1 - p_e(y_i)) + \sum_{s \in \mathcal{Y} \setminus \{y_i\}} \gamma(s|i) \log \frac{p_e(s)}{|\mathcal{Y}| - 1} \right]. \tag{34}$$

Setting the derivative of the above expression with respect to $p_e(s)$ to zero gives

$$- \sum_{i|y_i = s} \frac{\gamma(y_i|i)}{1 - p_e(s)} + \sum_{i|y_i \neq s} \frac{\gamma(s|i)}{p_e(s)} = 0 \tag{35}$$

which eventually gives the update rule for error probabilities

$$p_e(s) = \frac{1}{\Gamma(s)} \sum_{i|y_i \neq s} \gamma(s|i). \tag{36}$$

References

Barandela, R., Gasca, E., 2000. Decontamination of training samples for supervised pattern recognition methods. In: SSPR/SPR. In: Lecture Notes in Computer Science, vol. 1876. Springer, pp. 621–630.
Battiti, R., 1994. Using mutual information for selecting features in supervised neural net learning. IEEE Transactions on Neural Networks 5, 537–550.
Bellman, R.E., 1961. Adaptive Control Processes—A Guided Tour. Princeton University Press.
Bentley, J.L., 1975. Multidimensional binary search trees used for associative searching. Communications of the ACM 18 (9), 509–517.
Bishop, C.M., 2007. Pattern Recognition and Machine Learning, first ed. In: Information Science and Statistics, Springer.
Bootkrajang, J., Kaban, A., 2011. Multi-class classification in the presence of labelling errors. In: Proceedings of the 19th European Symposium on Artificial Neural Networks.
Bouveyron, C., Girard, S., 2009. Robust supervised classification with mixture models: learning from data with uncertain labels. Pattern Recognition 42, 2649–2658.
Brodley, C.E., Friedl, M.A., 1999. Identifying mislabeled training data. Journal of Artificial Intelligence Research 11, 131–167.
Carin, L., Krishnapuram, B., Hartemink, A., 2004. Gene expression analysis: joint feature selection and classifier design. In: Kernel Methods in Computational Biology. Bradford Books.
Caruana, R., Freitag, D., 1994. Greedy attribute selection. In: Proceedings of the Eleventh International Conference on Machine Learning. Morgan Kaufmann, pp. 28–36.
Côme, E., Oukhellou, L., Denoeux, T., Aknin, P., 2008. Mixture model estimation with soft labels. In: SMPS. In: Advances in Soft Computing, vol. 48. Springer, pp. 165–174.
Côme, E., Oukhellou, L., Denoeux, T., Aknin, P., 2009. Learning from partially supervised data using mixture models and belief functions. Pattern Recognition 42 (3), 334–348.

Cover, T., Hart, P., 1967. Nearest neighbor pattern classification. IEEE Transactions on Information Theory 13 (1), 21–27.
Dash, M., Liu, H., 1997. Feature selection for classification. Intelligent Data Analysis 1, 131–156.
Daub, C., Steuer, R., Selbig, J., Kloska, S., 2004. Estimating mutual information using b-spline functions—an improved similarity measure for analysing gene expression data. BMC Bioinformatics 5 (1), 118.
Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society. Series B (Methodological) 39 (1), 1–38.
Doquire, G., Verleysen, M., 2012. A comparison of multivariate mutual information estimators for feature selection. In: ICPRAM'12. pp. 176–185.
François, D., Rossi, F., Wertz, V., Verleysen, M., 2007. Resampling methods for parameter-free and robust feature selection with mutual information. Neurocomputing 70 (7–9), 1276–1288.
Frank, A., Asuncion, A., 2010. UCI machine learning repository. http://archive.ics.uci.edu/ml.
Friedman, J.H., Bentley, J.L., Finkel, R.A., 1977. An algorithm for finding best matches in logarithmic expected time. ACM Transactions on Mathematical Software 3 (3), 209–226.
Gómez-Verdejo, V., Verleysen, M., Fleury, J., 2009. Information-theoretic feature selection for functional data classification. Neurocomputing 72, 3580–3589.
Guyon, I., Elisseeff, A., 2003. An introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157–1182.
Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L.A., 2006. Feature Extraction: Foundations and Applications. Springer-Verlag New York, Inc.
Guyon, I., Matic, N., Vapnik, V., 1996. Discovering informative patterns and data cleaning. In: Advances in Knowledge Discovery and Data Mining. American Association for Artificial Intelligence, pp. 181–203.
Guyon, I., Weston, J., Barnhill, S., Vapnik, V., 2002. Gene selection for cancer classification using support vector machines. Machine Learning 46 (1–3), 389–422.
Hall, P., Xue, J.-H., 2014. On selecting interacting features from high-dimensional data. Computational Statistics & Data Analysis, 71, 694–708.
Helleputte, T., Dupont, P., 2009. Feature selection by transfer learning with linear regularized models. In: Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases: Part I. ECML PKDD'09. pp. 533–547.
Kononenko, I., Kukar, M., 2007. Machine Learning and Data Mining: Introduction to Principles and Algorithms. Horwood Publishing.
Kozachenko, L.F., Leonenko, N., 1987. Sample estimate of the entropy of a random vector. Problems of Information Transmission 23, 95–101.
Kraskov, A., Stögbauer, H., Grassberger, P., 2004. Estimating mutual information. Physical Review E 69 (6), 066138.
Lawrence, N.D., Schölkopf, B., 2001. Estimating a kernel Fisher discriminant in the presence of label noise. In: Proceedings of the Eighteenth International Conference on Machine Learning. ICML'01. Morgan Kaufmann Publishers Inc., pp. 306–313.
Lee, J.W., Lee, J.B., Park, M., Song, S.H., 2005. An extensive comparison of recent classification tools applied to microarray data. Computational Statistics & Data Analysis 48 (4), 869–885.
Li, C.-S., Cheng, C., 2004. Stable classification with applications to microarray data. Computational Statistics & Data Analysis 47 (3), 599–609.
Li, Y., Wessels, L.F.A., de Ridder, D., Reinders, M.J.T., 2007. Classification in the presence of class noise using a probabilistic Kernel Fisher method. Pattern Recognition 40, 3349–3357.
Mukherjee, S., 2003. Classifying microarray data using support vector machines. In: A Practical Approach to Microarray Data Analysis. Kluwer Academic Publishers.
Paulino, C.D., Silva, G., Achcar, J.A., 2005. Bayesian analysis of correlated misclassified binary data. Computational Statistics & Data Analysis 49 (4), 1120–1131.
Pinto, D., Rosso, P., Jiménez-Salazar, H., 2009. On the assessment of text corpora. In: NLDB. pp. 281–290.
Rossi, F., Lendasse, A., François, D., Wertz, V., Verleysen, M., 2006. Mutual information for the selection of relevant variables in spectrometric nonlinear modelling. Chemometrics and Intelligent Laboratory Systems 80 (2), 215–226.
Shanab, A.A., Khoshgoftaar, T.M., Wald, R., 2012. Robustness of threshold-based feature rankers with data sampling on noisy and imbalanced data. In: FLAIRS Conference.
Shannon, C.E., 1948. A mathematical theory of communication. Bell System Technical Journal 27, 379–423. 623–656.
Steuer, R., Kurths, J., Daub, C.O., Weise, J., Selbig, J., 2002. The mutual information: detecting and evaluating dependencies between variables. Bioinformatics 18 (suppl. 2), S231–S240.
Stögbauer, H., Kraskov, E., Astakhov, S.A., Grassberger, P., 2004. Least-dependent-component analysis based on mutual information. Physical Review E 70, 066123.
Verleysen, M., Rossi, F., François, D., 2009. Advances in feature selection with mutual information. In: Biehl, M., Hammer, B., Verleysen, M., Villmann, T. (Eds.), Similarity-Based Clustering. In: Lecture Notes in Computer Science, vol. 5400. Springer, Berlin, Heidelberg, pp. 52–69.
Wang, X., Park, T., Carriere, K., 2010. Variable selection via combined penalization for high-dimensional data analysis. Computational Statistics & Data Analysis 54 (10), 2230–2243.
Xu, P., Brock, G.N., Parrish, R.S., 2009. Modified linear discriminant analysis approaches for classification of high-dimensional microarray data. Computational Statistics & Data Analysis 53 (5), 1674–1687.
Yu, L., Liu, H., 2003. Feature selection for high-dimensional data: a fast correlation-based filter solution. In: ICML. pp. 856–863.
Zhang, W., Rekaya, R., Bertrand, K., 2006. A method for predicting disease subtypes in presence of misclassification among training samples using gene expression: application to human breast cancer. Bioinformatics 22, 317–325.