Computational Statistics and Data Analysis 71 (2014) 681–693

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

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

Analysis of feature selection stability on high dimension and small sample data

David Dernoncourt [a,b,*], Blaise Hanczar [c], Jean-Daniel Zucker [a,d]

[a] Institut National de la Santé et de la Recherche Médicale, U872, Nutriomique, Équipe 7, Centre de Recherches des Cordeliers, 75006, Paris, France

[b] Université Pierre et Marie-Curie - Paris 6, 75006, Paris, France

[c] LIPADE, Université Paris Descartes, 45 rue des Saint-Pères, Paris, F-75006, France

[d] Institut de Recherche pour le Développement, IRD, UMI 209, UMMISCO, France Nord, F-93143, Bondy, France

ARTICLE INFO

Article history:
Received 30 June 2012
Received in revised form 7 July 2013
Accepted 7 July 2013
Available online 18 July 2013

Keywords:
Feature selection
Small sample
Stability
Low N/D ratio

ABSTRACT

Feature selection is an important step when building a classifier on high dimensional data. As the number of observations is small, the feature selection tends to be unstable. It is common that two feature subsets, obtained from different datasets but dealing with the same classification problem, do not overlap significantly. Although it is a crucial problem, few works have been done on the selection stability. The behavior of feature selection is analyzed in various conditions, not exclusively but with a focus on t-score based feature selection approaches and small sample data. The analysis is in three steps: the first one is theoretical using a simple mathematical model; the second one is empirical and based on artificial data; and the last one is based on real data. These three analyses lead to the same results and give a better understanding of the feature selection problem in high dimension data.

1. Introduction

Classification tasks in which the number of features D is much larger than the number of samples N are an increasingly frequent problem and became recently a research area of its own (Hastie et al., 2009). For instance, in computational biology, microarray data contain the simultaneous expression of tens of thousands of genes, and metagenomic data contain in the order of a few millions of genes…usually measured on (at most) a few hundreds patients. High dimensionality and small sample size pose a challenge to classification techniques, since they both increase the risk of overfitting and decrease the accuracy of classifiers (Jain and Chandrasekaran, 1982). Moreover, high dimensionality can increase computation time beyond reasonable limits, as classifiers usually do not scale too well to huge numbers of features. To deal with these problems, feature selection is used to reduce data dimensionality.

Feature selection refers to the process of removing irrelevant or redundant features from the original set of features $\mathcal{F} = \{f_1, f_2, \ldots, f_{|\mathcal{F}|=D}\}$, so as to retain a subset $S \subset \mathcal{F}$ containing only informative features useful for classification. Feature selection methods can be broken down into three categories: filter, wrapper and embedded methods (Saeys et al., 2007). It is generally agreed that wrappers or embedded methods should be preferred if technically feasible (Pudil and Somol, 2008), however, on very high dimensional data, filters remain the method of choice for tractability reasons, which is why we will focus on them.

* Correspondence to: Centre de Recherches des Cordeliers, Équipe 7 Nutriomique, 15 rue de l'École de Médecine, 75006 Paris, France. Tel.: +33 1 44 27 80 76.

E-mail addresses: me@daviddernoncourt.com, david.dernoncourt@crc.jussieu.fr (D. Dernoncourt).

0167-9473/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.csda.2013.07.012

Beyond classification performance, the other main objective of feature selection is to obtain a reliable and robust list of predictive variables (signature). A good signature must not overfit the available data and be exportable to other datasets related to the same classification problem. These conditions cannot be respected if the subset of selected features is highly variable. A lot of examples in the literature show that in small-sample or high dimension settings, the feature selection is not stable. For instance, in Miecznikowski et al. (2010), five classification tasks dealing with a similar problem (breast cancer prognosis prediction from gene expression data) were performed on five different datasets, leading to highly variable results of the individual gene analysis. Several other studies, such as Ioannidis (2005), Michiels et al. (2005), Ein-Dor et al. (2006) and Haury et al. (2011), emphasized the difficulty to obtain a reproducible gene signature on high-dimension small-sample data. This difficulty to find a common subset of predictors between such different but similar datasets, or even between different sample subsets from a same dataset, raises the problem of feature selection stability.

Few studies have already dealt with this problem, and most of them have focused on comparing the stability of different, pre-existing or new feature selection methods, without exploring how different types of variations in the training sets affect this stability (for instance, Kalousis et al., 2005; Somol et al., 2009 and Yao and Wang, 2013). Moreover, they most often used stability measures which could be biased by the proportion of selected features (most stability measures artificially increase when the proportion of selected features increases) or by the amount of non-selected features (some stability measures take into account the stability of both selected and unselected features, so can be excessively high on datasets containing a large proportion of easy to exclude, irrelevant features). In this work, we investigate the behavior of the feature selection stability and its impact on the classifiers. We first present the main measures of selection stability used in machine learning and propose corrections of some of them that are biased. Then we present our analysis of the behavior of feature selection in three steps. In the first step, we present a theoretical analysis of the performance and stability of feature selection on a simple Gaussian model. The second step is an empirical analysis performed on a large number of simulations based on artificial data. In the last step we present results of selection stability on real data. These three analyses lead to the same conclusions: in high dimensions feature selection is not stable and the probability for relevant features to be selected can be very low.

2. Stability measures

The stability of a feature selection method was defined in Kalousis et al. (2007) as *the robustness of the feature preferences it produces to differences in training sets drawn from the same generating distribution*. To evaluate this robustness, quite a few different stability measures have already been described. We follow the taxonomy presented by Somol and Novovičová (2010), who distinguished:

- *feature-focused* versus *subset-focused* measures: the former evaluate feature selection frequencies over all feature subsets considered together as a whole, while the latter evaluate similarities within every pairs of selected feature subsets. Both types provide complementary information, so we want to have at least one of each.
- *selection-registering* versus *selection-exclusion-registering* measures: the first only considers the stability of selected features while the latter also measures the stability of excluded features. On large datasets where a huge number of features are irrelevant and easy to exclude, *selection-exclusion-registering* measures will be strongly upward biased, so we will only be interested in *selection-registering* measures here.
- *subset-size-biased* versus *subset-size-unbiased* measures: the first yield values bounded more tightly than [0; 1], with most notably the lower bound strongly increasing with the proportion of selected features, the latter are adjusted to be actually bounded by [0; 1]. Obviously, for better generalization, we want to use *subset-size-unbiased* measures.

2.1. Relative weighted consistency, an unbiased feature-focused measure

Among the stability measures sorted in the above-mentioned taxonomy, only one was both *selection-registering* and *subset-size-unbiased*: the relative weighted consistency CW_{rel} (Somol and Novovičová, 2010). It was based on a *subset-size-biased* measure, the weighted consistency CW, corrected to be actually bounded by [0; 1] no matter the proportion of selected features. A value of 0 indicates the highest possible instability, while a value of 1 indicates the highest possible stability, i.e., if all feature subsets have the same cardinality, all subsets are identical.

Let $\mathcal{S} = \{S_1, S_2, \ldots, S_\omega\}$ be a system of ω feature subsets obtained from ω runs of the feature selection routine on different samplings, $\Omega = \sum_{i=1}^{\omega} |S_i|$ be the total number of occurrences of any feature in $\mathcal{S}$ and F_f be the number of occurrences of feature $f \in \mathcal{F}$ in system $\mathcal{S}$. CW was defined as follows:

$$CW(\mathcal{S}) = \sum_{f \in X} \frac{F_f}{\Omega} \cdot \frac{F_f - 1}{\omega - 1}, \tag{1}$$

and CW_{rel} was then derived by adjusting CW on its minimal and maximal possible values $CW_{\min}$ and $CW_{\max}$:

$$CW_{\text{rel}}(\mathcal{S}, \mathcal{F}) = \frac{CW(\mathcal{S}) - CW_{\min}(\Omega, \omega, \mathcal{F})}{CW_{\max}(\Omega, \omega) - CW_{\min}(\Omega, \omega, \mathcal{F})}. \tag{2}$$

2.2. Partially adjusted average Tanimoto index, an unbiased subset-focused measure

CW_{rel} is a *feature-focused* measure, so we looked for a *subset-focused* measure to complement it. Kuncheva's stability index (Kuncheva, 2007) and the stability measure defined in Křížek et al. (2007) are both *subset-focused*, but they can only be used on subsets of equal cardinality. We retained the Average Tanimoto Index *ATI*, also introduced in Somol and Novovičová (2010). *ATI* is a generalization based on Kalousis's similarity measure S_S between two sets S_i and S_j (Kalousis et al., 2005):

$$S_S(S_i, S_j) = \frac{|S_i \cap S_j|}{|S_i \cup S_j|}. \tag{3}$$

This similarity index is computed over all subset pairs, then averaged:

$$ATI(\mathcal{S}) = \frac{2}{\omega(\omega-1)} \sum_{i=1}^{\omega-1} \sum_{j=i+1}^{\omega} S_S(S_i, S_j). \tag{4}$$

ATI is *subset-focused* and *selection-registering*, but it is also *subset-size-biased*. We propose a correction of this index, the partially adjusted average Tanimoto index ATI_{PA}. It is defined as follows:

$$ATI_{PA}(\mathcal{S}) = \text{Max}\left(\frac{ATI(\mathcal{S}) - ATI_{\text{exp}}(\mathcal{S})}{ATI_{\text{max}}(\mathcal{S}) - ATI_{\text{exp}}(\mathcal{S})}, 0\right), \tag{5}$$

where ATI_{max} is the maximal possible value of *ATI* and ATI_{exp} is the expected value of *ATI* when feature subsets are randomly defined. Because we will use a feature selection method which outputs a subset of predefined size, $ATI_{\text{max}} = CW_{\text{max}} = 1$ (when all feature subsets are identical). To obtain ATI_{exp}, we used an experimentally-determined approximation, computed as a function of the proportion of selected features. It should be noted that the correction we perform in ATI_{PA} slightly differs from the one performed in CW_{rel}: CW_{rel} is adjusted on the smallest possible value, while ATI_{PA} is adjusted on the expected value. The max operator ensures that ATI_{PA} is within the [0; 1] interval and not negative as it could happen for the first argument of the max if the stability happens to be worse than random.

2.3. Correlation-based measures

Both *ATI* and *CW* focus on the stability of selected features. While this aspect is important for knowledge discovery, for the purpose of evaluating feature selection methods, the stability of the score over all features may be an interesting information, too. *Selection-exclusion-registering* measures will be too biased when the proportion of excluded features is too high. Correlations of features scores and ranks, on the other hand, provide a more balanced overview, even though the latter will be penalized when lots of features have a similar relevance, which occurs for example when lots of features are equally irrelevant in a very high-dimensional dataset. So, we used the average score (or weight) correlation $\overline{S_W}$ and the average rank correlation $\overline{S_R}$, as described in Kalousis et al. (2005).

3. Analysis on the mathematical model

The objective of this analysis is, by using a very simple model, to compute theoretically the performance and stability of the feature selection depending on the data parameters.

Let us consider a classification problem in D dimensions. The two classes C_1 and C_2 are equally likely. All features are independent. On each feature f_i, the two classes follow respectively a Gaussian distribution $\mathcal{N}(-\mu_i, 1)$ and $\mathcal{N}(\mu_i, 1)$. We consider two types of features: informative features and non-informative features. For all informative features $\mu_i = \mu^*$ and for all non-informative features $\mu_i = 0$. That means only informative features can discriminate the classes and be useful for classification. Let us consider a set of D_g informative features called F_g and a set of D_b non-informative features called F_b, we have $D = D_g + D_b$. A perfect feature selection will keep the D_g informative features and drop the D_b non-informative features. Let us consider a dataset containing N examples drawn from this model. We analyze analytically the behavior of the feature selection performed on this dataset. We express in particular the probability of selecting an informative feature and the stability of the selection.

In our model (Gaussian and independent features), the optimal feature selection method is the t-test selection. This method computes a score for each feature as follows:

$$\hat{S}c(f_i) = \frac{(\hat{\mu}_{i,C_1} - \hat{\mu}_{i,C_2})^2}{\hat{\sigma}^2_{i,C_1} + \hat{\sigma}^2_{i,C_2}},$$

where $\hat{\mu}_{i,C_k}$ and $\hat{\sigma}_{i,C_k}$ represent the estimated mean and variance of feature f_i in class C_k. The higher the score, the more discriminating the feature is. The selection keeps the d features with the highest scores. $(\hat{\mu}_{i,C_1} - \hat{\mu}_{i,C_2})$ follows a Gaussian distribution $\mathcal{N}(2\mu_i, 2/N)$. The probability distribution of $\hat{S}c(f_i)$ can be expressed in our model using the noncentral χ^2 distribution with one degree of freedom:

$$p_{\hat{S}c(f_i)}(x) = F_i\left(\frac{N}{2}x\right) \quad \text{where } F_i \rightsquigarrow \chi^2_1(N\mu_i^2).$$

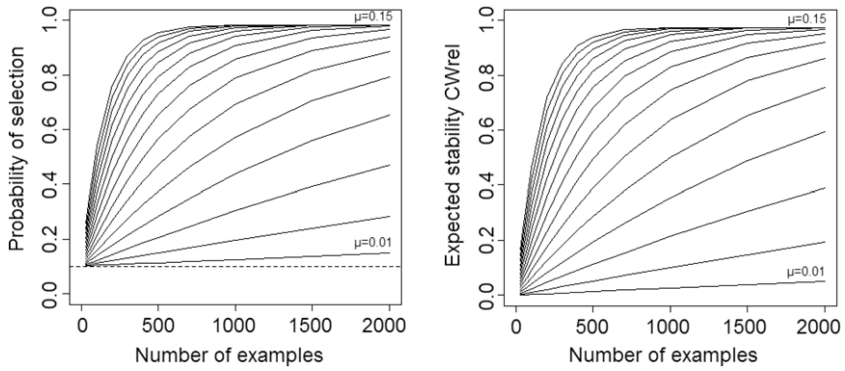

Fig. 1. Left panel: probability for an informative feature to be selected in function on N and μ^*. Right panel: expected stability $E[CW_{\text{rel}}(S)]$ in function on N and μ^*.

We can define two distributions: one for the informative features p_g and one for the non-informative features p_b. Let P_g and P_b be their cumulative distributions. From these probability distributions, we can express the probability for an informative feature to be selected. Let $f_g \in F_g$ be an informative feature and let $x = \hat{S}c(f_g)$ be its score, we have:

$$p_{\text{select}}(f_g) = \int_0^{+\infty} p_g(x) \cdot p_{\text{select}}(f_g|x)dx.$$

If d features are selected, the conditional probability for a feature to be selected corresponds to the probability to be ranked in the d first features, i.e. the probability that the score of f_g is higher than the scores of at least $D - d$ features:

$$p_{\text{select}}(f_g|x) = p(\#(x > \hat{S}c(f_i)) \geq D - d).$$

This can be expressed as the probability that the score of f_g is higher than the scores of i informative features times the probability that the score of f_g is higher than at least the scores of $D - d - i$ non-informative features.

$$p_{\text{select}}(f_g|x) = \sum_{i=0}^{D_g-1} p(\#(x > \hat{S}c(f_i)|f_i \in F_g) = i) \cdot p(\#(x > \hat{S}c(f_i)|f_i \in F_b) \geq D - d - i).$$

All these values follow binomial distribution, the probability of f_g to be selected can therefore be written by:

$$p_{\text{select}}(f_g|x) = \sum_{i=0}^{D_g-1} b(i, D_g - 1, P_g(x)) \cdot B(D - d - i, D_b, P_b(x)),$$

where b and B are respectively the binomial and cumulative binomial distributions. The probability of a non-informative feature to be selected $p_{\text{select}}(f_b)$ can be express in the same way.

We can also express the expected stability CW_{rel} of a system S containing ω feature selections. The expected numbers of occurrences of informative and non-informative features are respectively $p_{\text{select}}(f_g)\omega$ and $p_{\text{select}}(f_b)\omega$. When we include these values in formula (2) we obtain the expected stability $E[CW_{\text{rel}}(S)]$.

In Fig. 1 we show the probability for an informative feature to be selected (left panel) and the expected stability $E[CW_{\text{rel}}(S)]$ in function of N and μ^*. We fixed $D = 1000, D_g = 100, D_b = 900, d = 100$ and $|S| = 1000$. N varies from 30 to 2000. μ^* varies from 0.01 to 0.15, which corresponds to the classification problems where the Bayes error goes from to 0.01 to 0.49.

We can see that the probability for informative features to be selected increases substantially with N. However, for the hardest problems ($\mu^* < 0.05$), even for a sample size of 2000 the probability for informative features to be selected does not reach 1 ($p_{\text{select}}(f_g) = 0.8$ for $\mu^* = 0.05$; 0.15 for $\mu^* = 0.01$). For the easiest problems, $p_{\text{select}}(f_g)$ increases rapidly, almost reaching 1 for $N = 500$ in the case where $\mu^* = 0.15$. Nonetheless, even on those easiest problems, $p_{\text{select}}(f_g)$ is low for very small sample sizes: for $N = 25$, $p_{\text{select}}(f_g) < 0.3$ and for $N = 100$, $p_{\text{select}}(f_g) < 0.5$. Note that, since we always select the 10% top variables, in the worst cases with the smallest sample size, the selection is random and every variable has an equal probability to be selected, so $p_{\text{select}}(f_g)$ tends to 0.1.

The expected stability $E[CW_{\text{rel}}(S)]$ follows similar patterns. Notably $E[CW_{\text{rel}}(S)]$ is below 0.2 for the smallest sample size ($N = 25$) and below 0.5 for $N = 100$, for the easiest problems. Just like the probability for informative features to be selected, on easy problems it increases fast, almost reaching 1 for $N = 500$, but for hard problems it remains low even for $N = 2000$. It should be noted that this setting is ideal for our feature selection methods not only because the t-test is optimal on those features, but also because we know the number of relevant features and configure the filter accordingly to keep this number of features, which optimizes the stability. On the datasets where the real number of relevant features is unknown, stability will likely be lower.

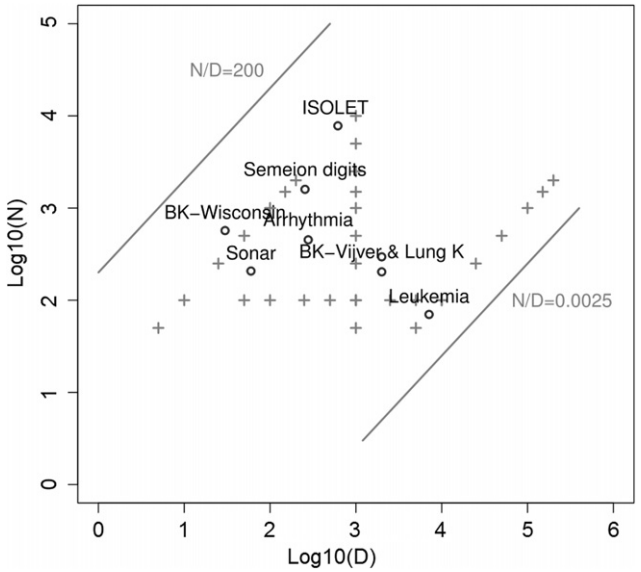

Fig. 2. N/D ratios of some of our artificial datasets (crosses) and real datasets (circles).

4. Analysis on artificial data

4.1. Generation of artificial data

We used two different artificial data structures. The first data distributions used to generate training and test sets consisted of a variable number ($D \in [50; 10\,000]$) of random, independent features. Training sets consisted of $N \in [25; 10\,000]$ examples, so that the N/D ratio goes from 0.0025 to 200, exceeding the range of N/D seen in our real datasets. Fig. 2 shows the N/D ratios of our artificial data and compares them with the ratios of the real data. We see that our artificial data cover the whole range of the real data. Each of the two classes follows a normal distribution defined respectively by $\mathcal{N}(\mu, \sigma^2)$ and $\mathcal{N}(-\mu, \sigma^2)$, where μ is a vector of means such that $|\mu| = D$ and the standard deviation is $\sigma = 1$ for all features. The elements μ_i of μ were drawn from a triangular distribution with a lower limit and mode equal to 0 (probability density function: $f(x) = 2 - 2x$ for $x \in [0; 1]$). To obtain various shapes of strictly decreasing probability densities, simulating varying feature dispersion and relevance, we then raised μ to a power of γ ($\mu_i = \mu_i^{\gamma}$, $\gamma \in [1; 10]$). Finally, μ was scaled down so that either $\mathcal{F}$ would yield a specified Bayes error (ϵ_{Bayes}) or so that the largest μ_i had a specific value μ_{imax}. In our experiments, we chose $\epsilon_{\text{Bayes}} = 0.10$ or $\mu_{\text{imax}} = 0.15$.

In order to study some more realistic data, we also generated data with similar characteristics, with a fixed number of variables ($D = 1000$) and with covariance matrix:

$$\Sigma = \begin{vmatrix} \Sigma_1 & 0 & \cdots & 0 \\ 0 & \Sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \Sigma_{100} \end{vmatrix},$$

where Σ is a block diagonal matrix and Σ_i is a 10×10 square matrix with elements 1 along its diagonal and 0.5 off its diagonal, similarly to the method used in Han and Yu (2010). We created two kinds of correlated data using this covariance matrix: some data with the same μ as with the uncorrelated data, and some data with $\mu \approx 0.1$ for 100 informative variables and $\mu = 0$ for the 900 non-informative variables.

The score used to rank features on the training data was the absolute value of the t-score. Then the top d features with the highest scores were selected.

Picking the right feature selection threshold is still an open problem. In practice we can either fix a threshold on the relevance score, or on a predetermined number of features. In our stability study, these two approaches are related to the same problem: the impact of the size of the selected features set on the stability. In our simulations, we want to study the impact of each parameter one by one. If we use a threshold on the score, the number of selected features will change with the other parameters, making it more difficult to interpret the results. Thus, we chose to fix the number of selected features in order to compare selections of the same size. We did run some simulations with a threshold on the p-value, they are presented in Fig. 1 of supplementary materials which can be found online at http://dx.doi.org/10.1016/j.csda.2013.07.012.

We focused on the t-test because it should perform optimally on independent and normally distributed features such as our artificial data and because it was shown to be as or more stable than other selection methods on small sample microarray

data (Haury et al., 2011). In order to assess the influence of the choice of the selection method, we also performed some selections using shrinkage correlation-adjusted t-score (''CAT score'') (Zuber and Strimmer, 2009), mutual information as implemented in R package SlimPLS, and one-step recursive feature elimination based on support vector machine (SVM-RFE) (Guyon et al., 2002).

For various combinations of parameters N, D, d and γ, 100 training sets were generated. For each of them, feature selection was performed and a linear discriminant analysis (LDA) classifier was trained. Each classifier was then applied to a test set consisting of 10 000 samples. Besides the stability measures described in Section 2, we measured the average classification error rate and the frequency with which each feature was selected, and computed two Bayes errors: $\epsilon_{\text{BayesOptimal}}$ and $\epsilon_{\text{BayesObs}}$. $\epsilon_{\text{BayesOptimal}}$ is the error rate of the Bayes classifier in the best feature subset of size d. It represents the best classification error rate if we select the true d best features and then build an ideal classifier on them. This value can be used as a measure of the problem difficulty, as well as a base point to evaluate the feature selection and classification. $\epsilon_{\text{BayesObs}}$ is the error rate of the Bayes classifier in a given feature selection. This value can be used as a measure of the feature subset quality. The closer $\epsilon_{\text{BayesObs}}$ is to $\epsilon_{\text{BayesOptimal}}$, the better the selection.

4.2. Results on the artificial data

In this set of simulations, we present the performance and stability of the feature selection depending on dataset parameters.

Fig. 3 provides an intuitive overview of feature scoring stability in two extreme settings: one with a very small sample ($N = 50$, left column), the other one with a large sample ($N = 5000$, right column). In the small sample case, feature scores (Fig. 3(a)) do not vary much with feature μ_i, and even though the most relevant features have a slightly higher score than the least relevant ones on average, their scores vary approximately on the same range. This contrasts with the large sample case, where the most relevant features have scores in the [8;12] range, far away from the least relevant ones, which stay in the [0;3] range and are thus easy to tell apart. The correlation between feature scores and μ_i decreases with N.

The resulting ranks reflect the inconsistency of the scores. Fig. 3(b) represents observed feature ranks given feature μ_i, Fig. 3(c) provides a slightly different visualization (observed feature ranks given true feature ranks). Due to the way our model was conceived, the least relevant features can be considered as noise, even though technically they do have some very tiny relevance. So having the worst features poorly ranked among each other is not a bad result. However, in the small sample case, even the true best features only have a slightly better average rank than the other features. For over 90% of the remaining features, the assigned rank is pure noise, as illustrated by scatter plot and standard deviation lines (gray curves) on Fig. 3. In the large sample case, the true best features are ranked much more accurately, even though some noise remains among them, and only the worst half of features are assigned a mostly noisy rank. The results show that in small-sample data there is no correlation between the feature score and ranking obtained from the selection methods and the actual quality of the features.

From our simulations, we computed empirically the probability of each feature to be selected. Fig. 4 presents the evolution of this probability given μ_i. We can see that in the small sample case (Fig. 4(a)), the probability for the most relevant features to actually be selected does not reach 35%, while even the least relevant features have a non negligible probability to be selected. In the large sample case (Fig. 4(b)), the selection is much more accurate: all features with $\mu_i > 0.10$ have a probability to be selected close to 1 and all features with $\mu_i < 0.05$ are almost never selected. Fig. 4(c) shows the evolution of the regression curve from $N = 25$ to $N = 10\,000$: as the sample size increases, the logistic shape increasingly stands out, illustrating how the selection progressively becomes more accurate. But only when the sample size reaches around 1000 observations is the feature selection algorithm able to select the most relevant features with a good sensitivity. In small sample data, the probability to reliably select good features is therefore very low. Fig. 5 presents results obtained on similar simulations but with other filters: CAT score and mutual information. Mutual information performed a bit worse than t-score and CAT score. SVM-RFE (results in Fig. 2 of supplementary material, which can be found online at http://dx.doi.org/10.1016/j.csda.2013.07.012) performed worse than mutual information and was too heavy to compute on $N \geq 5000$. Still, the results all have a similar evolution with sample size.

Fig. 6 presents the evolution of stability measures under varying dataset parameters. The stability is much influenced by the sample size N, with stability measures close to zero when the sample size is around 100 and increasing a lot when additional samples are added to the training set, up to 0.6+ for AIT_{PA} and almost 0.9 for $\overline{S_W}$. It is also much influenced by the total number of variables D, with fairly high values (0.4–0.6) when the dataset only contains 100 samples and 50 variables, quickly reaching close to zero with so few as 1000 variables.

To a lesser extent, stability is also influenced by the selection threshold d. In this case, CW_{rel} is minimal when we select very few variables, then it increases to reach a maximum when we select around 150–180 variables, finally it slowly but regularly decreases as we add more unreliable variables. The shape of this curve illustrates the difficulty to reliably identify even the most relevant variables: trying to keep just the 2 best features will yield highly unstable results, while trying to keep 50 best features will much more likely include maybe the 5 or 10 best features with a very high reliability, leading to a higher stability even though the rest of the selection is not as stable. Note that in this setting, obviously $\overline{S_W}$ and $\overline{S_R}$ do not vary, as they do not take into account the fact that a feature was selected or not.

Variable distribution γ also has some influence on stability: stability measures are minimal when variables are distributed on the triangular distribution, and increase with γ, but following different patterns. $\overline{S_W}$ always increases: this

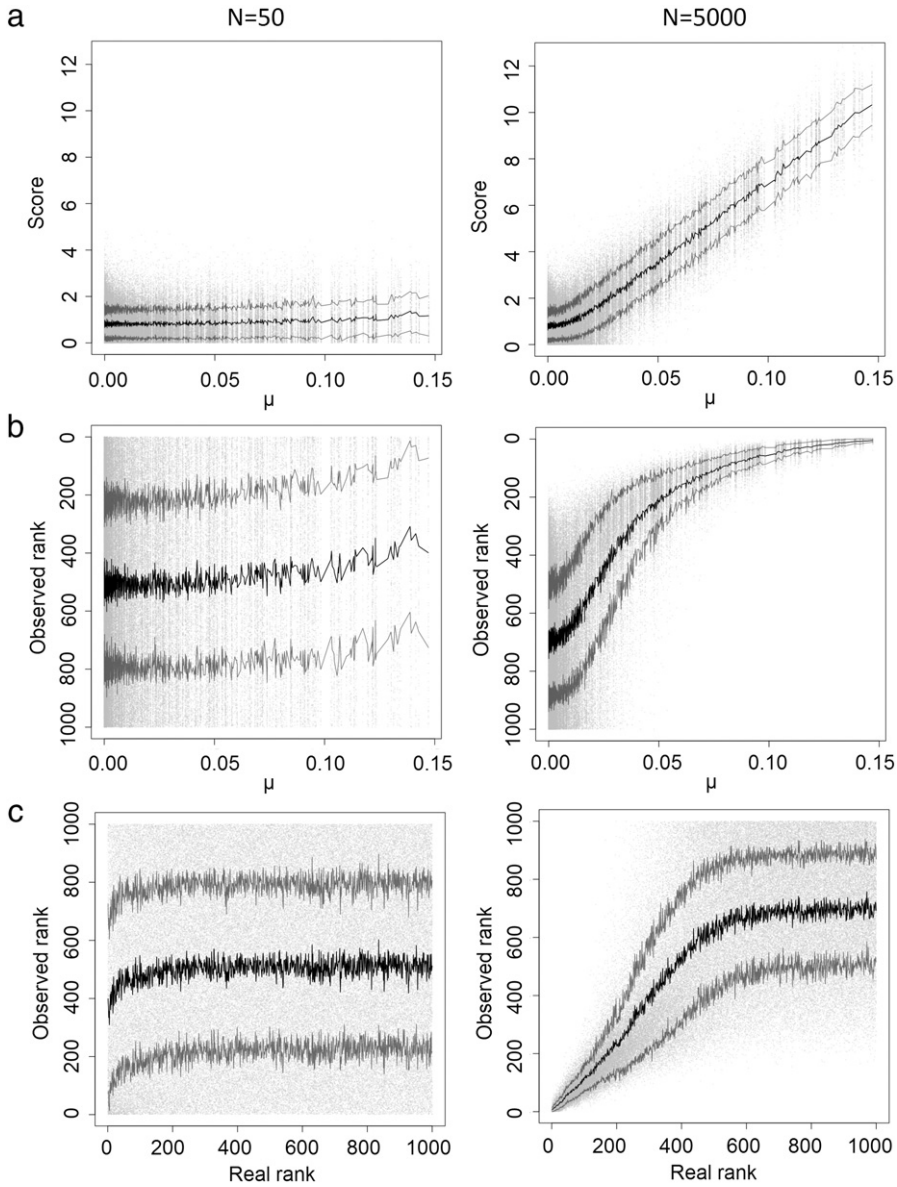

Fig. 3. On artificial data, with $D = 1000$, $d = 100$, $\gamma = 2$ and $N = 50$ (left) or $N = 5000$ (right). (a) Observed score (in absolute value) given real μ_i, (b) observed rank given real μ_i, (c) observed rank given real rank. One point per feature and per training set, the black curve is the average per feature, the gray curves the average $\pm$ standard deviation.

measure is not penalized by ranking difficulties or by instability in the final selection, and only benefits from variables taking extreme values: variables with initial μ_i close to zero do not really lose much score correlation when they get squished even closer to zero, while variables with higher μ_i do benefit from getting more isolated farther away from zero. $\overline{S_R}$ increases at first, but then starts decreasing after reaching a maximum at $\gamma = 5$: this measure first benefits from the increased dispersion of variables with high or intermediate μ_i, but at some point this effect is overcome by the increased difficulty to rank variables with intermediate μ_i (because we kept a constant, realistic Bayes error, the more we stretched the distribution the harder intermediately relevant variables became to identify), which eventually get too close to zero. CW_{rel} and ATI_{PA}, which perform their selection based on a cutoff in the ranks, evolve as a consequence of $\overline{S_R}$, only their decrease is somewhat delayed because they are only affected by the top d rankings. It is likely that a subset-size optimizing feature selection method would see a higher influence of data distribution over selection stability, because it would probably drop the decreasingly relevant variables while keeping the ones increasingly easier to identify.

Fig. 7 presents the evolution of stability measures with sample size using t-score and CAT score filters on non correlated and correlated datasets. Non correlated and correlated data (or the two kinds of correlated data) cannot really be formally compared since a Bayes error cannot be computed on correlated data, and introducing the correlations modifies the

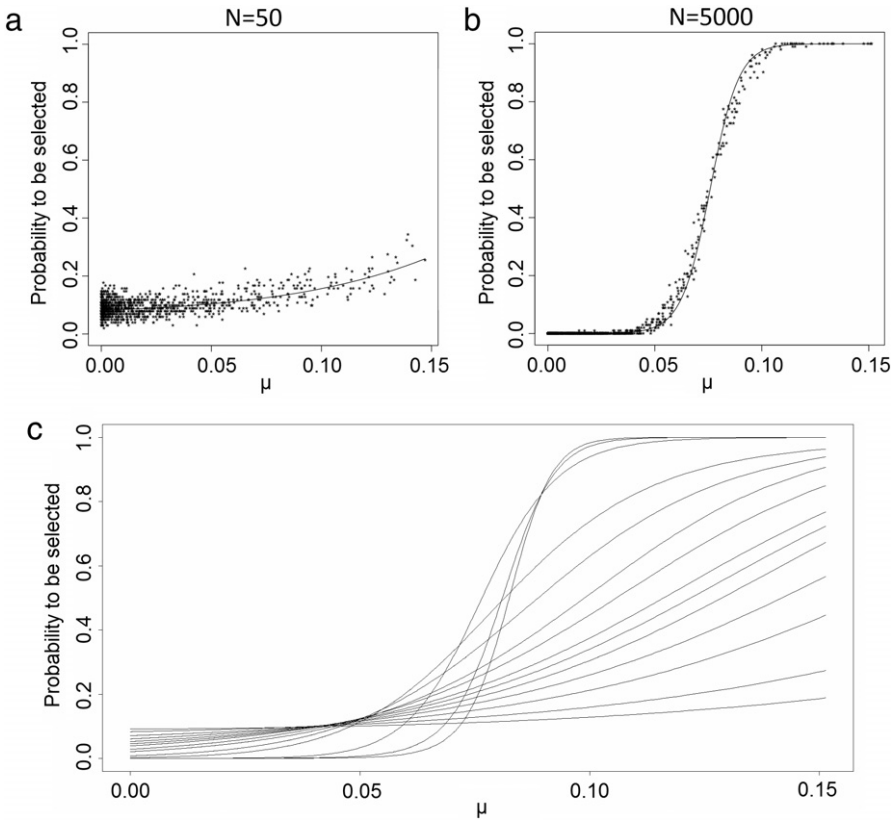

Fig. 4. Observed probability for a feature to be selected given its real μ_i. On artificial data, with $D = 1000$, $d = 100$ and $\gamma = 2$. (a) $N = 50$ (b) $N = 5000$, one point per feature and the curve was obtained via logistic regression. (c) N varying from 25 (curve with the lowest value at $\mu_i = 0.15$) to 10 000 (first curve to reach 1). As the sample size grows, the logistic shape increasingly stands out.

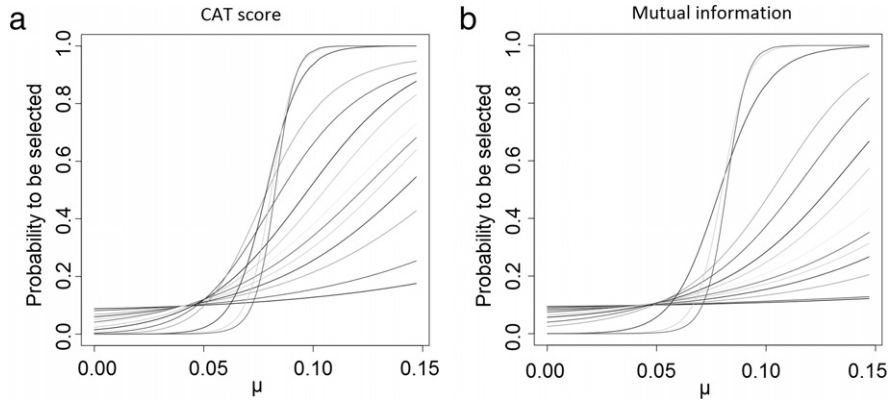

Fig. 5. Observed probability for a feature to be selected given its real μ_i. On artificial data, with $D = 1000$, $d = 100$, $\gamma = 2$ and N varying from 25 (curve with the lowest value at $\mu_i = 0.15$) to 10 000 (first curve to reach 1). (a) CAT score filter. (b) Mutual information filter.

distribution of μ, but we can still observe that correlation does not fundamentally change the stability nor its evolution with sample size. CAT score performed worse than t-score on correlated data, even on large samples. This could be explained by the fact that this method tries not to select correlated variables, so it tends to keep fewer variables per block even in the most significant blocks.

Fig. 8 reports the classification error rates obtained from the same selections as in Fig. 6. The dashed curves indicate $\epsilon_{\text{BayesOptimal}}$, the best possible classification error rate on the dataset with d features (selecting the true d best features then building an ideal classifier on it), the gray curves indicate $\epsilon_{\text{BayesObs}}$, the best possible classification error rate if building an ideal classifier on the selected features, the black curve indicate the observed error rate. The error rate and Bayes error on the selected features increase when the training sample size decreases. For small sample, this increase is very strong and we see large differences between $\epsilon_{\text{BayesOptimal}}$ and $\epsilon_{\text{BayesObs}}$, meaning that the feature selections have bad performance. The

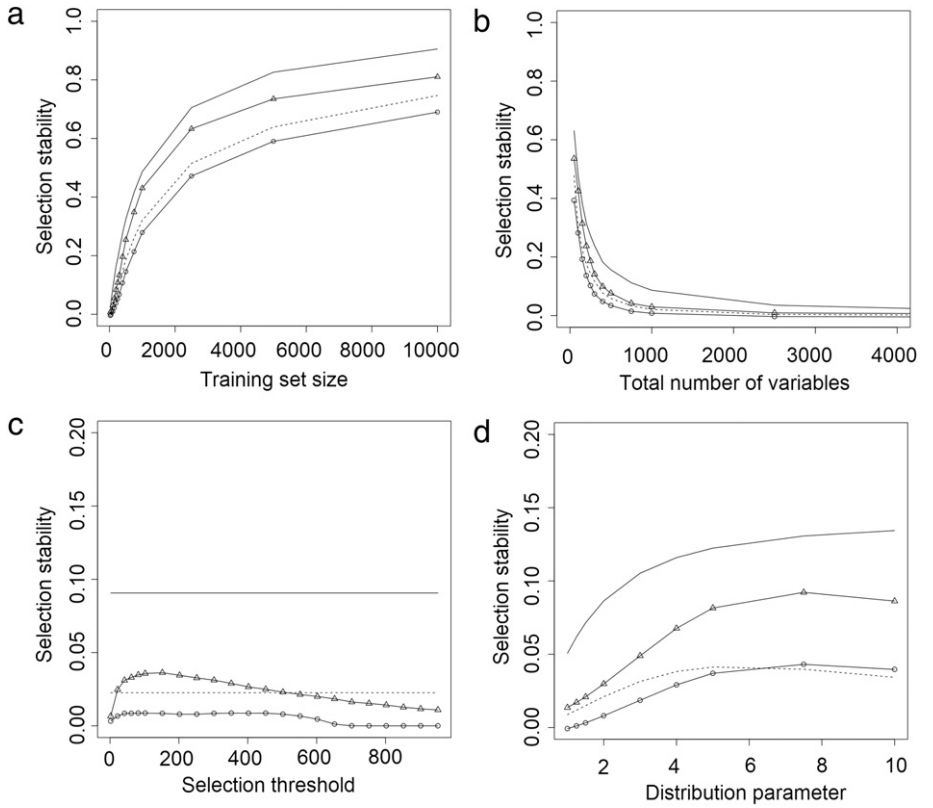

Fig. 6. Evolution of stability measures CW_{rel} (triangles), ATI_{PA} (circles), $\overline{S_W}$ (continuous line) and $\overline{S_R}$ (dashes) given: (a) $N \in [25;\ 10\,000]$ (b) $D \in [50;\ 10\,000]$ (c) $d \in [2;\ 1000]$ and (d) $\gamma \in [1;\ 10]$. When the were not the one being iterated on, parameter values were: $N = 100$, $D = 1000$, $d = D \cdot 10\%$, $\gamma = 2$.

error rate is also influenced by an increase in the total number of variables (Fig. 8(b)): on the small tested sample ($N = 100$), with only 50 variables the classification error rate is lower than 20% (for an optimal Bayes error of over 16%), but when we reach 2500 variables the error rate is over 40%, with a Bayes error on the selected features over 30%. This is a consequence of the dilution of the information: since we kept a constant $\epsilon_{\text{BayesOptimal}}$ (to account for the fact that real datasets do not necessarily contain all the variables needed to do a perfect prediction even if we had an unlimited supply of samples), when the dimension increases, the information gets spread over weak features, so $\epsilon_{\text{BayesObs}}$ becomes higher. These weak features do not contain a lot of information, so the classification error rate increases.

The evolution of the error rate with the selection threshold d (Fig. 8(c)) might seem a little more surprising. Particularly, the regular decrease of the Bayes error on the selection may give the impression that, as we increase the threshold, the feature selection keeps including relevant variables in the proper order. This, of course, is not the case: as we loosen the threshold for inclusion, we include more and more slightly relevant variables, more and more randomly (as can be deducted from the stability measures seen previously). Even for small thresholds, the selection does not contain necessarily the most relevant variables. This point is illustrated by the rapid growth of the distance between the $\epsilon_{\text{BayesOptimal}}$ and the $\epsilon_{\text{BayesObs}}$ curves. When the $\epsilon_{\text{BayesObs}}$ curve finally reaches the $\epsilon_{\text{BayesOptimal}}$ curve, it is not because the selection is good but because all variables are included. That is why the classification error does not decrease much when $d > 200$ (computation time, though, increases substantially): the information contained in the most relevant variables is flooded by the poorly relevant but selected variables. Similar results were obtained when increasing the sample size to $N = 1000$, only with lower error rates and higher stability values (results not shown).

Distribution parameter γ seems to have a higher influence on error rate than on stability. The higher γ, the lower the error rate (Fig. 8(d)): as γ increases, the number of highly relevant features decreases but their discrimination power increases and allow for a better selection (this is diluted by the increased instability on the other features when we observe stability measures, but this stands out when we observe how $\epsilon_{\text{BayesObs}}$ gets closer and closer to $\epsilon_{\text{BayesOptimal}}$) and a better classification accuracy. However, even though the classification error rate does improve, it does so slower than $\epsilon_{\text{BayesObs}}$. An explanation to this difference is that, despite an improvement in the most relevant variables, the classifier is still hampered by the remaining not-so-relevant variables.

Fig. 9 shows the stability CW_{rel} as a function of the number of training examples for a constant N/D ratio. The different curves correspond to different N/D ratios (from 0.01 to 10). We see the stability is constant for a fixed N/D ratio, except for some variations in the lowest dimension values for $N/D \geq 5$, caused by random variations in the problem difficulty (very few selected variables on those specific points). For small-sample problem, where $N/D \ll 1$, the stability depends

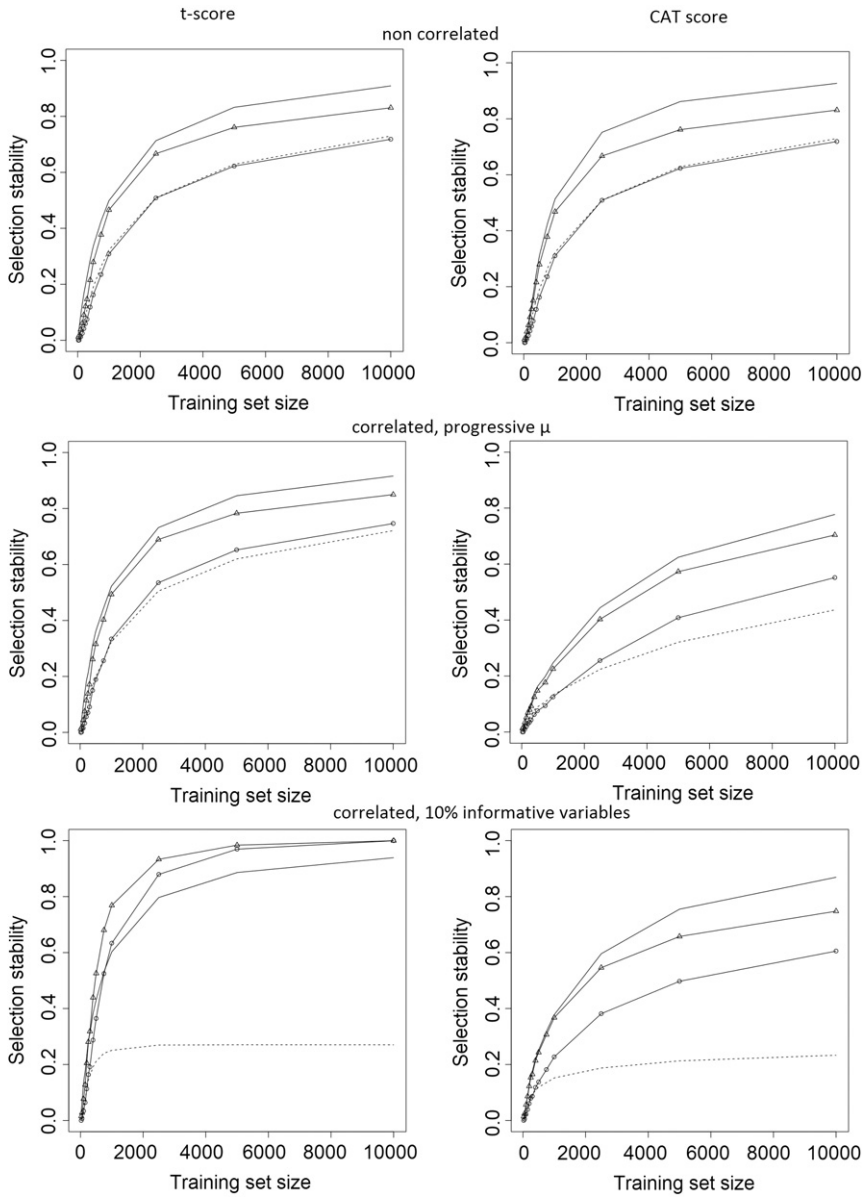

Fig. 7. Evolution of stability measures CW_{rel} (triangles), ATI_{PA} (circles), $\overline{S_W}$ (continuous line) and $\overline{S_R}$ (dashes) when using t-score (left) and CAT score (right) filter, on non correlated data (top), correlated data with progressive μ (middle), and correlated data with 100 informative variables vs 900 non-informative variables (bottom). $N = 100$, $D = 1000$, $d = 100$, $\gamma = 2$.

on the N/D ratio, not on N or D alone. In gene expression data, the N/D ratio typically ranges from 0.001 to 0.1, which leads to a maximum stability of 0.2 in our simulations. Note that those simulations are based on Gaussian, uncorrelated features, which is one of the easiest classification problems. Moreover we use a selection based on the t-test score, which is the optimal feature selection method in this context. In real data, the distribution of the classes is much more complex than Gaussian and the optimal feature selection is unknown. So, the stability on real data should be lower than the stability on artificial data: the values reported on Fig. 9 should be considered as upper bounds.

5. Analysis on real data

5.1. Description of the real data

We experimented with eight publicly available datasets, presented in Table 1. When the original dataset dealt with a multiclass classification task, we reduced the problem to a 2-class, 1-vs-all classification. For each datasets, for different

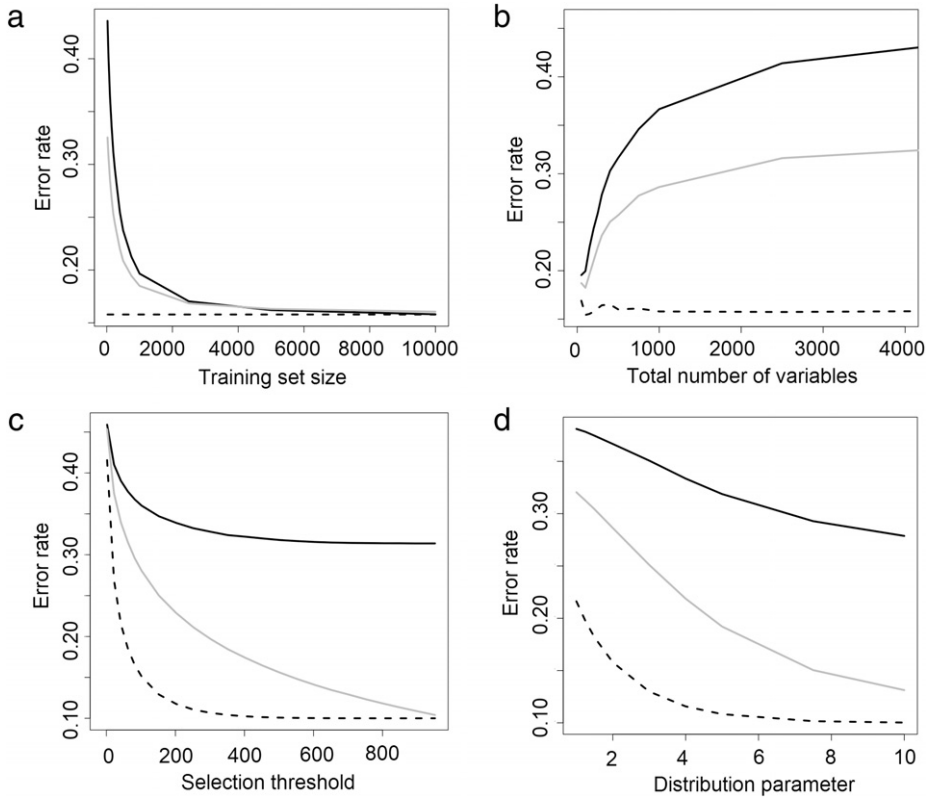

Fig. 8. Evolution of error rate (black), $\epsilon_{\text{BayesObs}}$ (gray) and $\epsilon_{\text{BayesOptimal}}$ (dashes) given: (a) $N \in [25; 10\,000]$ (b) $D \in [50; 10\,000]$ (c) $d \in [2; 1000]$ and (d) $\gamma \in [1; 10]$. When the were not the one being iterated on, parameter values were: $N = 1000$, $D = 1000$, $d = D \cdot 10\%$, $\gamma = 2$.

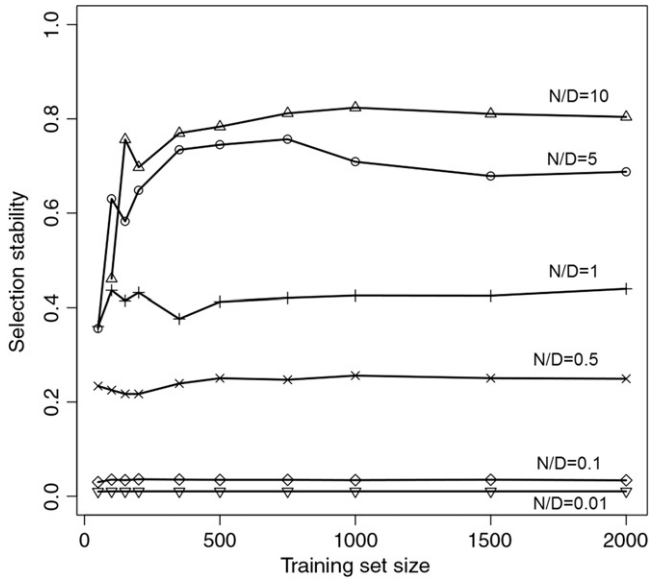

Fig. 9. Evolution of CW_{rel} with the number of training examples for a constant N/D ratio. The different curves correspond to different N/D ratios (lowest: 0.01; highest: 10).

values of N, 200 training sets were generated by randomly drawing examples from the dataset (without replacement). For each of them, feature selection was performed and a classifier was trained (using the same methods as with the artificial data). Each classifier was then applied to a test set consisting of the samples not included in the corresponding training set. We measured the stability of the feature selection across the training sets and the average classification error rate.

Table 1
Characteristics of the real datasets.

Name	N	D	N/D	Source
Leukemia	72	7129	0.01	(Golub et al., 1999)
Lung cancer	203	2000	0.10	(Bhattacharjee et al., 2001)
Breast cancer Vijver	295	2000	0.15	(van de Vijver et al., 2002)
Arrhythmia	452	279	1.6	UCI repository (Frank and Asuncion, 2010)
Sonar	208	60	3.5	UCI repository
Semeion digits	1593	256	6.2	UCI repository
ISOLET	7797	617	12.6	UCI repository
Breast cancer Wisconsin	569	32	17.7	UCI repository

Table 2
Classification error rate and selection stability on the real datasets.

	Breast cancer Vijver ($D = 2000$)		Lung cancer ($D = 2000$)		Leukemia ($D = 7129$)	
	$N = 50$ $N/D = 0.025$	$N = 100$ $N/D = 0.05$	$N = 50$ $N/D = 0.025$	$N = 100$ $N/D = 0.05$	$N = 20$ $N/D = 0.003$	$N = 35$ $N/D = 0.005$
Error rate (%)	38.2	37.4	8.0	6.1	9.2	4.0
CW_{rel}	0.20	0.26	0.46	0.51	0.24	0.30
ATI_{PA}	0.06	0.10	0.26	0.30	0.13	0.18
$\overline{S_R}$	0.09	0.14	0.51	0.58	0.22	0.26
$\overline{S_W}$	0.33	0.41	0.81	0.85	0.55	0.60
	Breast cancer Wisconsin ($D = 30$)		Sonar ($D = 60$)		Semeion digits ($D = 256$)	
	$N = 50$ $N/D = 1.67$	$N = 284$ $N/D = 9.48$	$N = 50$ $N/D = 0.83$	$N = 100$ $N/D = 1.67$	$N = 100$ $N/D = 0.39$	$N = 796$ $N/D = 3.11$
Error rate (%)	9.5	9.0	32.5	31.0	4.0	3.6
CW_{rel}	0.75	0.97	0.28	0.30	0.54	0.65
ATI_{PA}	0.62	0.95	0.14	0.15	0.34	0.46
$\overline{S_R}$	0.91	0.97	0.51	0.59	0.73	0.92
$\overline{S_W}$	0.93	0.97	0.68	0.74	–	0.96
	ISOLET ($D = 618$)			Arrhythmia ($D = 279$)		
	$N = 100$ $N/D = 0.16$	$N = 618$ $N/D = 1$	$N = 3900$ $N/D = 6.31$	$N = 50$ $N/D = 0.18$	$N = 100$ $N/D = 0.36$	$N = 226$ $N/D = 0.81$
Error rate (%)	4.3	2.7	2.4	34.0	30.3	27.5
CW_{rel}	0.32	0.71	0.86	0.34	0.41	0.48
ATI_{PA}	0.15	0.53	0.75	0.16	0.22	0.28
$\overline{S_R}$	0.42	0.81	0.93	0.42	0.46	0.51

5.2. Results on the real data

Table 2 presents stability measures and error rates observed on real datasets for various training set sizes. Because of varying underlying problem difficulties, differences in absolute values between datasets cannot be attributed only to differences in dimensions. Nonetheless, those results provide an overview of the kind of values which can be expected, and of their dependency on sample size. Globally, error rates decrease when N increases, with a faster evolution with N when N/D is low, which is quite similar to what we observed on our artificial data.

Stability measures, although different in absolute value, share a same trend, opposite to the error rate. Stability increases with training set size, but globally it remains rather low as long as $N/D < 1$, although higher than on our artificial data with similar dimensions. Notably, the Vijver breast cancer dataset has a higher stability but a similar error rate compared to our non-correlated artificial dataset. This discrepancy might be caused by the different correlation structure in our artificial datasets compared to those in the real datasets: in our second set of correlated artificial data, we observed a higher stability with a higher error rate. With an easier problem, we would have a higher stability with a lower error rate. So by adjusting both the problem difficulty (μ distribution) and the amount of correlation, it might be possible to obtain results which get closer to the Vijver dataset, as well as the others. Another explanation could be the differences in the experimental design between artificial and real data. In artificial data, the stability is computed from a set of 100 independent datasets. In real data, we have only one dataset that is split in two subsets, this process is repeated 200 times. The 200 splits are not independent, there is therefore a bias that artificially increases the measured stability.

Nonetheless, in a similar way to the error rate, stability increases with N, with a faster evolution with N when N/D is low. And stability values are the lowest when the N/D ratio is the lowest. Performing a simple regression on the data presented in Table 2 shows an independent correlation between, on one hand, stability (CW_{rel}) and problem difficulty (measured as the smallest error rate obtained on the dataset), and on the other hand, stability and the N/D ratio.

6. Discussion and conclusion

In this paper, we analyzed the performance of feature selection and especially its stability in small-sample high-dimension settings. We used existing measures of stability and we introduced ATI_{PA}, a modification of the ATI stability measure adjusted to avoid a bias on the number of selected features. We investigated the behavior of stability depending on the number of examples, features, selected features and distribution of the discrimination power of features. We show that in small sample problems the probability to select the best features is very low even with an optimal feature selection method. We show empirically that for Gaussian data, the stability depends on the N/D ratio. The results of our simulations on artificial data show that the stability is dramatically low (almost 0) for $N/D \leq 0.1$. In real data, where variable distributions are unknown but certainly more complex than Gaussian models, the used feature selection methods are not perfectly adapted, so we cannot expect better results. It is even highly likely that the situation is worse, so we can consider our simulation results as an upper bound of stability for real data in function of their size (N and D). This leads to the conclusion that for any high-dimension small-sample data, it is not possible to obtain a stable feature selection for a classification task. Although we mainly explored the use of the t-test filter with a fixed threshold on the number of selected variable, this conclusion also holds for the other selection methods we tested, and the method of choice of the feature selection threshold (a predetermined number of variables versus a threshold on the relevance score) did not make much difference either. These conclusions could explain a lot of results published in various domains. For example, in medicine, several gene expression signatures of a given cancer have been identified on different microarray datasets, but there is almost no overlap between signatures when the results on a given dataset are verified on the other datasets (Miecznikowski et al., 2010).

To improve the stability of feature selection, the first option is simply to increase the number of examples in the datasets. While research projects are necessarily limited in that respect, it is definitely hopeless to try to construct a stable classifier based only on a few tens of examples. A second way would be to find reliable methods to reduce the dimensionality of the data prior to applying usual filters. For instance, *a priori* knowledge and unsupervised methods could be used to filter out some of the irrelevant variables. It could also be interesting to exploit the redundancy of the feature to compress the dimensions of the data. As future work, it would be interesting to perform similar tests with more complex data and on other feature selection methods, particularly on those specifically designed to improve stability, such as Consensus Group Stable feature selection or Complementary Pairs Stability Selection (Shah and Samworth, 2011).

References

Bhattacharjee, A., Richards, W.G., Staunton, J., Li, C., Monti, S., Vasa, P., Ladd, C., Beheshti, J., Bueno, R., Gillette, M., et al., 2001. Classification of human lung carcinomas by mrna expression profiling reveals distinct adenocarcinoma subclasses. Proceedings of the National Academy of Sciences of the United States of America 98, 13790–13795.

Ein-Dor, L., Zuk, O., Domany, E., 2006. Thousands of samples are needed to generate a robust gene list for predicting outcome in cancer. Proceedings of the National Academy of Sciences 103, 5923–5928.

Frank, A., Asuncion, A., 2010. UCI machine learning repository.

Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., 1999. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286, 531–537.

Guyon, I., Weston, J., Barnhill, S., Vapnik, V., 2002. Gene selection for cancer classification using support vector machines. Machine Learning 46, 389–422.

Han, Y., Yu, L., 2010. A variance reduction framework for stable feature selection. In: Webb, G.I., Liu, B., Zhang, C., Gunopulos, D., Wu, X. (Eds.), ICDM. IEEE Computer Society, pp. 206–215.

Hastie, T., Tibshirani, R., Friedman, J., 2009. The Elements of Statistical Learning, second ed. In: Springer Series in Statistics, Springer New York Inc., New York, NY, USA.

Haury, A.-C., Gestraud, P., Vert, J.-P., 2011. The influence of feature selection methods on accuracy, stability and interpretability of molecular signatures. PLoS One 6, e28210.

Ioannidis, J.P., 2005. Microarrays and molecular research: noise discovery? Lancet 365, 454–455.

Jain, A.K., Chandrasekaran, B., 1982. 39 dimensionality and sample size considerations in pattern recognition practice. In: Handbook of Statistics, Vol. 2. pp. 835–855.

Kalousis, A., Prados, J., Hilario, M., 2005. Stability of feature selection algorithms. In: ICDM. IEEE Computer Society, pp. 218–225.

Kalousis, A., Prados, J., Hilario, M., 2007. Stability of feature selection algorithms: a study on high-dimensional spaces. Knowledge and Information Systems 12, 95–116.

Krížek, P., Kittler, J., Hlavác, V., 2007. Improving stability of feature selection methods. In: Kropatsch, W., Kampel, M., Hanbury, A. (Eds.), Computer Analysis of Images and Patterns. In: Lecture Notes in Computer Science, vol. 4673. Springer, Berlin, Heidelberg, pp. 929–936.

Kuncheva, L.I., 2007. A stability index for feature selection. In: Devedzic, V. (Ed.), Artificial Intelligence and Applications. IASTED/ACTA Press, pp. 421–427.

Michiels, S., Koscielny, S., Hill, C., 2005. Prediction of cancer outcome with microarrays: a multiple random validation strategy. Lancet 365, 488–492.

Miecznikowski, J.C., Wang, D., Liu, S., Sucheston, L., Gold, D., 2010. Comparative survival analysis of breast cancer microarray studies identifies important prognostic genetic pathways. BMC Cancer 10, 573.

Pudil, P., Somol, P., 2008. Identifying the most informative variables for decision-making problems—a survey of recent approaches and accompanying problems. Acta Oeconomica Pragensia 2008, 37–55.

Saeys, Y., Inza, I., Larrañaga, P., 2007. A review of feature selection techniques in bioinformatics. Bioinformatics 23, 2507–2517.

Shah, R.D., Samworth, R.J., 2011. Variable selection with error control: another look at stability selection. ArXiv e-prints.

Somol, P., Grim, J., Pudil, P., 2009. Criteria ensembles in feature selection. In: Benediktsson, J.A., Kittler, J., Roli, F. (Eds.), MCS. In: Lecture Notes in Computer Science, vol. 5519. Springer, pp. 304–313.

Somol, P., Novovičová, J., 2010. Evaluating stability and comparing output of feature selectors that optimize feature subset cardinality. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 1921–1939.

van de Vijver, M.J., He, Y.D., van 't Veer, L.J., Dai, H., Hart, A.A., Voskuil, D.W., Schreiber, G.J., Peterse, J.L., Roberts, C., Marton, M.J., Parrish, M., Atsma, D., Witteveen, A., Glas, A., Delahaye, L., van der Velde, T., Bartelink, H., Rodenhuis, S., Rutgers, E.T., Friend, S.H., Bernards, R., 2002. A gene-expression signature as a predictor of survival in breast cancer. New England Journal of Medicine 347, 1999–2009.

Yao, W., Wang, Q., 2013. Robust variable selection through MAVE. Computational Statistics & Data Analysis 63, 42–49.

Zuber, V., Strimmer, K., 2009. Gene ranking and biomarker discovery under correlation. Bioinformatics 25, 2700–2707.