Control and Cybernetics

vol. **40** (2011) No. 2

The Monte Carlo feature selection and interdependency discovery is unbiased*

by

Michał Dramiński[1], Marcin Kierczak[2], Agnieszka Nowak-Brzezińska[3], Jacek Koronacki[1] and Jan Komorowski[4,5]

[1] Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland

[2] Swedish University of Agricultural Sciences, Uppsala, Sweden

[3] Institute of Computer Science, University of Silesia, Sosnowiec, Poland

[4] The Linnaeus Centre for Bioinformatics, Uppsala University and The Swedish University of Agricultural Sciences, Uppsala, Sweden

[5]Interdisciplinary Centre for Mathematical and Computer Modelling, Warsaw University, Poland

Abstract: We show that the Monte Carlo feature selection algorithm for supervised classification proposed, by Dramiński et al. (2008), is not biased towards features with many categories (levels or values). While the algorithm, later extended to include the functionality of discovering interdependencies between features, is surprisingly simple and has been successfully used on many biological data and transactional data of commercial origin, and it has never revealed any bias of the type mentioned, the alleged property of its unbiasedness required a closer scrutiny which is thus provided here. Admittedly, the algorithm does reveal some bias coming from another source, but it is negligible. Hence our final claim is that the algorithm is practically unbiased and the results it provides can be considered fully reliable.

Keywords: supervised classification, feature selection, feature interactions, high-dimensional problems, applications to genomic and proteomic data.

1. Introduction

Preselection of informative features or attributes for supervised classification is a crucial task whenever observations (records, samples) include a very large number of features and only a few of them contribute to a given classification problem, while all the remaining ones are essentially a noise or a nuisance.

*Submitted: November 2010; Accepted: March 2011.

The task becomes particularly challenging when data sets contain a very small number of observations, say, of the order of tens, versus thousands of features per observation. Typical examples include genomic and proteomic data. Another obvious example is some transactional data of commercial origin.

No wonder that feature selection for supervised classification has attracted much attention and that a significant progress in this area of research has been achieved in recent years; for a brief account, up to 2002, see Dudoit and Fridlyand (2003), and for an extensive survey and later developments see Saeys et al. (2007) (an early successful method, not mentioned by Saeys et al., 2007, and called nearest shrunken centroids, was developed by Tibshirani et al., 2002 and 2003). Recently, a Bayesian technique of automatic relevance determination, the use of support vector machines, and the use of ensembles of classifiers, all these either alone or in combination, have proved particularly promising. For further details see Li et al. (2002), Lu at al. (2007), Chrysostomou et al. (2008) and the literature there. In the context of feature selection, the last developments by the late Leo Breiman deserve special attention. In his Random Forests (RFs), he proposed to make use of the so-called variable (i.e. feature) importance for feature selection (see Breiman and Cutler, 2008, and, e.g., Diaz-Uriarte and de Andres, 2006). While feature selection by measuring variable importance in RFs should be seen as a very promising method, the problem with this approach is that variable importance as originally defined is biased towards variables with many categories (levels or values) and variables that are correlated; see Strobl et al. (2007), Archer and Kimes (2008). Accordingly, proper debiasing is needed in order to obtain true ranking of features; see Strobl et al. (2008). The problem is real, not just academic, as examples in Strobl et al. (2007) show; indeed, it arises when, e.g., both genetic and environmental variables are considered as potential predictors.

Generally speaking, feature selection may be performed either prior to building the classifier, or as an inherent part of this process. The first of these two approaches is referred to as filter methods. One potential advantage of the filter approach is that it provides a subset of features that contribute the most to a given classification task, and therefore are informative or "relatively important" to the task *regardless* of the classifier that will be used. In other words, the filter approach should be seen as a way of providing an objective measure of relative importance of each feature for a particular classification task. Of course, for this to be the case, a filter method used for feature selection should be capable of incorporating interdependencies between the features. Indeed, the fact that a feature may prove informative only in conjunction with some other features, but not alone, should be taken into account. Clearly, the aforementioned algorithms for measuring variable importance in RFs possess this capability.

In 2008, a novel and effective filter method for ranking features according to their importance for a given supervised classification task has been introduced by Dramiński et al. (2008). The method, based on Monte Carlo approach and termed accordingly the Monte Carlo Feature Selection (or MCFS) algorithm, takes into account interdependencies between features when building the ranking. It bears some remote similarity to the RF methodology, but differs entirely in the way feature ranking is performed. Specifically, the method is conceptually simpler. A more important property of the MCFS algorithm is that it provides *explicit* information about interdependencies among informative features; see Dramiński et al. (2010) for the MCFS-ID algorithm (with ID standing for Interdependency Discovery), in which the functionality of discovering interdependencies among informative features is included.

The MCFS and MCFS-ID algorithms have been successfully used on many biological data as well as transactional data of commercial origin, geological and U.S. Census data; see Dramiński et al. (2008, 2010), Kierczak et al. (2009, 2010), Kierczak (2009). While they have never revealed any bias towards features with many categories, the alleged property of the algorithm unbiasedness called for a systematic verification. Indeed, the bias observed for the RFs of Breiman stems, among others, from the fact that they are based on constructing many decision trees and that each separate tree is known to be a biased classifier in the sense mentioned (see Strobl et al., 2007). And, although built in a completely different way, the MCFS-ID algorithm is also based on constructing many decision trees (it is worth a note here that while the other source of the RFs bias is their reliance on sampling with replacement, the MCFS-ID relies on sampling without replacement which is known to introduce no bias).

The MCFS algorithm from Dramiński et al. (2008) is briefly recapitulated in Section 2. In Section 3, an experimental study is presented, which shows that the MCFS (and, thus, the MCFS-ID) algorithm is not biased towards features with many categories, although it does reveal a mild bias from some other source. Since the latter is negligible, we can rely on the ranking of features provided by the algorithm and we claim the MCFS-ID to be practically unbiased. We close with concluding remarks in Section 4.

2. Monte Carlo feature selection

The Monte Carlo feature selection (MCFS) part of the algorithm is conceptually simple, although computer-intensive. We consider a particular feature to be important, or informative, if it is likely to take part in the process of classifying samples into classes "more often than not". This "readiness" of a feature to take part in the classification process, termed *relative importance* of a feature, is measured via intensive use of classification trees. We emphasize that the method utilizes thousands of classifiers, not in order to build an overall general classifier but rather to identify which features are important for the given classification task. Once the subset of such features is known, we may (but do not necessarily have to) proceed to build a general classifier.

In the main step of the procedure, we estimate relative importance of features by constructing trees for randomly selected subsets of features. More precisely, out of all d features, we select s subsets of m features, m being fixed and $m << d$,

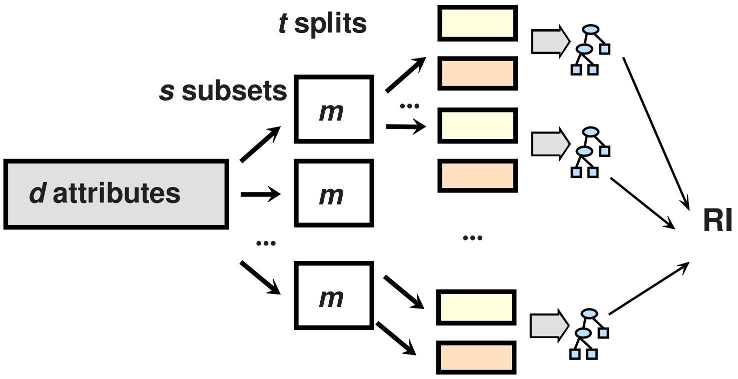

Figure 1. Block diagram of the main step of the MCFS procedure

and for each subset of features, t trees are constructed and their performance assessed. Each of the t trees in the inner loop is trained and evaluated on a different, randomly selected training and test data sets. These sets come from a random split of the full data set into two subsets. Every time, about 66% out of all n samples, is used for training and the remaining samples are used for testing. The split is performed in a stratified manner, i.e., the proportions of classes in the original data set are preserved. See Fig. 1 for a block diagram of the procedure.

Eventually, $s \cdot t$ trees are constructed and evaluated in the main step of the procedure. Both s and t should be sufficiently large, so that each feature has a chance to appear in many different subsets of features and that randomness due to inherent variability in the data is properly accounted for. A crude (and biased) measure of relative importance of a particular feature could be defined as the overall number of splits made on that feature in all nodes of all the $s \cdot t$ trees. However, it is clear that for any particular split, its contribution to the overall relative importance of the feature should be weighted by the information gain achieved by the split, the number of samples in the split node and by the classification ability of the whole tree.

In order to determine relative importance of a particular feature, let us first recall weighted accuracy of a tree as a means to assess classification ability of the tree on a test set. For a classification problem with c classes, let n_{ij} denote the number of samples from class i classified as those from class j; clearly, $i, j, = 1, 2, \ldots, c$ and $\sum_{i,j} n_{ij} = n$, the number of all samples. Now, we define weighted accuracy as

$$wAcc = \frac{1}{c} \sum_{i=1}^{c} \frac{n_{ii}}{n_{i1} + n_{i2} + \cdots + n_{ic}}, \tag{1}$$

i.e., as the mean of c true positive rates.

The relative importance of feature g_k, RI_{g_k}, is defined as

$$\mathrm{RI}_{g_k} = \sum_{\tau=1}^{st} (wAcc_\tau)^u \sum_{n_{g_k}(\tau)} \mathrm{IG}(n_{g_k}(\tau)) \left(\frac{\text{no. in } n_{g_k}(\tau)}{\text{no. in } \tau} \right)^v, \tag{2}$$

where summation is over all the $s \cdot t$ trees, $wAcc_\tau$ stands for the weighted accuracy of the τ-th tree and, within each τ-th tree, the summation is over all nodes $n_{g_k}(\tau)$ of the tree on which the split is made on feature g_k, $\mathrm{IG}(n_{g_k}(\tau))$ stands for information gain for node $n_{g_k}(\tau)$, (no. in $n_{g_k}(\tau)$) denotes the number of samples in node $n_{g_k}(\tau)$, (no. in τ) denotes the number of samples in the root of the τ-th tree, and u and v are fixed positive reals. Information gain can be measured, e.g., by Gini Index, entropy or Gain Ratio (in our implementations we use j48 tree from WEKA, which is a version of C4.5 tree with Gain Ratio, this ratio being a way to reduce the bias of the tree towards features with many categories).

Note that by taking, say, $u = 2$, trees with low $wAcc$ are penalized more severely than when taking $u = 1$. Similarly, the greater the v, the smaller the influence of node $n_{g_k}(\tau)$ with a given ratio (no. in $n_{g_k}(\tau)$)/(no. in τ) on RI_{g_k}, unless $n_{g_k}(\tau)$ is the root of the tree. And, for any fixed positive v, the influence of any particular node on RI_{g_k} decreases monotonically with the number of samples in this node. In this way, and especially for low-level nodes in a tree, the fact that information gains can be very high is taken into account, while only very small subsets of data are split.

In the experimental study described in the next section (as well as in our current applications of the method) we use a normalized version of RI_{g_k} for feature g_k, in which "raw" RI_{g_k} is divided by the number of these sets out of all the s randomly selected subsets of features which include feature g_k.

In the procedure, there are five parameters, m, s, t, u and v to be set by an experimenter. A detailed discussion on how to set values of these parameters can be found in Dramiński et al. (2008). Our experience suggests to use u and v set to 1 as the default value. The choice of subset size m of features selected for each series of t experiments should take into account the trade-off between the need to prevent informative features from being masked too severely by the relatively most important ones and the natural requirement that s be not too large. Indeed, the smaller m, the smaller the chance of masking the occurrence of a feature. However, a larger s is then needed, since all features should have a high chance of being selected into many subsets of the features. For classification problems of dimension d ranging from several thousands to tens of thousands, we have found that taking m equal to a few hundred (say, $m = 300$ to 500) and t equal to maximum 20 (even $t = 5$ usually suffices) is a good choice in terms of reliability and overall computational cost of the procedure.

Now, for a given m, s can be made a running parameter of the procedure, and the procedure executed for $s = s_1, s_1 + 10, s_1 + 20, \ldots$ until the rankings of the top scoring ν% features prove (almost) the same for successive values of s.

Minimal number of subsets, s_1, should in fact be random and such that the ranking based on these subsets includes $\nu\%$ of all the features present in the full data sample.

A distance between two successive rankings has to be defined, and the procedure is then run until the values of the distance stabilize at some acceptably low level, i.e., close to zero. The distance between the ranking obtained after s subsets of m features have been used in the procedure and the ranking reached after using $s - 10$ subsets is defined as follows:

$$\mathrm{Dist}(s, s-10) = \frac{1}{d_\nu} \sum_{g_k} |\mathrm{rank}(g_k, s) - \mathrm{rank}(g_k, s-10)|, \tag{3}$$

where summation is over top $\nu\%$ features obtained after having used $s - 10$ subsets; $\mathrm{rank}(g_k, r)$ is the rank of feature g_k after having used r subsets, and d_ν is the normalizing constant equal to the number of features taken into account ($d_\nu = d\nu/100$). Parameter ν should not be too large and it is suggested that it lies between 5 and 20.

Note that ranking by relative importance does not enable one to discern between informative and uninformative features. A cut-off between these two types of features is needed. We address this issue by comparing the ranking of features obtained for the original data with that obtained for the data modified in such a way that the class attribute (label) becomes independent of the vector of all features. Such a data set is obtained via a random permutation of the values of the class attribute (i.e. of the class labels of the samples). We omit the details regarding this issue and refer the reader to Dramiński et al. (2010). A special statistical test is proposed there and, at a predetermined significance level, feature g_k is declared informative if its relative importance RI_{g_k} in the original ranking (without any permutation) exceeds the corresponding critical value for this test. However, in the reference cited, no recommendation is made concerning the desired significance level of the test. According to our experience, this level should be chosen adaptively, on the basis of classification results obtained for different significance levels. Clearly, the larger the significance level, the smaller the corresponding critical value and the greater number of features is deemed informative. We suggest using 0.05 as a default or an initial value of significance level, but then verify if increasing this level and thus enlarging the set of allegedly informative features does not lead to an improvement of classification results.

Let us conclude this section by adding that in Dramiński et al. (2008) we describe statistical tests to verify first, prior to the whole analysis, that the data are informative (i.e., that they indeed provide information on the classification problem of interest); then to verify that the features found as most informative are such indeed; and finally statistical significance of the results is confirmed.

3. Experimental study

Here we attempt to verify whether the MCFS algorithm is biased towards features with many categories (levels or values). In our study we build on the simulation study performed earlier by Strobl et al. (2007) for the original Breiman RFs as well as for modified RFs which differ from those of Breiman in that, most notably, CART trees are replaced by unbiased classification trees based on the conditional inference framework as developed by Hothorn et al. (2006). Actually, we repeat that earlier study, which was performed for samples of size 5, and extend it by a study with samples of size 50.

First, as in Strobl et al. (2007), let the data set consist of 120 independent samples (observations), each containing 5 features and a class or decision attribute (label); in this section, for greater clarity of figures which follow, we denote feature g_k by Xk, $k = 1, \ldots, d$. Values of the first feature, $X1$, are drawn at random from a standard normal distribution and the remaining four features, $X2, \ldots, X5$, are drawn from discrete uniform distributions, U{0, 1}, U{0, 1, 2, 3}, U{0, 1, 2, ..., 9} and U{0, 1, 2, ..., 19}, respectively, where U{0, ..., $p - 1$}, $p = 2, 4, 10, 20$, stands for a uniform distribution on p integers from 0 to $p - 1$. Values of the five features are drawn independently. Each observation belongs either to class 0 or to class 1. The class label is also drawn at random: if, for a given observation, $X2 = 0$, then the class of this observation is 0 with probability $0.5 + r$; if $X2 = 1$, then the class is 0 with probability $0.5 - r$; within one experiment, r, termed the relevance parameter, assumes a fixed value from the set {0, 0.05, 0.1, 0.15, 0.2}. For each $r \in \{0, 0.05, 0.1, 0.15, 0.2\}$, 1000 simulation runs are performed, i.e., 1000 data sets are drawn at random (each data set containing 120 labeled observations).

Clearly, for $r = 0$ no feature is informative and for $r > 0$ only one feature, $X2$, which assumes one of just two possible values, is informative. It is easily seen that in such an experiment the correlation coefficient between $X2$ and the class attribute is equal to $2r$ (regardless of the number of uninformative features).

The MCFS algorithm was run with the following parameters: $m = 3$, $s = 100$ and $t = 10$ (default value 1 was used for u and v). Note that for $d = 5$ and $m = 3$ there are only 10 different subsets of 3 out of 5 features, but we take $s = 100$ to add some more randomness to the whole ranking process (otherwise, we could run the algorithm for just 10 different subsets of features, but use $t = 100$).

In Figs. 2 and 3 mean values (over 1000 simulation runs) of normalized RI_{Xk}'s, $k = 1, \ldots, 5$ are shown for $r = 0, 0.05, 0.1, 0.2$, respectively (the whiskers on top of each bar show ± one standard deviation of the observed values of normalized relative importance). The results obtained suggest that there exists no bias of the MCFS algorithm towards features with many categories, at least for positive r (even mildly bounded away from zero). One should note here that, in view of the remarks at the end of the preceding section, we are equipped with

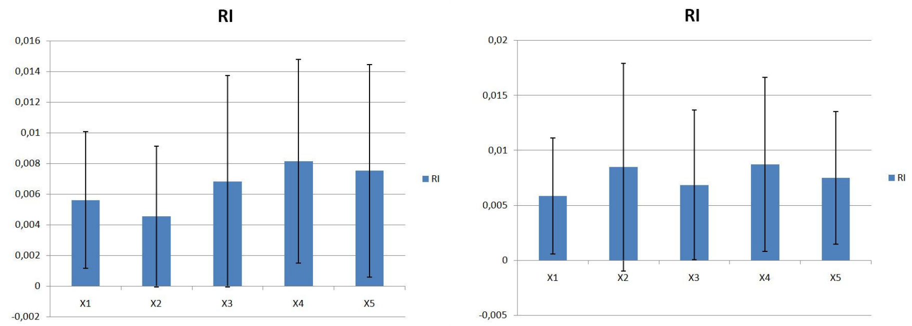

Figure 2. Mean values of normalized RI_{Xk}'s for $r = 0$ (left) and $r = 0.05$ (right); bar Xk corresponds to normalized RI_{Xk} (recall that the correlation coefficient between $X2$ and the class attribute is $2r$ and such correlations for all other features are 0)

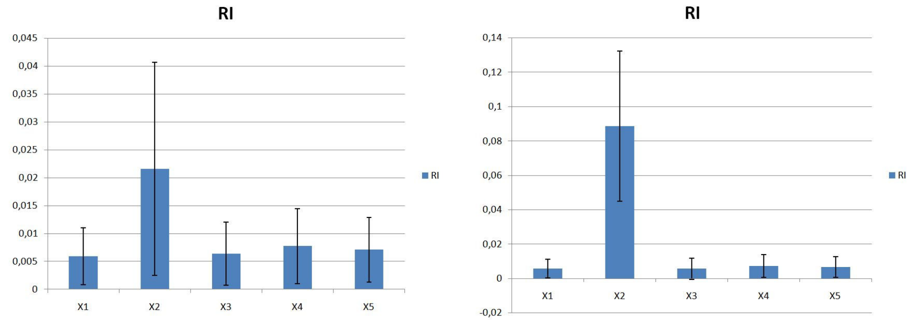

Figure 3. Mean values of normalized RI_{Xk}'s, $r = 0.1$ and $r = 0.2$; bar Xk corresponds to normalized RI_{Xk} (recall that the correlation coefficient between $X2$ and the class attribute is $2r$ and such correlations for all other features are 0)

a means to recognize that in a given problem there are no informative features and hence, under the circumstances, we should not make any ranking of features using the MCFS algorithm. Still, we have included the case with $r = 0$ to show that although the MCFS is not reliable then and some mild bias can perhaps be supposed, the values of normalized RI_{Xk} are almost one order of magnitude smaller from those when an informative feature does exist (to put it otherwise, the case with informative features present in the data can be readily recognized, and then the MCFS can be used and trusted).

It is instructive to compare the simulation results obtained with those from Strobl et al. (2007). For $r = 0$, the MCFS provides results markedly different from those obtained by Breiman's RF. Only the latter reveals an obvious bias towards features with many categories. In turn, for positive r, the MCFS provides results comparable to those for the modified RF, although the former is much simpler than the latter.

At the same time, the MCFS cannot be claimed to be completely unbiased, although the bias is of a more complex type and can best be described as that against features with the smallest number of categories. Its source can, perhaps, be tied to the way relative importance (2) is calculated. Most importantly, however, this bias is negligible whenever the relationship between the features and the decision attribute is not negligible.

Actually, the MCFS algorithm should not be used on any data with only five features. It has been designed to discover a few informative features among hundreds or, rather, thousands of noninformative ones. The experiments described have been performed for comparative purposes - to show that the algorithm performs reasonably well even on such simple data as used in the otherwise most revealing and thorough study of Strobl et al. (2007).

In order to get closer to problems of our real interest which concern first and foremost data from life sciences, but to somehow stay within the framework set up by Strobl et al. (2007), we have performed another set of experiments, which differs from the former one in just two respects. First, we draw randomly not 5 but 50 features, where each successive 10 features are drawn exactly in the same way as $X1$, $X2$, ..., $X5$ were drawn, respectively. That is, $X1, \ldots, X10$, are drawn independently from a standard normal distribution, $X11, \ldots, X20$ from U$\{0, 1\}$, etc. And second, we have utilized the fact that we have now more possibilities to make the observation class dependent on the features. Namely, we performed the following experiments which differ in the way the observation class is determined:

(a) If, for a given observation, $X11 = 0$, then the observation class is 0 with probability $0.5 + r$; if $X11 = 1$, then the observation class is 0 with probability $0.5 - r$.

(b) If, for a given observation, $X11 = X12 = 0$, then the observation class is 0 with probability $0.5 + r$; otherwise, the observation class is 0 with probability $0.5 - r$.

(c) If, for a given observation, $X11 = X12 = 0$ or $X13 = X14 = 0$, then the observation class is 0 with probability $0.5 + r$; otherwise, the observation class is 0 with probability $0.5 - r$.

In (a), the correlation coefficient between $X11$ and the class attribute is again equal to $2r$. It can be readily found that the correlation coefficient between any fixed informative feature and the class attribute is equal to r in case (b) and just $0.75r$ in case (c).

In these three experiments, the MCFS algorithm was run with the following parameters: $m = 30$, $s = 1000$ and $t = 10$ (as usual, default value 1 was used for u and v).

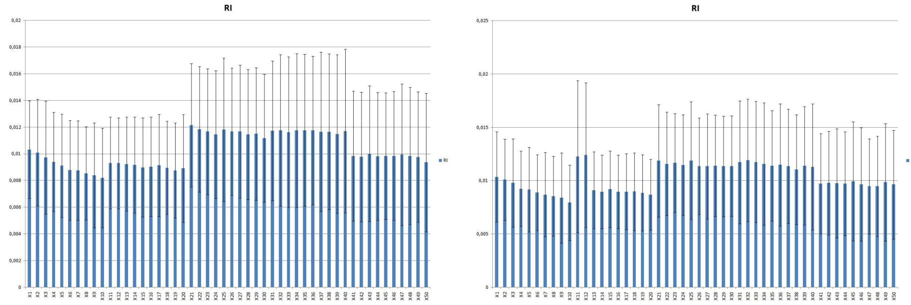

Figure 4. Mean values of normalized RI_{Xk}'s, $r = 0$ (left) and $r = 0.1$ (right) - case (b); successive bars correspond to normalized RI_{Xk}'s, $k = 1, \ldots, 50$ (recall that the only possibly informative features are $X11$ and $X12$, and their correlation coefficients with the class attribute are each equal to r)

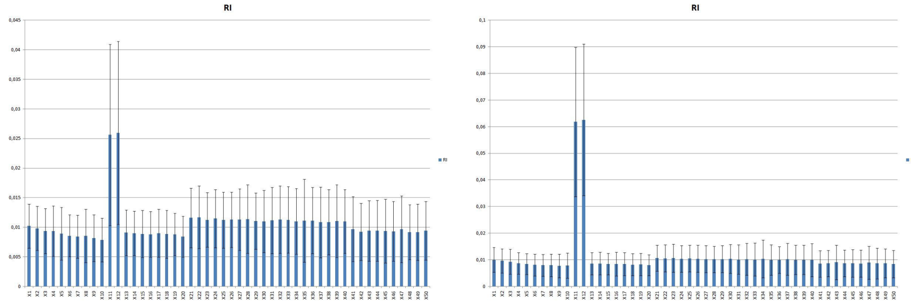

Figure 5. Mean values of normalized RI_{Xk}'s, $r = 0.2$ and $r = 0.3$ - case (b); successive bars correspond to normalized RI_{Xk}'s, $k = 1, \ldots, 50$ (recall that the only informative features are $X11$ and $X12$, and their correlation coefficients with the class attribute are each equal to r)

In Figs. 4 and 5 bar plots are given for case (b). The mean values (over 1000 simulation runs) of normalized RI_{Xk}'s, $k = 1, \ldots, 50$, are shown for $r = 0, 0.1, 0.2, 0.3$, respectively. Similarly, for case (c) and the same values of r, the mean values of normalized RI_{Xk}'s, $k = 1, \ldots, 50$, are shown in Figs. 6 and 7. Illustrations for case (a) are skipped, since they confirm the pattern, albeit in a much more apparent way, as the correlation between any single informative feature and the class attribute is much greater.

It is clear that the conclusions drawn for experiments with observations comprising just 5 features hold for the set of experiments with observations of size 50.

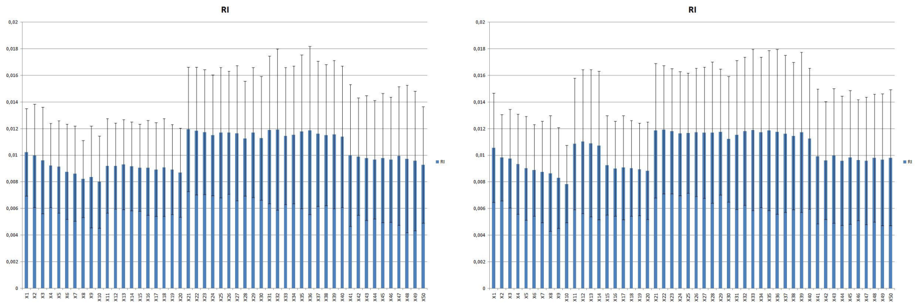

Figure 6. Mean values of normalized RI_{Xk}'s, $r = 0$ (left) and $r = 0.1$ (right) - case (c); successive bars correspond to normalized RI_{Xk}'s, $k = 1, \ldots, 50$ (recall that the only possibly informative features are $X11$ to $X14$, and their correlation coefficients with the class attribute are each equal to $0.75r$)

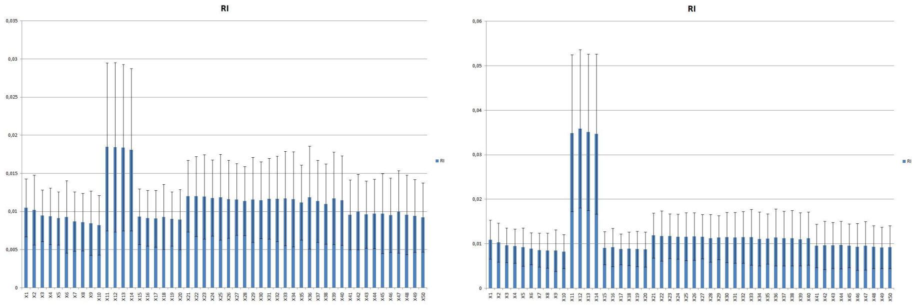

Figure 7. Mean values of normalized RI_{Xk}'s, $r = 0.2$ and $r = 0.3$ - case (c); successive bars correspond to normalized RI_{Xk}'s, $k = 1, \ldots, 50$ (recall that the only informative features are $X11$ to $X14$, and their correlation coefficients with the class attribute are each equal to $0.75r$)

4. Concluding remarks

It was already mentioned in the Introduction that the MCFS and MCFS-ID algorithms have been successfully used on many data sets. We proved being able to reproduce a number of previously obtained results. Working with one of these sets of data, we managed to detect genes that are weakly but sufficiently differentiated, linked to the onset of or a proclivity for the given type of cancer (such genes were rediscovered in addition to those expressing especially dramatic changes caused by cancer). As far as we are aware we were the first to obtain such a result by a purely computational method; see Dramiński et al. (2008). We were able to rediscover numerous mechanisms of HIV-1 drug-resistance and suggest several new mechanisms (stemming from interactions discovered) which should be further investigated; see Dramiński et al. (2010), Kierczak et al. (2009, 2010), Kierczak (2009).

Clearly, such rediscoveries confirm the reliability of the method and thus make new results provided by the method and not yet known to domain experts truly worth further study.

Our present study has shown that the reliability of our approach is not incidental, but is an inherent property of the MCFS and consequently also the MCFS-ID algorithm.

Indeed then, the approach appears a promising method to reveal and analyze interaction networks that underly biological and other phenomena which fall into the domain of supervised classification problems, in particular when important features must be identified among hundreds if not thousands of them. We therefore consider it likely that applications of this approach for unknown problems of systems biology will allow *in silico* discovery of new mechanisms that so far avoided human explanations.

References

Archer, K.J. and Kimes, R.V. (2008) Empirical Characterization of Random Forest Variable Importance Measures. *Comp Stat & Data Anal*, **52**(4), 2249-2260.

Breiman, L. and Cutler, A. (2008) *Random Forests - Classification/Clustering Manual.* http://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm

Chrysostomou, K., Chen, S.Y. and Liu, X. (2008) Combining Multiple Classifiers for Wrapper Feature Selection. *Int. J. Data Mining, Modelling and Management*, **1**, 91-102.

Diaz-Uriarte, R. and de Andres, S.A. (2006) Gene Selection and Classification of Microarray Data Using Random Forest. *BMC Bioinformatics*, **7**(3), doi:10.1186/1471-2105-7-3.

Dramiński, M., Rada-Iglesias, A., Enroth, S., Wadelius, C., Koronacki, J., Komorowski, J. (2008) Monte Carlo Feature Selection for Supervised Classification. *Bioinformatics*, **24**(1), 110-117.

Dramiński, M., Kierczak, M., Koronacki, J., Komorowski, J. (2010) Monte Carlo feature selection and interdependency discovery in supervised classification. In: J. Koronacki, Z.W. Ras, S.T. Wierzchon, J. Kacprzyk, eds., *Advances in Machine Learning*, vol. II, Springer, 371-385.

Dudoit, S. and Fridlyand, J. (2003) Classification in Microarray Experiments. In: T. Speed, ed., *Statistical Analysis of Gene Expression Microarray Data*, Chapman & Hall/CRC, 93-158.

Hothorn, T., Hornik, K. and Zeileis, A. (2006) Unbiased Recursive Partitioning: A Conditional Inference Framework. *J. Computational and Graphical Statistics*, **15**, 651-674.

Kierczak, M., Ginalski, K., Dramiński, M., Koronacki, J., Rudnicki, W. and Komorowski, J. (2009) A Rough Set-Based Model of HIV-1 Reverse Transcriptase Resistome. *Bioinformatics and Biology Insights*, **3**, 109-127. http://www.la-press.com/a-rough-set-based-model-of-hiv-1-reverse-transcriptase-resistome-a1685

Kierczak, M., Dramiński, M., Koronacki, J. and Komorowski, J. (2010) Computational analysis of molecular interaction networks underlying change of HIV-1 resistance to selected reverse transcriptase inhibitors. *Bioinformatics and Biology Insights* **4**, 137-146.
http://www.la-press.com/computational-analysis-of-molecular-interaction-networks-underlying-ch-article-a2395)

Kierczak, M. (2009) *From Physicochemical Properties to Interdependency Networks: A Monte Carlo Approach to Modeling HIV-1 Resistome and Post-translational Modifications*. PhD Thesis, Uppsala University (for an introduction to the thesis see Publications at http://www.kierczak.pl)

Li, Y., Campbell, C. and Tipping, M. (2002) Bayesian Automatic Relevance Determination Algorithms for Classifying Gene Expression data. *Bioinformatics*, **18**,(10), 1332-1339.

Lu, C., Devos, A., Suykens, J.A.K., Arus, C. and Van Huffel, S. (2007) Bagging Linear Sparse Bayesian Learning Models for Variable Selection in Cancer Diagnosis. *IEEE Trans Inf Technol Biomed*, **11**, 338-347.

Saeys, Y., Inza, I. and Larrañaga, P. (2007) A Review of Feature Selection Techniques in Bioinformatics. *Bioinformatics*, **23** (19), 2507-2517.

Strobl, C., Boulesteix, A.-L., Zeileis, A. and Hothorn, T. (2007) Bias in Random Forest Variable Importance Measures: Illustrations, Sources, and a Solution. *BMC Bioinformatics*, **8**(25), doi:10.1186/1471-2105-8-25.

Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T. and Zeileis, A. (2008) Conditional Variable Importance for Random Forests. *BMC Bioinformatics*, **9**(307), doi:10.1186/1471-2105-9-307.

Tibshirani, R., Hastie, T., Narasimhan, B. and Chu, G. (2002) Diagnosis of multiple cancer types by nearest shrunken centroids of gene expressions. *Proc Natl Acad Sci USA*, **99**, 6567-6572.

Tibshirani, R., Hastie, T., Narasimhan, B. and Chu, G. (2003) Class Prediction by Nearest Shrunken Centroids, with Applications to DNA Microarrays. *Statistical Science*, **18**, 104-117.