Monte Carlo feature selection and interdependency discovery in supervised classification

Michał Damiński¹, Marcin Kierczak¹, Jacek Koronacki and Jan Komorowski

Abstract Applications of machine learning techniques in Life Sciences are the main applications forcing a paradigm shift in the way these techniques are used. Rather than obtaining the best possible supervised classifier, the Life Scientist needs to know which features contribute best to classifying observations into distinct classes and what are the interdependencies between the features. To this end we significantly extend our earlier work [Damiński *et al.* (2008)] that introduced an effective and reliable method for ranking features according to their importance for classification. We begin with adding a method for finding a cut-off between informative and non-informative features and then continue with a development of a methodology and an implementation of a procedure for determining interdependencies between informative features. The reliability of our approach rests on multiple construction of tree classifiers. Essentially, each classifier is trained on a randomly chosen subset of the original data using only a fraction of all of the observed features. This approach is conceptually simple yet computer-intensive. The methodology is validated on a large and difficult task of modelling HIV-1 reverse transcriptase resistance to drugs which is a good example of the aforementioned paradigm shift. We construct a classifier but of the main interest is the identification of mutation points (i.e. features) and their combinations that model drug resistance.

Michał Damiński and Jacek Koronacki

Institute of Computer Science, Polish Acad. Sci., Ordona 21, Warsaw, Poland, e-mail: Michal.Daminski@ipipan.waw.pl, Jacek.Koronacki@ipipan.waw.pl

Marcin Kierczak

The Linnaeus Centre for Bioinformatics, Uppsala University and The Swedish University of Agricultural Sciences, Box 758, Uppsala, Sweden, e-mail: Marcin.Kierczak@lcb.uu.se

Jan Komorowski

The Linnaeus Centre for Bioinformatics, Uppsala University and The Swedish University of Agricultural Sciences, Box 758, Uppsala, Sweden
Interdisciplinary Centre for Mathematical and Computer Modelling, Warsaw University, Poland, e-mail: Jan.Komorowski@lcb.uu.se

¹ These authors contributed equally.

1 Introduction

A major challenge in the analysis of many data sets, especially those presently generated by advanced biotechnologies, is their size: a very small number of records (samples, observations), of the order of tens, versus thousands of attributes or features per each record. Typical examples include microarray gene expression experiments (where the features are gene expression levels) or data coming from next generation DNA or RNA sequencing projects. Another obvious example are transactional data of commercial origin. In all these tasks supervised classification is quite different from a typical data mining problem in which every class has a large number of examples. In the latter context, the main task is to propose a classifier of the highest possible quality of classification. In class prediction for typical gene expression data it is not the classifier *per se* that is crucial; rather, selection of informative (discriminative) features and the discovered interdependencies among them are to give the Life Scientists a much desired possibility of the interpretation of the classification results.

Given such data, all reasonable classifiers can be claimed to be capable of providing essentially similar results (if measured by error rate or the like criteria; cf. [Dudoit and Fridlyand (2003)]). However, since it is rather a rule than an exception that most features in the data are *not* informative, it is indeed of utmost interest to select the few ones that are informative and that may form the bases for class prediction. Equally interesting is a discovery of interdependencies between the informative features.

Generally speaking, feature selection may be performed either prior to building the classifier, or as an inherent part of this process. These two approaches are referred to as *filter* methods and *wrapper* methods, respectively. Currently, the wrapper methods are often divided into two subclasses: one retaining the name "wrapper methods" and the other, termed *embedded* methods. Within this finer taxonomy, the former refer to such classification methods in which feature selection is "wrapped" around the classifier construction and the latter one to those in which feature selection is directly built into the classifier construction.

A significant progress in these areas 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)]. Regarding the wrapper and embedded approaches, an early successful method, not mentioned by [Saeys *et al.* (2007)], was developed by Tibshirani *et al.* (see [Tibshirani *et al.* (2002), Tibshirani *et al.* (2003)]) and is called nearest shrunken centroids. Most 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 *et 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, he proposed to make use of the so-called variable (i.e. feature) importance for feature selection. Determination of the importance of the variable is not necessary for random forest

construction, but it is a subroutine performed in parallel to building the forest; cf. [Breiman and Cutler (2008)]. Ranking features by variable importance can thus be considered to be a by-product of building the classifier. While ranking variables according to their importance is a natural basis for a filter, nothing prevents one from using such importances within, say, the embedded approach; cf., e.g., [Diaz-Uriarte and de Andres (2006)]. In any case, feature selection by measuring variable importance in random forests should be seen as a very promising method, albeit under one proviso. Namely, the problem with variable importance as originally defined is that it is biased towards variables with many categories and variables that are correlated; cf. [Strobl *et al.* (2007), Archer and Kimes (2008)]. Accordingly, proper debiasing is needed, in order to obtain true ranking of features; cf. [Strobl *et al.* (2008)].

One potential advantage of the filter approach is that it constructs a group of features that contribute the most to the classification task, and therefore are informative or "relatively important", to a given classification 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 random forests possess the last capability.

Recently, a novel, effective and reliable 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 is capable of incorporating interdependencies between features. It bears some remote similarity to the Random Forest methodology, but differs entirely in the way features ranking is performed. Specifically, our method does not require debiasing and is conceptually simpler. A more important and new result is that it provides explicit information about interdependencies among features. Within our approach, discovering interdependencies builds on identifying features which "cooperate" in determining that samples belong to the same classes. It is worthwhile to emphasize that this is completely different from the usual approach which aims at finding features that are similar in some sense.

The procedure from [Dramiński *et al.* (2008)] for Monte Carlo feature selection is briefly recapitulated in Section 2. Since the original aim was only to rank features according to their classification ability, no distinction was made among informative and non-informative features. In that section we introduce an additional procedure to find a cut-off separating informative from non-informative features in the ranking list. In Section 3, a way to discover interdependencies between features is provided. In Section 4 application of the method is illustrated on the HIV-1 resistance to Didanosine. Interpretation of the obtained results is provided in Subsection 4.1. We close with concluding remarks in Section 5.

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

2 Monte Carlo Feature Selection

The Monte Carlo feature selection (MCFS) part of the algorithm is conceptually simple, albeit 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. The use of classification trees is motivated by the fact that they can be considered to be the most flexible classifiers within the family of all classification methods. In our method, however, the classifiers are used for measuring relative importance of features, not for classification *per se*.

In the main step of the procedure, we estimate relative importance of features by constructing thousands of trees for randomly selected subsets of the features. More precisely, out of all d features, we select s subsets of m features, m being fixed and $m \ll d$, 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 so that 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 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, we 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, $i, j = 1, 2, \dots, c$ and $\sum_{i,j} n_{ij} = n$, the number of all samples. Now, we define weighted accuracy as

$$wAcc = \frac{1}{c} \sum_{i=1}^c \frac{n_{ii}}{n_{i1} + n_{i2} + \dots + n_{ic}}, \quad (1)$$

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

Further, if a particular split is made on feature g_k , then the more informative this feature is, the greater is $wAcc$ for the whole tree. Similarly, both the information gain on the split and the number of samples in the split node are greater. Information gain can be measured, e.g., by Gini Index or Gain Ratio, and the relative importance of feature g_k , RI_{g_k} , can be defined as

$$RI_{g_k} = \sum_{\tau=1}^{st} (wAcc)^u \sum_{n_{g_k}(\tau)} IG(n_{g_k}(\tau)) \left(\frac{\text{no. in } n_{g_k}(\tau)}{\text{no. in } \tau} \right)^v, \quad (2)$$

where summation is over all the $s \cdot t$ trees and, within each τ -th tree, overall nodes $n_{g_k}(\tau)$ of the tree on which the split is made on feature g_k , $IG(n_{g_k}(\tau))$ stands for information gain for node $n_{g_k}(\tau)$, $(\text{no. in } n_{g_k}(\tau))$ denotes the number of samples in node $n_{g_k}(\tau)$, $(\text{no. in } \tau)$ denotes the number of samples in the root of the τ -th tree, and u and v are fixed positive reals.

Note that by taking, say, $u = 2$, trees with low $wAcc$ are penalized more severely than when taking $u = 1$. Similarly, the greater is v , the smaller is the influence of node $n_{g_k}(\tau)$ with a given ratio $(\text{no. in } n_{g_k}(\tau))/(\text{no. in } \tau)$ 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.

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 hundreds (say, $m = 300 - 500$) 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 $s = s_1, s_1 + 10, s_1 + 20, \dots$ until the rankings for successive values of the s top scoring $p\%$ features prove (almost) the same. Minimal number of subsets, s_1 , should in fact be random and such that the ranking based on these subsets includes $p\%$ of all the features present in the full data sample. Note that after having used s subsets of m features, at most $s \cdot m$ features can be ranked, and the probability of achieving this upper bound practically equals zero. More precisely, a distance between two successive rankings is defined, and the procedure is 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:

$$\text{Dist}(s, s - 10) = \frac{1}{d_p} \sum_{g_k} |\text{rank}(g_k, s) - \text{rank}(g_k, s - 10)|, \quad (3)$$

where summation is over top $p\%$ features obtained after having used $s - 10$ subsets; $\text{rank}(g_k, r)$ is the rank of feature g_k after having used r subsets, and d_p is the normalizing constant equal to the number of features taken into account ($d_p = dp/100$).

Note that the number of features ranked accordingly to their relative importance, d_p , should not be too large. Indeed, if d_p is such that only, say, $d_p/10$ features are informative, convergence of the distance function given by (3) to some positive value as s increases will be slow and this distance, as a function of s , will show a persistently oscillatory behavior. A moment of thought suffices to realize, as was amply confirmed by simulations, that distance (3) converges to some positive value, however erratically, even if all features are non-informative, i.e. the vector of all features is independent of the decision or class attribute.) Summarizing, while d_p should be such that all informative features are included in the ranking, distance (3) should not only converge but it should approach its asymptote smoothly as s increases (the latter property can be considered as an indication that the convergence is possibly fast).

The above calls for a means to discern between informative and non-informative features, as measured by their relative importance. Only in this way we can learn that all informative features indeed have been found and ranked. We address this issue by comparing the ranking obtained for the original data with the one 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). Note that this modification does not change the multivariate distribution of feature vectors.

Let Π such permutations be made and thus Π modified data sets with no relationship between the features and the class attribute be obtained. For each π -th data set, $\pi = 1, 2, \dots, \Pi$, rank the features according to their relative importance (2) and retain the maximal value of importance (2) among all the features. This value will be denoted $\max\text{RI}_\pi$. Obviously, these $\max\text{RI}_\pi$ values are random. It follows from our experiments performed on many different data sets that $\max\text{RI}_\pi$ may be considered

normally distributed (if judged, e.g., by the Shapiro-Wilk test). In all experiments, it was sufficient to set Π to 30. If this is the case, then the way to find statistically significant, that is informative, features in the original data set is straightforward. Since $\max \text{RI}_\pi$ has, after proper normalization, Student's t-distribution with $\Pi - 1$ degrees of freedom, it suffices to provide a critical value for the one-sided Student's t-test at any given significance level. A feature g_k is declared informative (at a given significance level) if its relative importance RI_{g_k} in the original ranking (without any permutation) exceeds the desired critical value.

For the two data sets that we analyzed in [Dramiński *et al.* (2008)], namely the leukemia data of [Golub *et al.* (1999)] and the second one: lymphoma data of [Alizadeh *et al.* (2000)], we found that, at significance level of 0.05, there are 22 informative features in the first data set and 50 informative features in the second data set. Concluding this section we should note that the above significance analysis does not tell anything about the validity of the ranking itself. However, in [Dramiński *et al.* (2008)] we describe in detail additional steps of the MCFS procedure that allow appraising the statistical significance of the resulting ranking of features.

3 Discovering feature interdependencies

Ranking features by the MCFS procedure is an efficient method of selecting the set of informative (discriminative) features. The other natural issue to be raised concerns possible interdependencies among features. Likewise in experimental design and analysis of variance, these interdependencies are often modeled using interactions. Perhaps the most widely used approach to recognizing interdependencies is finding correlations between features or finding groups of features that behave in some sense similarly across samples. A typical bioinformatics example of this problem is finding co-regulated features, most often genes or, rather more precisely, their expression profiles. Searching groups of similar features is usually done with the help of various clustering techniques, frequently specially tailored to a task at hand. See [Smyth *et al.* (2003), Hastie *et al.* (2001), Saeys *et al.* (2007), Gyenesei, A. *et al.* (2007)] and the literature there.

Our approach to interdependency discovery (abbreviated MCFSID) is significantly different in that we focus on identifying features that "cooperate" in determining that a sample belongs to a particular class. To be more specific, assume that for a given a training set of samples, a rule-based classifier, a decision or a classification tree, has been constructed. Now, for each class, a set of decision rules defining this class is provided. Each decision rule is in fact a conjunction of conditions imposed on particular separate features and, thus, points to some interdependencies between the features appearing in the conditions.

The given idea of looking at interdependencies among features seems to be rather plausible but we need to refine it so that it is not dependent on just one classifier. Given a single classifier, our trust in the decision rules that are learned, and thus

Fig. 2 An example of a tree graph.

in the discovered interdependencies, is naturally limited by the predictive ability of that classifier. Even more importantly, the classifier is trained on just one training set and the final set of rules is dependent on the classifier. Therefore, our conclusions are necessarily dependent on the classifier and are conditional upon the training set, since they follow from just one solution of the classification problem. In the case of decisions trees, the problem is aggravated by their high variance, i.e., their tendency to provide varying results even for slightly different training sets. It should now be obvious, however, that the way out of the trouble is through an aggregation of the information provided by all the $s \cdot t$ trees, which anyhow are built within the MCFS part of the MCFSID algorithm.

In each of the $s \cdot t$ trees the nodes represent features; i.e. a given node represents the feature on which the split is made. For each path in a tree, we define the distance between two nodes as the number of edges between these two nodes. For example, in Fig. 2 the distance between node n_1 and n_5 is equal to 2 and between n_1 and n_9 is equal to 3. In turn, one may define the strength of the interdependence between features g_i and g_j as

$$\text{Dep}(g_i, g_j) = \sum_{\tau=1}^{st} \sum_{\xi_\tau} \sum_{n_{g_i}(\xi_\tau), n_{g_j}(\xi_\tau)} \frac{1}{\text{dist}(n_{g_i}(\xi_\tau), n_{g_j}(\xi_\tau))}, \quad (4)$$

where summation is over all the $s \cdot t$ trees, within each τ -th tree over all paths ξ_τ and, within each path ξ_τ , over all pairs of nodes $(n_{g_i}(\xi_\tau), n_{g_j}(\xi_\tau))$ on which the splits are made, respectively, on feature g_i and feature g_j ; $\text{Dep}(g_i, g_j) = 0$ if in none of the trees there is a path along which there are two splits made, respectively, on g_i and g_j . The rationale behind this definition is obvious. $\text{Dep}(g_i, g_j)$, calculated on the basis of thousands of trees provides an incomparably more stable and reliable information about the strength of interdependency between two features than that derived from just one single tree.

Analysis of all pairs of nodes for all decision trees is a time consuming task. To deal with this problem we introduce a parameter k that determines a maximum distance between two nodes to be taken into account in (4). By default, we set k to 3. Final results of the analysis may be presented in the form of a graph with nodes representing features and edges showing strength of dependence among the connected

nodes as measured by $\text{Dep}(g_i, g_j)$. Only the nodes corresponding to the informative features should be included in the graph. If the graph proves hardly readable due to a large number of nodes and edges, it can be easily simplified by showing only these edges that represent a desired fraction of the strongest interdependencies. Typically, we show 20% of all the edges found for a selected set of features. Despite the obvious advantages of the presented approach, it should be kept in mind that it is a heuristic and, in particular, definition (4) is arbitrary. The real value of the presented method can be assessed by tests only. This can be achieved by examining a number of examples where the resulting graphs are scrutinized by domain experts who are able to verify and explain (or falsify) the claimed interdependencies.

4 Biological validation study

In one of our applications, we constructed classifiers for modeling drug resistance. We used publicly available HIV-1 drug resistance data from the Stanford HIV Resistance Database [HIV Resistance Database]. Using our domain knowledge, we evaluated the interdependencies discovered with the MCFSID methodology.

The data set consists of several protein sequences of the HIV-1 reverse transcriptase (RT) enzyme annotated with their drug resistance fold relative to the wild-type HIV-1 strain. Following [Rhee *et al.* (2006)], we have discretized these continuous resistance values into three classes: "susceptible", "intermediately resistant" and "resistant". These classes are commonly used by clinicians while deciding upon the treatment regimen. Every amino acid in the sequence is subsequently represented as a 7-vector of its biochemical properties (values relative to the wild-type virus). After [Rudnicki and Komorowski (2004)], we have used the following set of biochemical descriptors:

- D1 - Transfer free energy from octanol to water.
- D2 - Normalized van der Waals volume.
- D3 - Isoelectric point.
- D4 - Polarity.
- D5 - Normalized frequency of turn.
- D6 - Normalized frequency of alpha-helix.
- D7 - Free energy of solution in water.

In the sequel, we present and discuss the results that were obtained for Didanosine, a nucleoside analogue of adenosine, and a commonly administered anti-viral drug. Our results for the 6 other anti-viral drugs: Abacavir, Efavirenz, Nevirapine, Stavudine, Tenofovir and Zidovudine constitute a large problem that will appear in a separate paper.

Didanosine belongs to the so-called nucleoside RT inhibitors (NRTI) that mimic nucleosides that are natural substrates for the enzyme. Once inside the cell, the nucleoside analogs are phosphorylated to their triphosphate (TP) form and, as soon

Fig. 3 Interdependencies graph for Didanosine important features.

as they are incorporated into a newly synthesized HIV-1 DNA chain, they terminate its elongation and, thus, inhibit viral replication [Jonckheere, H. *et al.* (2000), Bauman, JD *et al.* (2008), Menéndez-Arias, L. (2008)].

In this 3-class classification task, there were 706 samples, each described by 3920 features. We recall that there are coming from 560 amino acids in RT. Each aa is represented by 7 easy-to-interpret, low-correlated biochemical properties. The following parameters have been used in the MCFS part of the MCFSID algorithm: $s = 2500$, $t = 5$, $m = 0.05 \cdot 3920 = 196$, $u = 1$ and $v = 1$. The resulting graph of interdependencies is given in Fig. 3. The darker is the label of a graph node, the higher is the RI value of the corresponding feature. Following the default rule, only 20% of the strongest interdependencies are shown in the graph. This graph was created by a JAVA application called *graphViewer* that uses *jgraph* library [JGraph].

As it can be seen in the graph 3, there are 5 different biochemical properties were most frequently used by the MCFSID in constructing the trees.

4.1 Interpretation of the interdependencies

Reverse Transcriptase is a viral enzyme responsible for transcribing single-stranded HIV-1 RNA genome into a double-stranded DNA provirus. In the course of infection, the provirus is incorporated into the host cell genome and may be replicated and transcribed by the native molecular machinery of the host. This results in new viral particles that are capable of infecting subsequent cells and thus complete the HIV life cycle [Jonckheere, H. *et al.* (2000), Menéndez-Arias, L. (2008)].

Fig. 4 Structure of the HIV-1 RT enzyme fragment (PDB structure 1RTD). Thumb domain (red), finger domains (yellow) and palm domain (green) constitute p66 subunit. Residues constituting dNTP binding pocket are shown in magenta, YXDD motif members are within the red circle. The incoming dNTP and two magnesium ions are shown in light green. (For clarity, the structure to the right is rotated 180°).

The entire RT enzyme is a heterodimer consisting of two similar subunits: p51 (51 kDa) and p66 (66 kDa). The heavier p66 subunit contains two distinct domains: the N-terminal polymerase domain and the C-terminal RNase H domain. The lighter p51 is a product of the p66 proteolytic cleavage. It lacks RNase H domain and catalytic activity. Often, the ternary structure of the p66 domain is compared to the right hand and, accordingly, the palm, thumb and finger domains are distinguished (see Fig. 4). The palm domain contains three catalytically active re-si-du-es: Asp 110, Asp 185 and Asp 186 embedded in a hydrophobic region; cf. [Valverde-Garduño *et al.* (1998), Menéndez-Arias, L. (2008), Ren, J. and Stammers, DK. (2008)]. The two latter residues, together with Tyr 183 and Met 184, constitute a highly conserved YXDD motif that is common to all reverse transcriptases ([Kaushik *et al.* (1996)]).

The fingers and both alpha-helices of the thumb are thought to form a clamp that holds the nucleic acid in place over the palm part that is required for polymerization. The template/primer interactions occur between the sugar-phosphate backbone of the DNA/RNA and p66 residues [Sarafianos, SG. *et al.* (1999)]. Apart from the catalytic triad, the palm domain contains a dNTP-binding pocket; cf. Fig. 4. As polymerization proceeds, new deoxyribonucleotide-triphosphates (dNTPs) come to the binding pocket and are incorporated into a newly synthesized provirus DNA strand. The catalytic triad plays crucial role in this process by interacting with magnesium cations that serve as activators of the enzymatic reaction. Mutagenesis studies described in [Valverde-Garduño *et al.* (1998), Harris, D. *et al.* (1998),

Fig. 5 Structure of the HIV-1 RT enzyme p66 subunit. Alpha-atoms of the residues selected by MCFSID as contributing to resistance to Didanosine represented as red spheres. The incoming dNTP and two magnesium ions shown in light green. (For clarity, the structure to the right is rotated 180°).

Bauman, JD *et al.* (2008)] suggest that mutations surrounding the catalytic triad may lead to incorporation of non-specific nucleotides.

Our results for Didanosine are presented in Fig. 5. We show alpha-atoms of the residues selected as contributing to resistance to the drug.

The algorithm identified 5 biochemical properties that span over 6 aa residues. Both Fig. 6, graph 1 and Fig. 6, graph 4 contain site 184 and either of the 215 or the 218 sites. The two latter sites are both located on the side of the dNTP binding pocket opposite to residue 184. While the amino acids at positions 184 and 215 are known to be directly interacting with the incoming nucleotide, Asp 218 interacts directly with Lys 219 that is a part of the dNTP binding pocket; cf. [Harris, D. *et al.* (1998), Menéndez-Arias, L. (2008)]. "Polarity" has been selected as important at site 218, "free energy of solution in water" at site 215 and both "normalized van der Waals volume" and "free energy of solution in water" at site 184. While mutations at sites 184 and 215 may directly disturb properties of the dNTP binding pocket, mutations at site 218 may affect its geometry in an indirect way.

Pairs involving residues 41, 74 and 75 (Fig. 6, graphs: 2 and 3) appear to be good descriptors of the fingers domain properties. The fingers domain is known to be involved in the viral RNA template positioning; cf. [Sarafianos, SG. *et al.* (1999), Menéndez-Arias, L. (2008)].

We conclude that two distinct resistance mechanisms characterize Didanosine: one is induced by mutations directly affecting the dNTP binding pocket and the other is caused by changes in the template positioning.

Fig. 6 Block diagram of the interdependencies between aminoacids descriptors of the didanosine. Shape corresponds to property, color to site number.

5 Concluding remarks

The MCFSID algorithm provides an effective and reliable method for ranking features according to their importance in supervised classification. It achieves that by determining the set of informative features as contrasted with the remaining non-informative ones, and by discovering interdependencies between the informative features that jointly are instrumental in determining which samples belong to their particular classes and to what degree.

Reliability of the algorithm follows from the Monte Carlo approach; resampling is sufficiently numerous and extensive, and the tree classifier amply flexible. It also

is a consequence of (i) the way in which we choose the number s of subsets of features while other parameters remain fixed (here we mean the requirement that the distance (3) between successive rankings stabilizes at some acceptably low level); and (ii) the way in which the strength of dependence among features is defined. It should be emphasized that the algorithm has been designed specifically to rank features with respect to their classification ability, not only to find those features that are important for classification.

Effectiveness and reliability of the algorithm has been confirmed on many data sets of biological and commercial origin. In addition to the HIV-1 case, we used the MCFSIF approach to identify interdependent features important to post-translational modifications of proteins. Our analyses also included transactional data from a major multinational FMCG company, geological data from oil wells operated by a major American company and data sets of samples with attributes being some functions of U.S. Census data. At the moment of completing this paper the MCFS part of the algorithm has been successfully used on some 15 data sets and the whole MCFSID algorithm on about 10.

In our opinion, the MCFSID approach appears to be quite promising in the area of systems biology where important features must be identified among hundreds if not thousands of them. In the case of HIV-1 we were able to automatically rediscover all the parameters that previously were identified by human experts as the important ones and thus rediscover known mechanisms of drug resistance. It is likely that applications of this approach for unknown problems will allow *in silico* discovery of new mechanisms that so far avoided human explanations.

References

- [Alizadeh *et al.* (2000)] Alizadeh, A.A. *et al.* (2000) Distinct Types of Diffuse Large B-cell Lymphoma Identified by Expression Profiling. *Nature* 403:503-511.
- [Archer and Kimes (2008)] Archer, K.J. and Kimes, R.V. (2008) Empirical Characterization of Random Forest Variable Importance Measures. *Comp Stat & Data Anal*, 52(4): 2249-2260.
- [Bauman, JD *et al.* (2008)] Bauman, JD. *et al.* (2008) Crystal engineering of HIV-1 reverse transcriptase for structure-based drug design. *Nucleic Acid Res*, 36:5083-5092.
- [Breiman and Cutler (2008)] Breiman, L. and Cutler, A. (2008) Random Forests - Classification/Clustering Manual. http://www.math.usu.edu/~adele/forests/cc/_home.htm
- [Chrysostomou *et al.* (2008)] Chrysostomou, K. *et al.* (2008) Combining Multiple Classifiers for Wrapper Feature Selection. *Int J Data Mining, Modelling and Management*, 1: 91-102.
- [Diaz-Uriarte and de Andres (2006)] 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.
- [Damiński *et al.* (2008)] Damiński, M. *et al.* (2008) Monte Carlo Feature Selection for Supervised Classification. *Bioinformatics*, 24(1):110-117.
- [Dudoit and Fridlyand (2003)] Dudoit, S. and Fridlyand, J. (2003) Classification in Microarray Experiments. In: *Statistical Analysis of Gene Expression Microarray Data*, T. Speed (ed.), Chapman & Hall/CRC, 93-158.
- [Golub *et al.* (1999)] Golub, T.R. *et al.* (1999) Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. *Science* 286:531-537.

- [Gyenesei, A. *et al.* (2007)] Gyenesei, A. *et al.* (2007) Mining Co-regulated Gene Profiles for the Detection of Functional Associations in Gene Expression data. *Bioinformatics*, 23(15):1927-1935.
- [Harris, D. *et al.* (1998)] Harris, D. *et al.* (1998) Functional analysis of amino acid residues constituting the dNTP binding pocket of HIV-1 reverse transcriptase. *J Biol Chem*, 273:33624-33634.
- [Hastie *et al.* (2001)] Hastie, T. *et al.* (2001) Supervised Harvesting of Expression Trees. *Genome Biology*, 2(1):research0003.1-0003.12.
- [HIV Resistance Database] Stanford HIV Drug Resistance Database, <http://hivdb.stanford.edu>
- [JGraph] JGraph - The Java Open Source Graph Drawing Component, <http://www.jgraph.com/jgraph.html>
- [Jonckheere, H. *et al.* (2000)] Jonckheere, H. *et al.* (2000) The HIV-1 reverse transcription (RT) process as target for RT inhibitors. *Med Res Rev*, 20:129-154.
- [Kaushik *et al.* (1996)] Kaushik *et al.* (1996) Biochemical analysis of catalytically crucial aspartate mutants of human immunodeficiency virus type 1 reverse transcriptase. *Biochemistry* 35(36):11536-11546.
- [Li *et al.* (2002)] Li, Y. *et al.* (2002) Bayesian Automatic Relevance Determination Algorithms for Classifying Gene Expression data. *Bioinformatics*, 18 (10): 1332-1339.
- [Lu *et al.* (2007)] Lu, C. *et al.* (2007) Bagging Linear Sparse Bayesian Learning Models for Variable Selection in Cancer Diagnosis. *IEEE Trans Inf Technol Biomed*, 11: 338-347.
- [Menéndez-Arias, L. (2008)] Menéndez-Arias, L. (2008) Mechanisms of resistance to nucleoside analogue inhibitors of HIV-1 reverse transcriptase. *Virus Res*, 134:124-146.
- [Ren, J. and Stammers, DK. (2008)] Ren, J. and Stammers, DK. (2008) Structural basis for drug resistance mechanisms for non-nucleoside inhibitors of HIV reverse transcriptase. *Virus Res*, 134:157-170.
- [Rhee *et al.* (2006)] Rhee, SY /it *et al.* (2006) Genotypic predictors of human immunodeficiency virus type 1 drug resistance., *Proc Natl Acad Sci USA*, 103: 17355-17360.
- [Rudnicki and Komorowski (2004)] Rudnicki W.R. and Komorowski J., (2004) Feature synthesis and extraction for the construction of generalized properties of amino acids. In: *Rough Sets and Current Trends in Computing*, S. Tsumoto, R. Słowiński and J. Komorowski (eds.), Lecture Notes in Computer Science 3066/2004, Springer, 786-791.
- [Saeys *et al.* (2007)] Saeys, Y. *et al.* (2007) A Review of Feature Selection Techniques in Bioinformatics. *Bioinformatics*, 23(19):2507-2517.
- [Sarafianos, SG. *et al.* (1999)] Sarafianos, SG. *et al.* (1999) Touching the heart of HIV-1 drug resistance: the fingers close down on the dNTP at the polymerase active site. *Chem & Biol*, 6:R137-R146.
- [Smyth *et al.* (2003)] Smyth, G.K. *et al.* (2003) Statistical Issues in cDNA Microarray Data Analysis. In: *Functional Genomics: Methods and Protocols*, M.J. Brownstein and A.B. Khodursky (eds.), Methods in Molecular Biology, vol. 224, Humana Press, 111-136.
- [Strobl *et al.* (2007)] Strobl, C. *et al.* (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 *et al.* (2008)] Strobl, C. *et al.* (2008) Conditional Variable Importance for Random Forests. *BMC Bioinformatics*, 9(307), doi:10.1186/1471-2105-9-307.
- [Tibshirani *et al.* (2002)] Tibshirani, R. *et al.* (2002) Diagnosis of Multiple Cancer Types by Nearest Shrunken Centroids of Gene Expressions. *Proc Natl Acad Sci USA*, 99: 6567-6572.
- [Tibshirani *et al.* (2003)] Tibshirani, R. *et al.* (2003) Class Prediction by Nearest Shrunken Centroids, with Applications to DNA Microarrays. *Statistical Science*, 18:104-117.
- [Valverde-Garduño *et al.* (1998)] Valverde-Garduño *et al.* (1998) Functional analysis of HIV-1 reverse transcriptase motif C: site-directed mutagenesis and metal cation interaction. *J Mol Evol*, 47: 73-80.
- [Yousef *et al.* (2007)] Yousef, M. *et al.* (2007) Recursive Cluster Elimination (RCE) for Classification and Feature Selection from Gene Expression Data. *BMC Bioinformatics*, 8(144), doi:10.1186/1471-2105-8-144.