Boruta – A System for Feature Selection

ARTICLE *in* FUNDAMENTA INFORMATICAE · JANUARY 2010

Impact Factor: 0.72 · DOI: 10.3233/FI-2010-288 · Source: DBLP

3 AUTHORS, INCLUDING:

Aleksander Jankowski
European Molecular Biology Laboratory
6 PUBLICATIONS **44** CITATIONS
SEE PROFILE

Witold Rudnicki
University of Bialystok
52 PUBLICATIONS **593** CITATIONS
SEE PROFILE

Fundamenta Informaticae 101 (2010) 271–285
DOI 10.3233/FI-2010-288
IOS Press

Boruta – A System for Feature Selection

Miron B. Kursa*, **Aleksander Jankowski, Witold R. Rudnicki**
ICM, University of Warsaw
Pawińskiego 5a, Warsaw, Poland
W.Rudnicki@icm.edu.pl

Abstract. Machine learning methods are often used to classify objects described by hundreds of attributes; in many applications of this kind a great fraction of attributes may be totally irrelevant to the classification problem. Even more, usually one cannot decide a priori which attributes are relevant. In this paper we present an improved version of the algorithm for identification of the full set of truly important variables in an information system. It is an extension of the random forest method which utilises the importance measure generated by the original algorithm. It compares, in the iterative fashion, the importances of original attributes with importances of their randomised copies. We analyse performance of the algorithm on several examples of synthetic data, as well as on a biologically important problem, namely on identification of the sequence motifs that are important for aptameric activity of short RNA sequences.

1. Introduction

In many cases the goal of a biological experiment is to discern between two classes, basing on a results of some test. For example a clinical study where identical gene expression tests may be performed on control group of healthy subjects along with the group of diseased patients. Identification of genes expressed differently in each group is a main step towards the construction of a diagnostic test and developing a therapy. This is just a single example of data source, where number of samples is an order of magnitude smaller than a number of variables, however, there are many similar problems in processing biological data. Such data is difficult to analyse with traditional statistical methods, which work well when a number of attributes describing objects (p) is much smaller than number of objects (n).

The analysis of large p small n data is often performed using a machine learning algorithms, such as neural networks [1], Bayesian networks [2], support vector machines (SVM)[3] or rough sets [4]. They

*Address for correspondence: ICM, University of Warsaw, Pawińskiego 5a, Warsaw, Poland

can be used for building a predictive model in such a case. Often the SVM is the method of choice for biological data. It has been shown that the random forest method performs equally well or better than other methods on a diverse set of problems [5]. Recently, it has been widely used in bioinformatical applications, for example, to predict replication capacity of HIV virus [7], HIV drug resistance [8] and protein interactions [9] as well as in modelling the quantitative structure-activity relationship (QSAR) [10, 11] and mortality process [12]. It has been often used for analysis when both classifier and identification of important variables were the goals of the study, see for example [13, 14]. The relative performance of SVM and RF in various bioinformatical applications is disputed [6, 15], nevertheless the RF method provides two key advantages: it does not need any parameter optimisation to reach its highest performance and it has the built-in statistical estimation of the variable importance. It is especially useful for discovering mechanisms. The ability to assess variable importance is one of the reasons for growing application of RF method in bioinformatics.

Unfortunately, there are several problems affecting importance measures used in RF. The validity of the estimate of the variable importance is based on the assumption that the individual trees grown in the random forest are uncorrelated [5]. It has been shown by simulations [16, 17, 18] and observation of experimental data [8] that this assumption may be false for some data sets. Also, when the number of variables is large, it is difficult to discern truly important variables from these which gain importance due to random correlations in data. To solve this problem we have developed algorithm which compares the apparent importance of the original variables with that of the randomised ones [8]. This algorithm was used to identify mutations which are important for the drug resistance of the HIV virus.

Here we present an improved version of this algorithm. The new version was applied to the synthetic data designed to mimic drug resistance. We have also applied it to a new problem of biological and chemical relevance, namely the identification of significant sequence motifs in aptamers.

The remaining part of the article is organised as follows. We start with a brief description of the Random Forest algorithm in Section 2.1, then Boruta algorithm is described in Section 2.2, followed by an analysis of the synthetic data sets in Section 3 and the biological data sets in Section 4. Finally a discussion and concluding remarks are presented in Section 5.

2. Methods

2.1. Random Forests

Random forest is designed to form an ensemble of weak unbiased classifiers which combine their results during the final classification of each object. Individual classifiers are built as classification trees. Each tree is constructed using different bootstrap sample of the training set. Each bootstrap sample is a result of drawing with replacement the same number of objects as in the original training set. As a result, roughly $\frac{1}{3}$ of objects is not used for building a tree and instead is used for performing an out of bag (OOB) error estimate, and for importance measurement.

At the each step of the tree construction a different subset of attributes is randomly selected. The split is performed using the attribute which leads to the best distribution of data between nodes of the tree. This procedure is performed until the whole tree is built. Constructed tree is used to classify its OOB objects, and the result is used for obtaining the approximations of the classification error and computation of confusion matrices. New objects are classified by all trees in the forest, and the final decision is made by simple voting.

The importance of each variable is estimated in the following way. First the classification of all objects is performed. Each tree contributes its votes only to the classification of objects, which were not used for its construction. The number of votes for a correct class is recorded for each tree. Then the values of given variable are randomly permuted across objects, and the classification is repeated. The number of votes for a correct class is again recorded for each tree. The importance of the variable for the single tree can be then defined as a difference between the number of correct votes cast in original and permuted system divided by number of objects. The importance of the variable is then obtained by averaging importance measures for individual trees. One can also use Z-score, obtained by dividing average value by its standard deviation as an importance measure. The advantage of the latter measure is that it puts more weight in relatively small but stable decrease of classification performace.

2.2. Boruta Algorithm

It has been already mentioned that importance score alone is not sufficient to identify meaningful correlations between variables and the decision attribute. Breiman assumed that, due to low correlations between individual trees, the importance has normal distribution and hence, Z-score can be used to assess importance of the variance. Unfortunately, it has been shown that Breiman assumption is false and therefore one needs some reference which can help to discern the truly important attributes from the non-important ones.

To deal with this problem, we developed an algorithm which provides criteria for selection of important attributes. The algorithm arises from the spirit of random forest – we cope with problems by adding more randomness to the system. The essential idea is very simple: we make a randomised copy of the system, merge the copy with the original and build the classifier for this extended system. To asses importance of the variable in the original system we compare it with that of the randomised variables. Only variables for whose importance is higher than that of the randomised variables are considered important.

The following procedure is applied.

- We build an extended system, where each descriptive variable is replicated. The values of replicated variables are then randomly permuted across objects, consequently all correlations between the replicated variables and the decision attribute are random by design.
- We perform several RF runs, the replicated variables are randomised before each run, and therefore the random part of the system is different for each RF run.
- For each run we compute the importance of all attributes.
- The attribute is deemed important for a single run if its importance is higher than maximal importance of all randomised attributes.
- We perform a statistical test for all attributes. The null hypothesis is that importance of the variable is equal to the maximal importance of the random attributes (MIRA). The test is a two-sided equality test – the hypothesis may be rejected either when importance of the attribute is significantly higher or significantly lower than MIRA. For each attribute we count how many times the importance of the attribute was higher than MIRA (a hit is recorded for the variable). The expected number of hits for N runs is $E(N) = 0.5N$ with standard deviation $S = \sqrt{0.25N}$ (binomial distribution with $p = q = 0.5$). Variable is deemed important (accepted), when the number of hits is

significantly higher than the expected value, and is deemed unimportant (rejected), when the number of hits is significantly lower than the expected value. It is straightforward to compute limits for accepting and rejecting variable for any number of iterations for a desired confidence level.

- Variables which are deemed unimportant are removed from the information system, usually with their randomised mirror pair. In some cases the randomised variables can be kept in the system – it may help in reduction of the number of variables deemed important, without reducing accuracy of RF classifier.
- The procedure is performed for predefined number of iterations, or until all attributes are either rejected or conclusively deemed important, whichever comes first. In the former case, there are attributes left, which are neither approved nor rejected, and are further referred to as undetermined.

The procedure outlined above has been coded as a package in R – an open source statistical environment [19]. It uses the random forest method implemented in R [20]. In practical implementation it is preceded by a three short 'warm-up' phases, with the same general outline but more liberal test for importance. During warm-up phases the reference level is respectively the value of the importance of the fifth, third and second most important random attribute.

3. Synthetic Data Sets

Several variants of the synthetic data were generated. The main goal was to make them resemble real drug-resistance data, like the data analysed in [21]. Two general approaches have been applied, which we called Take I and Take II, each having a few variants. In all cases, a single data set consisting of 1000 objects described by 200 attributes. For each variant 50 experiments were performed, each for a different information system.

3.1. Pseudo-drug Resistance, Take I

The values of the attributes were drawn from the uniform distribution on $[0, 1]$. Objects were divided into two classes using a pseudo-resistance function, which was computed as a sum of 20 randomly chosen attributes (out of 200), each of them was multiplied by a randomly assigned weight. Random weights were assigned to the selected attribute in three alternative ways:

(a) Weights were constant, equal to 1.

(b) Weights were chosen form the uniform distribution on $[0, 1]$.

(c) Weights were chosen form the normal distribution of mean $1/2$ and standard deviation $1/6$.

In the further analysis, for each attribute we considered its importance. We tested two definitions of importance. In the first case, the importance of each attribute was defined as its random weight, scaled such that the maximal attribute importance in 50 simulations was equal to 1. In the second case we adopted the definition of importance proposed by Breiman for the random forest classifier. The importance of *i-th* attribute is defined as the expected value of classification error of generative set of rules, when the information about value of the *i-th* attribute is lost. The value of importance for each attribute is evaluated using the following procedure:

1. Permute the values of the $i-th$ attribute between objects of the information system;
2. Use the generative set of rules to classify the objects;
3. Compute the classification error;
4. Compute the average of the classification error;

This procedure is repeated 50 times and the average importance from 50 runs was used as the estimate of the true attribute's importance. Then the values of importance were scaled such that the maximal attribute importance in 50 simulations was equal to 1.

3.2. Pseudo-drug Resistance, Take II

We have also considered a more complex model the drug resistance, where the decision was based on non-linear relations between variables. In this model we generated rule-based information systems. For each attribute a_i, where $1 \leq i \leq 200$, a weight s_i was drawn from the uniform distribution on $[0, 1]$. For each object in the information system, the value for a attribute a_i was randomly assigned from the binary set $\{0, 1\}$, in such a way that the probability of assigning 1 was equal to s_i.

Afterwards, a set of rules for assigning the decision attribute was generated for each information system. We have used rules having clauses in the form of '$a_i(x) = v_i$', where $1 \leq i \leq 200$ and $v_i \in \{0, 1\}$. The conclusion of the rule was always 'x is in the first decision class'. The objects not matched by any rule were in the second decision class. We analysed variants with rule sets consisting of rules with a constant number of clauses, $n \in \{1, \ldots, 5\}$. as well as the variant where the number of clauses in each new rule was chosen form the uniform distribution on $\{1, \ldots, 5\}$.

Only rule sets, for which the decision classes consisted of $50\% \pm 1\%$ objects were tested.

In all Take II simulations we used Breiman importance. For a rule-based information system it is possible to compute the expected value of the classification error for a given set of rules. For example the theoretical value for set of rules with disjoint domains is given by the following formula:

$$\sum_{rules} 2P_i(1 - P_i)Q_i,$$

where P_i is probability that object satisfies whole rule, Q_i is a probability that object satisfies the rule without condition for the $i - th$ attribute. In the more general case, when each object can satisfy conditions of several rules one can compute the exact theoretical value for each set of rules. Nevertheless it is more practical to estimate the importance experimentally, using the procedure described earlier, and this is the way we used in this study.

4. Biological Data

Aptamers are short RNA or DNA chains that bind strongly and specifically to various micro- and macromolecular targets. [22] Sequences of aptamers binding to a particular targets can be acquired from public databases [23]. Boruta has been used to identify important motifs in aptamer sequences. We selected 23 data sets, consisting of at least 30 different aptamer sequences each. Each aptamer sequence was represented by its n-gram composition. We used 3-, 4-, and 5-gram spectra. Presence of the n-gram in

the sequence was denoted by 1, absence by 0. For each aptamer set we constructed a set of negative examples. This set consisted of equal number of sequences constructed by random shuffling of the original ones. Finally a decision attribute was assigned to each object – 1s for the aptamers and 0s for the negative examples. In total the used data set contained 1246 original aptamer sequences and the equal number of their randomised counterparts. Then the RF algorithm was used to build a classification for each separate aptamer set. In 20 cases the random forest classifier obtained the OOB error lower than 30%.

The experience with the simulated information systems shows, that this is a reasonably safe cut-off value separating classifiers built on random data from those built on data containing information. Finally Boruta algorithm was used to select important variables, and RF classifiers were constructed using reduced representation. An alternative procedure for identification of the important motifs was introduced to check the relevance of the motifs found by Boruta.

The results obtained with the Boruta algorithm were compared with a statistical model for the frequency of occurrence of n-grams in the aptameric sequences. In this model the frequency of occurrence for $(n+1)$-grams is obtained from the frequency of n-grams in an iterative fashion.

The expected frequencies of occurrence for $(n+1)$-grams were built using expected frequencies of n-grams and observed frequencies of nucleotides as

$f_{exp}(a_{i_1} \dots a_{i_n} \cdot a_i) = f(a_{i_1} \dots a_{i_n}) \cdot f(a_i)$.

In a random reference model this reduces to

$f_{exp}(a_{i_1} \dots a_{i_n} \cdot a_i) = f(a_{i_1}) \dots f(a_{i_n}) \cdot f(a_i)$.

If a significant over-representation was found for a particular n-gram then the observed frequency of that n-gram was used for generation of (n+1)-grams using the formula

$f_{exp}(a_{i_1} \dots a_{i_n} \cdot a_i) = f_{obs}(a_{i_1} \dots a_{i_n}) \cdot f(a_i)$.

All expected frequencies of other n-grams were scaled, to account for elevated expected frequency of n-gram $a_{i_1} \dots a_{i_k} \cdot a_i$. Then we compared:

- Two sets of significant n-grams.
- The coverage of aptamer sequences by significant n-grams obtained with two methods.
- The classification error of random forest classifiers, which were built using the significant n-grams obtained with both methods.

5. Results and Discussion

5.1. Synthetic Data

The measure of the algorithm's performance is its ability to discern between important and non-important attributes. The perfect algorithm should be able to select all the important attributes and reject non-important ones. We measured the number of

- *false positives* – attributes which were identified as important, while they were not used in generation of the decision attribute,
- *false negatives* – attributes which were not identified as important, while they were used in generation of the decision attribute.

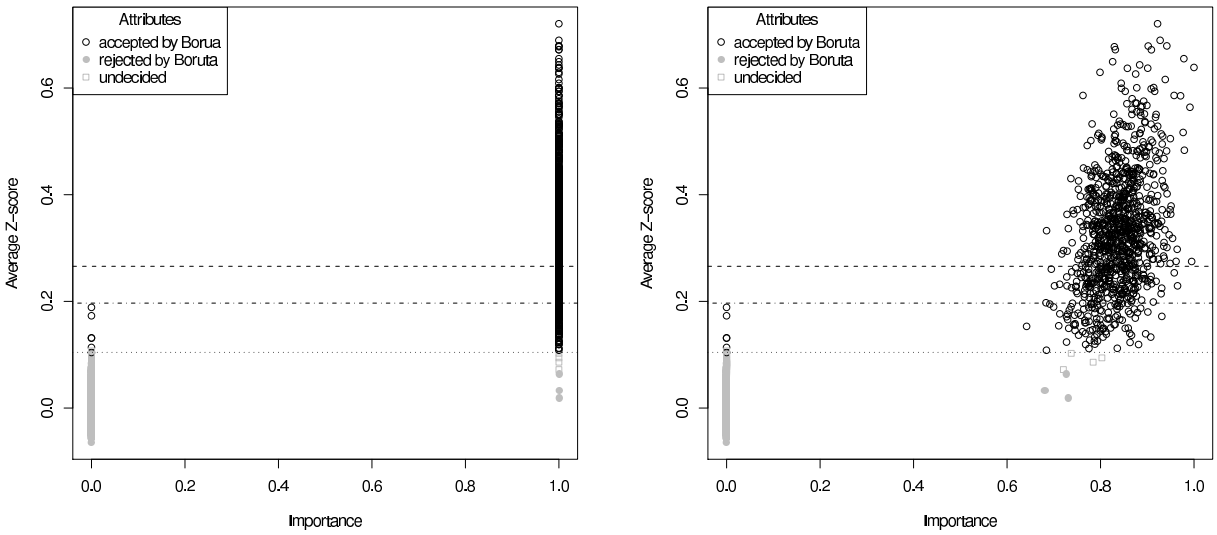

Figure 1. Pseudo-drug resistance, take I – binary attribute weights for important attributes. *Left:* weights based importance. *Right:* Breiman importance. For reference we show highest Z-score attained by the random attribute in all 50 data sets (top horizontal line), average of 50 highest Z-scores obtained in 50 runs of the Boruta algorithm (middle horizontal line) and average of the highest Z-scores obtained in each random forest run (bottom horizontal line).

5.1.1. Pseudo-drug Resistance, Take I

Boruta performs best for the binary system, with attribute weights 1 or 0, see Fig. 1. In this case almost all the important attributes were found, namely 993 out of 1000 with a small number of false positive cases (6 in 50 independent trials). One may notice that while all weights of the important attributes are equal by design (left panel), the Breiman importance of the attributes varies between 0.6 and 1.0. The values of importance are clearly separated along X axis (the a priori importance), but they are not well separated along Y axis (average Z-score of the random forest). In many cases, the importance of the non-important attributes was higher than that of the important ones. Nevertheless, the false positive identification of the non-important attributes happened only in 6 out of 50 tests.

It is interesting to compare these results with attempts performed with nearly identical procedure, differing only in number of variables used for generation of the decision. We used two sets, one with the decision generated as a sum of values of 10 attributes and another with decision generated as a sum of 5 attributes. The results for these sets are shown in Fig. 2. One can see, that now a clear separation along Y axis (average Z-score) appears between attributes which are important by design and those which are not important by design. In effect all attributes which were used for generation of the decision class were identified as important by the algorithm. On the other hand several attributes, which are not important by design attain the Z-score high enough to be classified as important. There are 25 false positives (compared to 500 true positives) when the number of attributes used for generation of the decision is 10, and 57 false positives (compared to 250 true positives) when a number of attributes used for generation of the decision is 5.

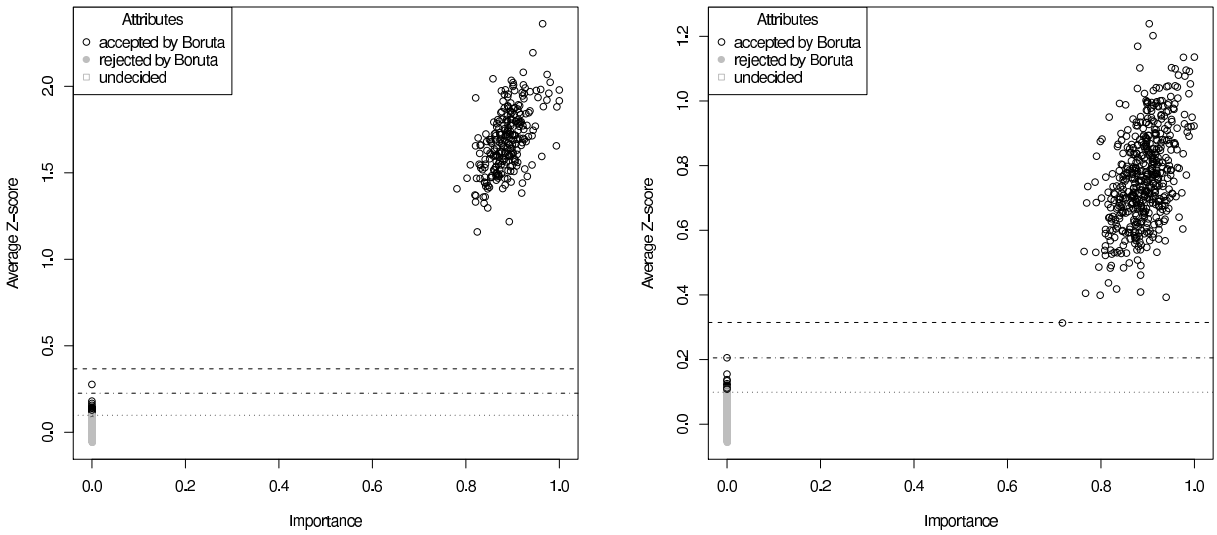

Figure 2. Pseudo-drug resistance, take I – binary attribute weights for important attributes. *Left:* Decision generated with 5 attributes. *Right:* Decision generated with 10 attributes. The reference values are displayed as in the previous figure.

This observation can be explained in the following way: by increasing the number of attributes in the generative formula one decreases significance of any single variable. Apparently, this is also true for the importance of the random attributes. Nevertheless, it seems that the value of MIRA is decreased less than both average importance of the truly important attributes and average importance of the randomly correlated attributes. In effect, the sensitivity of the algorithm is decreasing and specificity is increasing with the increasing number of truly important attributes.

This misclassification happens as a result of random correlations between attributes. These correlations are inevitable for small samples. One may notice that Z-scores of attributes selected as important are higher than average of the maximal Z-score of the random attribute computed for all random forest runs.

Performance of the algorithm is lower for systems with weights assigned randomly. Boruta finds 681 out of 1000 important attributes for systems with normally distributed weights and 586 out of 1000 important attributes for systems with uniformly distributed weights . The number of false positives is 15 and 14 in the case of normal distribution of weights and uniform distribution of weights, respectively.

The visual analysis of Fig. 3 and Fig. 4, shows that the Z-score computed by random forest is highly correlated with the *a priori* importance. The correlation coefficient varies between 0.84 for normally distributed weights and weight based importance and 0.94 for uniformly distributed weights and Breiman importance. As should be expected, the correlation coefficient between Z-score and Breiman importance is higher than that between Z-score and weight-based importance for both the uniform and the normal distribution. In both cases the difference is about 0.05.

Regardless of the attribute weights distribution, the attributes, which were important by design and were not found by our algorithm had relatively low importance (less than 0.4). This can be best seen on Fig. 5. A closer look at Fig. 5 shows that Boruta misses attributes with lower importance.

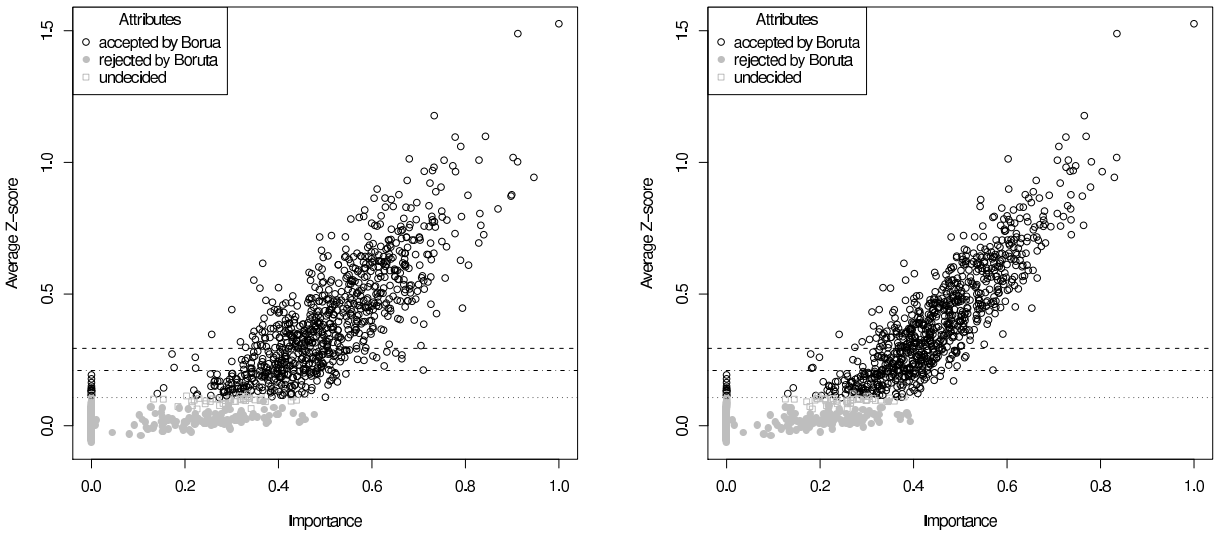

Figure 3. Pseudo-drug resistance, take I – normally distributed attribute weights. *Left:* weight-based importance. *Right:* Breiman importance. The reference values are displayed as in the previous figures.

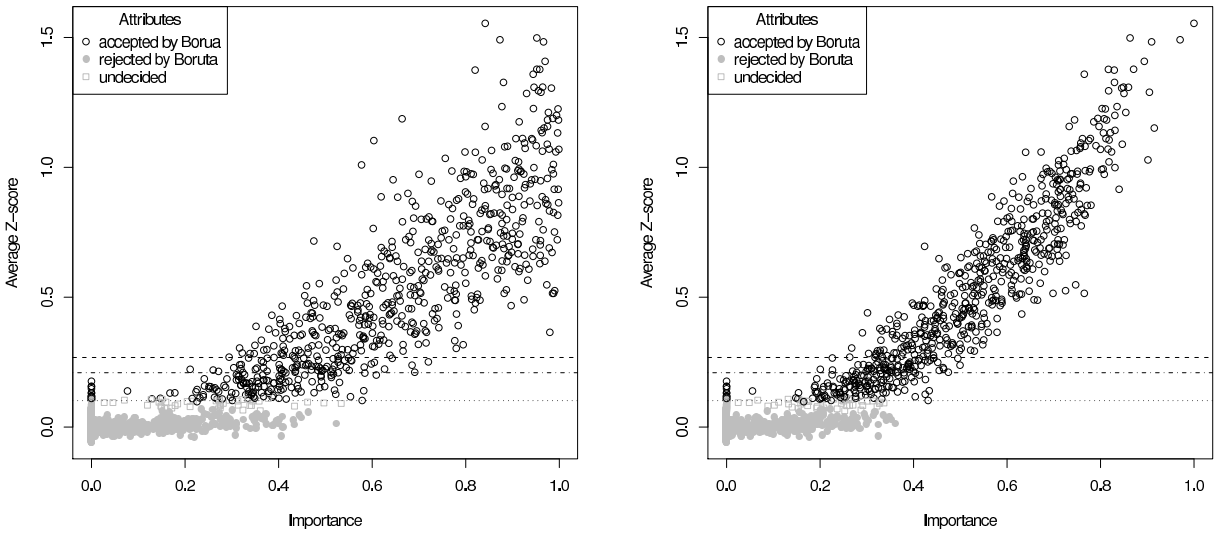

Figure 4. Pseudo-drug resistance, take I – uniformly distributed attribute weights. *Left:* weight-based importance. *Right:* Breiman importance. The horizontal lines show the highest Z-score attained by the random attribute in all 50 data sets (top), average of 50 highest Z-scores obtained in 50 runs of the Boruta algorithm (middle) and average of the highest Z-scores obtained in each random forest run (bottom).

It is also important to point out that the importance scores for attributes form continuous cloud and our algorithm is used for finding the practical cut-off value between important and unimportant attributes. It finds all attributes with Z-score higher than that of the highest scoring random attribute, as well as several important attributes with lower Z-scores.

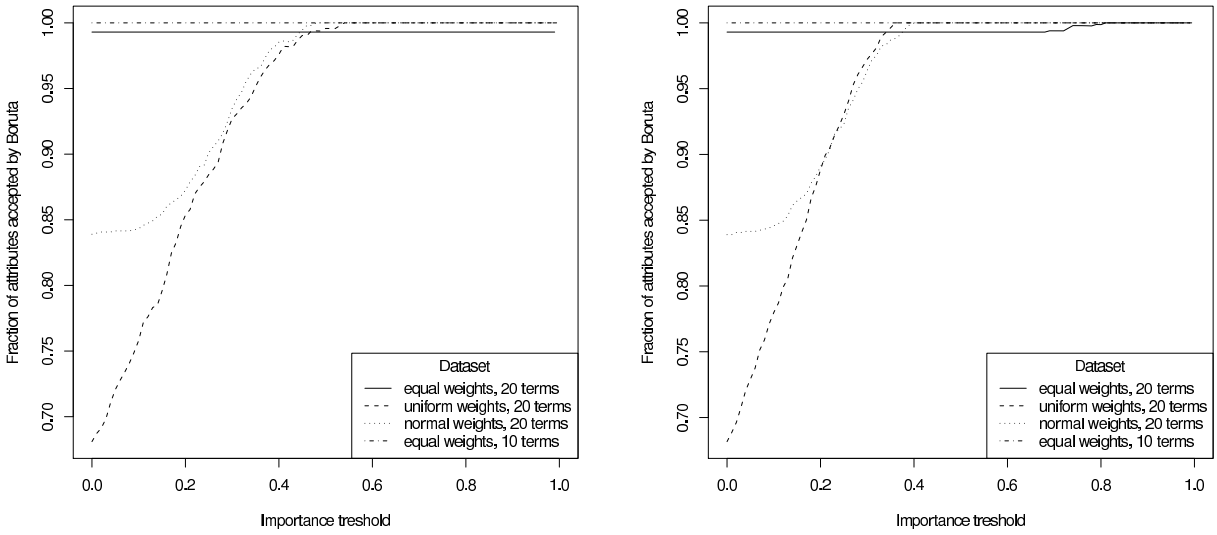

Figure 5. Pseudo-drug resistance, take I – fraction of attributes accepted by Boruta vs. importance threshold. *Left:* weight-based importance. *Right:* Breiman importance.

5.1.2. Pseudo-drug Resistance, Take II

In the take two the decision class is assigned by generative rules. The number of important attributes is growing proportionally to the number of clauses in the rules used for generating decision systems. There are 351, 558, 705 and 802 (out of 10000) important attributes in systems generated with 2-, 3-, 4- and 5-clause rules, respectively. For mixed systems with 1-5-clause rules, the number of important attributes is 471. Increase of the number of important attributes results in a decrease of the importance of a single attribute, and hence makes it more difficult to recognise. It is clearly seen in the top-left panel of Fig. 6 where the fraction of important attributes discovered by the algorithm with higher importance than the given cut-off is displayed.

For cut-off zero (all important attributes) algorithm finds over 95% of important attributes in the case of 2-clause rules, compared to 86% in the case of 5-clause rules. With increasing importance cut-off the fraction of important attributes found by algorithm is rapidly increasing (top left panel of Fig. 6. For example, the algorithm finds all important attributes for 2-, 3-, and 1-5- clause rules at the importance cut-off equal to 0.05. Algorithm finds all important attributes for all rules at the importance cut-off 0.16. In all cases algorithm is able to find many important attributes with Z-scores lower than the highest Z-score of the random attribute, see Fig. 6.

In all cases algorithm finds several false positives, that is attributes, which are not important by design, but nevertheless are selected as important. There are 17, 52, 104, 116, 45 false positives for rules with 2-, 3-, 4-, 5-, and 1-5- clauses, respectively. The false importance arises due to insufficient number of samples and random correlations between variables. When the number of variables is similar to the number of objects (200 and 1000, in our case, respectively) the random correlations between variables can be used by classifier to build a model for data. Unfortunately such situations are unavoidable in real life problems, where the number of objects is small and number of variables can be large, in many cases

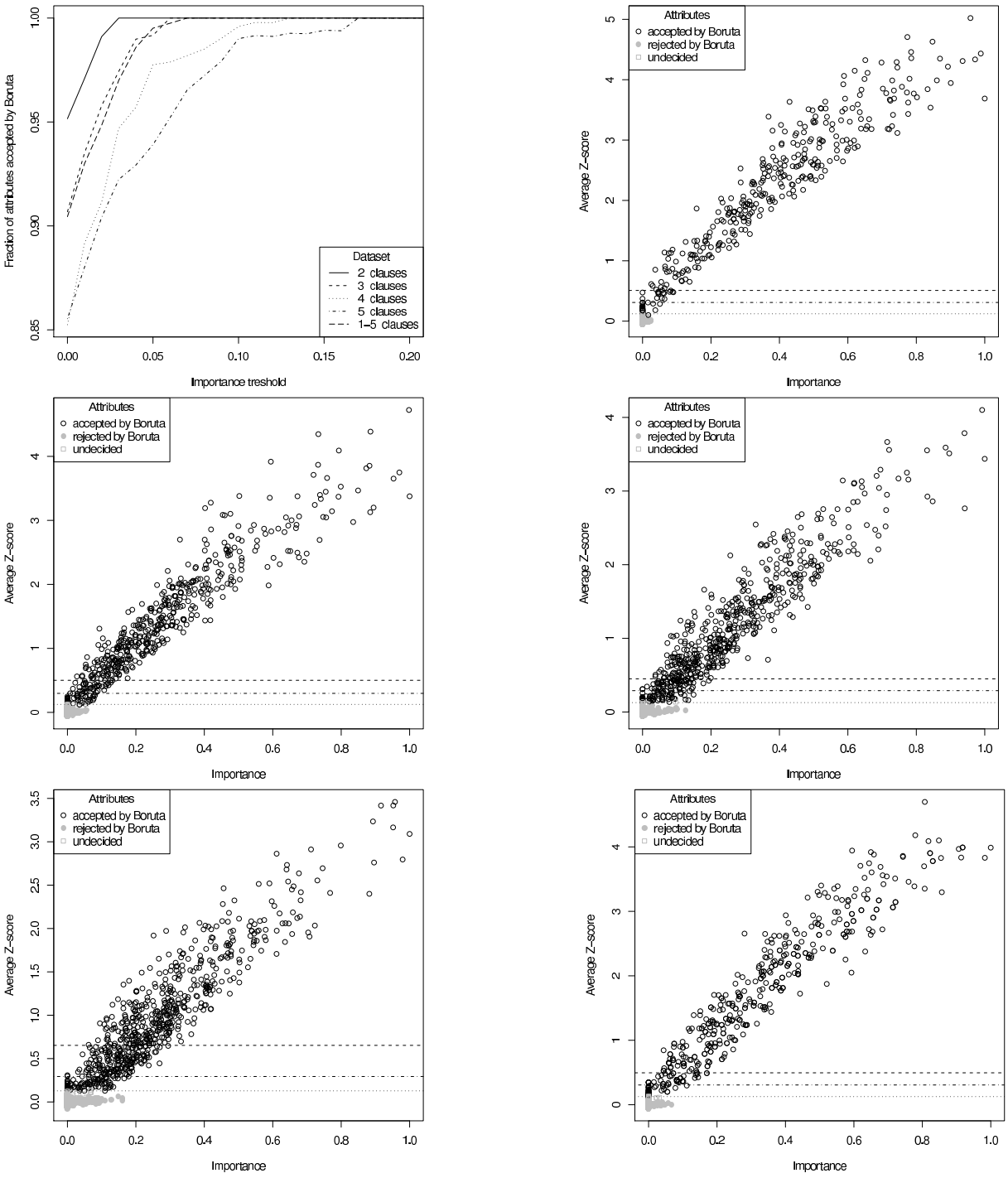

Figure 6. Pseudo-drug resistance, take II.

Left top: Fraction of attributes accepted by Boruta vs. importance threshold. *Right top:* Rules with 2 clauses. *Left middle:* Rules with 3 clauses. *Right middle:* Rules with 4 clauses. *Left bottom:* Rules with 5 clauses. *Right bottom:* Rules with mixed (1-5) number of clauses. The reference values are displayed as in the previous figures.

it can be even orders of magnitude higher than the number of objects. Fortunately, the closer inspection of the results shows, that the apparent importance of the false positive cases is rather low, much lower than that of the most important attributes.

In all cases the correlation between *a priori* importance and Z-score is remarkably high, varying between 0.97 for rules with 5 clauses and 0.98 for rules with 2 clauses. This result shows, that average of the Z-score computed by the random forest is very reliable measure of the true Breiman importance of the attributes.

5.2. Biological Data

The summary of results for the biological data is shown in Table 1. In 2 cases out of 23 the initial random forest classifier had OOB error higher than 30%, which is assumed to be a borderline between reliable and unreliable classifier. In both cases, Boruta algorithm could not find any important attributes describing the system. In one case application of the algorithm resulted in significant increase of the OOB error in the final classifier. Error increased from 14% to 38%, turned reliable classifier into a non-reliable one.

In remaining 20 cases the algorithm worked well. Average OOB error for this set was 11%, both in the original data set and in the data set after attribute selection procedure with Boruta. The average number of attributes used to describe the information system was reduced 65 times (from 1170 to 18). In 9 cases the OOB error has decreased after attribute selection, in 8 cases it has increased and in 3 cases it remained constant.

The average coverage of the aptamer sequences by important motifs is 60%. The alternative procedure for motif finding, which was based on statistical model of n-gram usually return less important n-grams, which cover only 40% of the sequence length. It is interesting to note, that important sequence motifs found with the help of two n-gram sets are very similar, nevertheless, they differ in a non-trivial way. We used the following asymmetric sequence similarity measure:

$$similarity = \sum_a \sum_i \frac{N^{a,i}_{common}}{N^{a,i}_{freq}}$$

where $N^{a,i}_{common}$ ($N^{a,i}_{freq}$) is a number of residues shared by motifs found by two methods (found by the alternative method) in $i-th$ sequence belonging to the $a-th$ aptamer set. The average similarity for our data sets is slightly less than 90%. More than 10% of total lengths of important motifs found with the alternative n-gram set has not been deemed important by Boruta. The comparison of the n-gram sets selected by two methods is interesting. While the similarity of the important sequence motifs is almost 90%, there is only a very small subset of n-grams which were found important by both methods. On the other hand, most n-grams in both sets can be constructed using n-grams from alternative set. This result is consistent with the biological interpretation of the data - in fact sequence motifs are relevant for aptameric properties and identical sequence motif can be constructed using different n-grams.

The n-gram based motifs found with Boruta algorithm can be used in the analysis of the aptamer sequences to reveal the regions which are important for their activity and structure. Boruta findings are consistent with the available experimental data. For instance, MSA analysis and NMR data suggest that `GAAGA` motif is responsible for binding adenine in adenine related aptamers. Boruta has found this motif as important in all classes of aptamers binding to three adenine-containing targets that were

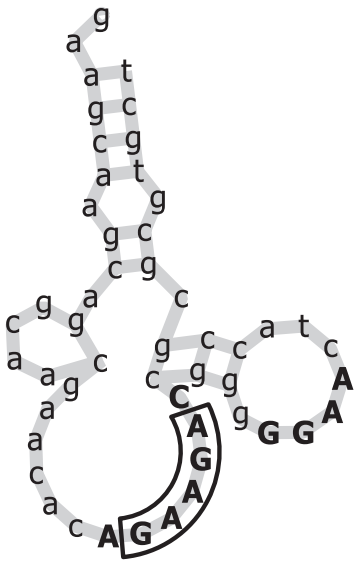

Figure 7. RNAfold prediction of one of ATP aptamers. Important fragments, found using Boruta and `GAAGA` motif, have been highlighted.

analysed (adenosine triphosphate, flavin adenine dinucleotide and S-adenosyl methionine). This motif is highlighted on the secondary structure of the ATP-binding aptamer, obtained with the RNAFold program [24] in Fig. 7. It is localised on the loop, and this is consistent also with the structural data [25].

5.3. Discussion

The biologically relevant results of the current study give another example showing that the selection of the important attributes can reveal important information. It opens, for example, a possibility of application of the random forest classifier as a filter for finding aptameric sequences in genes.

On the other hand, the analysis of the synthetic data shows that the results of the importance analysis could be possibly misleading. One should note, that in a system with small sample sizes and large number of attributes correlations between decision attribute and random combinations of attributes may be present. We have shown, that random correlations between attributes can lead to creation of false dependencies between attributes and decision, which are strong enough, to pass the statistical test of validity. In particular, correlations of the non important attributes with important attributes for small subsets of data are possible. The machine learning classifier might not be able to discern such correlations from genuine correlations with decision attribute. It means that for a given data set the attributes which are non informative by design, might still be informative by chance. Therefore the importance of the attribute in the machine learning classifier may be used as a hint for existence of a relationship between variables of the information system and decision attribute, not as a decisive proof.

Nevertheless, Boruta algorithm has clearly shown its usability. The absolute value of Z-score is not very informative of the attribute importance. We have shown previously [8] that Z-scores of random attributes can be as high as 8. On the other hand the Z-score of attributes in the synthetic systems studied here has never been higher than 2. For all but the simplest case, Boruta allowed much more

efficient selection of the important attributes than would be possible otherwise. It provides a criterion for a variable selection which is based on simple statistical test.

Acknowledgements
Computations were performed at ICM, University of Warsaw, grant G34-5.

References

[1] Bishop, C.M. (1996) *Neural Networks for Pattern Recognition*. Clarendon Press, Oxford

[2] Pearl, J. (1988) *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann Publishers, San Francisco.

[3] Vapnik, V. N. (1998) *Statistical learning theory*. New York, Wiley;

[4] Pawlak, Z. (1981) Information systems theoretical foundations, *Inf. Syst.* **6**, 205–218.

[5] Breiman, L. Random Forests, *Machine Learning* **45** (2001), 5–32. Also see the bibliography at: http://www.stat.berkeley.edu/˜breiman/RandomForests/cc_papers.htm

[6] Diaz-Uriarte, R., Alvarez de Andres, S (2006) Gene selection and classification of microarray data using random forest. *BMC Bioinformatics* **7**:3.

[7] Segal, M. R., Barbour, J. D., Grant, R. M. (2004) Relating HIV-1 Sequence Variation to Replication Capacity via Trees and Forests. *Stat. Appl. Gen. Mol. Biol.*, **3**:2.

[8] Rudnicki, W. R., Kierczak, M., Koronacki, J., Komorowski, H. J. (2006) A Statistical Method for Determining Importance of Variables in an Information System. In Greco, S., Hata, Y., Hirano, S., Inuiguchi, M., Miyamoto, S., Nguyen, H. S., Slowinski, R. (Eds.): *Lecture Notes in Computer Science ? 4259/2006 5th International Conference, RSCTC 2006, Kobe, Japan, November 6-8, 2006, Proceedings*, 557–566.

[9] Qi, Y., Bar-Joseph, Z., Klein-Seetharaman, J. (2006) Evaluation of Different Biological Data and Computational Classification Methods for Use in Protein Interaction Prediction. *Proteins*, **63**, 490–500.

[10] Guha, R., Jurs, P. C. (2003). Development of Linear, Ensemble, and Nonlinear Models for the Prediction and Interpretation of the Biological Activity of a Set of PDGFR Inhibitors. *J. Chem. Inf. Comp. Sci.*, **44**, 2179–2189.

[11] Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., Feuston, B. P. (2003). Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *J. Chem. Inf. Comp. Sci.*, **43**, 1947–1958.

[12] Ward, M. M., Pajevic, S., Dreyfuss, J., Malley, J. D. (2006). Short-Term Prediction of Mortality in Patients with Systemic Lupus Erythematosus: Classification of Outcomes Using Random Forests. *Arthritis and Rheumatism*, **55**, 74–80.

[13] Lunetta, K. L., Hayward, L. B., Segal, J., Eerdewegh, P. V, (2004). Screening Large-Scale Association Study Data: Exploiting Interactions Using Random Forests. *BMC Genetics*, **5**:32.

[14] Bureau, A., Dupuis, J., Falls, K., Lunetta, K. L., Hayward, B., Keith, T. P., Eerdewegh, P. V. (2005) Identifying SNPs Predictive of Phenotype Using Random Forests. *Gen. Epidem.*, **28**:171–182.

[15] Statnikov, A., Wang, L., Aliferis, C. F. (2008) A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. *BMC Bioinformatics* **9**:319

[16] Strobl, C., Boulesteix, A.-L., Zeileis, A., Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution *BMC Bioinformatics*, **8**:25

[17] Strobl, C., Zeileis, A., (2008). *Danger: High Power! ? Exploring the Statistical Properties of a Test for Random Forest Variable Importance. Technical Report Number 017.* Department of Statistics, University of Munich

[18] Strobl, C., Boulesteix, A.-L., Kneib, T., Augistin, T., Zeileis, A., (2008). *Conditional Variable Importance for Random Forests. Technical Report Number 23*. Department of Statistics, University of Munich

[19] R Development Core Team (2008). *R: A language and environment for statistical computing.* R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org

[20] Liaw, A., Wiener, M., (2002). Classification and Regression by randomForest. *R News* **2(3)**, 18–22.

[21] Kierczak, M., Rudnicki, W. R., Koronacki, J., Komorowski, H. J. Kierczak M, Rudnicki WR, and Komorowski J. Construction of rough sets-based classifiers

[22] Ellington, A. D., Szostak, J. W. (1990) In vitro selection of RNA molecules that bind specific ligands. *Nature*, **346**, 818–822.

[23] Lee, J. F., Hesselberth, J. R, Meyers, L. A, Ellington, A. D., (2004) Aptamer database. *Nucleic Acids Res.*, **32**, D95–D100.

[24] Hofacker, I. L., Fontana, W., Stadler, P. F., L. Sebastian Bonhoeffer, L. S., Tacker, M., Schuster, P. (1994) Fast Folding and Comparison of RNA Secondary Structures (The Vienna RNA Package) *Monatsh. Chem.* **125**, 167–188.

[25] Jiang, F., Kumar, R. A., Jones, R. A., Patel, D. J. (1996) Structural basis of RNA folding and recognition in an AMP-RNA aptamer complex. *Nature*, **382**, 183–186.

Table 1. Results of the aptamer sets analysis

			Error OOB		# Attributes	
#	Target	# Seq.	RF only	RF + Boruta	RF only	RF + Boruta
1	Isoleucine	139	0.02	0.02	1308	28
2	FAD	46	0.05	0.02	1181	11
3	REL-GATA	125	0.02	0.03	934	10
4	VEGF165	44	0.05	0.03	1342	4
5	Vasopressin	43	0.03	0.03	1284	6
6	SAM	34	0.06	0.04	1335	8
7	MBGT	32	0.09	0.08	1103	3
8	ATP	66	0.07	0.09	1178	2
9	Arginine	41	0.17	0.10	1062	16
10	WT-suppressor	30	0.08	0.10	1296	60
11	IDI4	49	0.07	0.10	1302	45
12	Codeine	47	0.17	0.11	1124	23
13	NTS-1	30	0.13	0.12	1287	11
14	hnRNP	108	0.13	0.13	1262	6
15	Pefloxacin	42	0.23	0.18	1097	9
16	MS2-cp	82	0.15	0.19	1313	45
17	g5p	32	0.11	0.20	1099	23
18	Hematoporphyrin	34	0.26	0.22	1165	31
19	Dopamine	54	0.20	0.22	821	16
20	Elastase	52	0.19	0.24	866	7
21	HIV1-Tar	52	0.14	0.38	1073	22
22	KH domain	33	0.36	1.00	1182	0
23	NPY	31	0.60	1.00	978	0