Identification of Statistically Significant Features from Random Forests

Jérôme Paul, Michel Verleysen, and Pierre Dupont

Université catholique de Louvain – ICTEAM/Machine Learning Group
Place Sainte Barbe 2, 1348 Louvain-la-Neuve – Belgium
{jerome.paul,michel.verleysen,pierre.dupont}@uclouvain.be
http://www.ucl.ac.be/mlg/

Abstract. Embedded feature selection can be performed by analyzing the variables used in a Random Forest. Such a multivariate selection takes into account the interactions between variables but is not easy to interpret in a statistical sense. We propose a statistical procedure to measure variable importance that tests if variables are significantly useful in combination with others in a forest. We show experimentally that this new importance index correctly identifies relevant variables. The top of the variable ranking is, as expected, largely correlated with Breiman's importance index based on a permutation test. Our measure has the additional benefit to produce p-values from the forest voting process. Such p-values offer a very natural way to decide which features are significantly relevant while controlling the false discovery rate.

1 Introduction

Feature selection aims at finding a subset of most relevant variables for a prediction task. To this end, univariate filters, such as a t-test, are commonly used because they are fast to compute and their associated p-values are easy to interpret. However such a univariate feature ranking does not take into account the possible interactions between variables. In contrast, a feature selection procedure embedded into the estimation of a multivariate predictive model typically captures those interactions.

A representative example of such an embedded variable importance measure has been proposed by Breiman with its Random Forest algorithm (RF) [1]. While this importance index is effective to rank variables it is difficult to decide how many such variables should eventually be kept. This question could be addressed through an additional validation protocol at the expense of an increased computational cost. In this work, we propose an alternative that avoids such additional cost and offers a statistical interpretation of the selected variables.

The proposed multivariate RF feature importance index uses out-of-bag (OOB) samples to measure changes in the distribution of class votes when permuting a particular variable. It results in p-values, corrected for multiple testing, measuring how variables are useful in combination with other variables of the

model. Such p-values offer a very natural threshold for deciding which variables are statistically relevant.

The remainder of this document is organised as follows. Section 2 presents the notations and reminds Breiman's RF feature importance measure. Section 3 introduces the new feature importance index. Experiments are discussed in Section 4. Finally, Section 5 concludes this document and proposes hints for possible future work.

2 Context and Notations

Let $X^{n\times p}$ be the data matrix consisting of n data in a p-dimensional space and y a vector of size n containing the corresponding class labels. A RF model [1] is made of an ensemble of trees, each of which is grown from a bootstrap sample of the n data points. For each tree, the selected samples form the bag (B), the remaining ones constitute the OOB ($\overline{B}$). Let $\mathcal{B}$ stand for the set of bags over the ensemble and $\overline{\mathcal{B}}$ be the set of corresponding OOBs. We have $|\mathcal{B}| = |\overline{\mathcal{B}}| = T$, the number of trees in the forest.

In order to compute feature importances, Breiman[1] proposes a permutation test procedure based on accuracy. For each variable x_j, there is one permutation test per tree in the forest. For an OOB sample $\overline{B}_k$ corresponding to the k-th tree of the ensemble, one considers the original values of the variable x_j and a random permutation $\tilde{x}_j$ of its values on $\overline{B}_k$. The difference in prediction error using the permuted and original variable is recorded and averaged over all the OOBs in the forest. The higher this index, the more important the variable is because it corresponds to a stronger increase of the classification error when permuting it. The importance measure J_a of the variable x_j is then defined as:

$$J_a(x_j) = \frac{1}{T}\sum_{\overline{B}_k\in\overline{\mathcal{B}}}\frac{1}{|\overline{B}_k|}\left(\sum_{i\in\overline{B}_k} I(h_k^{\tilde{x}_j}(i)\neq y_i) - I(h_k(i)\neq y_i)\right) \tag{1}$$

where y_i is the true class label of the OOB example i, I is an indicator function, $h_k(i)$ is the class label of the example i as predicted by the tree estimated on the bag B_k, $h_k^{\tilde{x}_j}(i)$ is the predicted class label from the same tree while the values of the variable x_j have been permuted on $\overline{B}_k$. Such a permutation does not change the tree but potentially changes the prediction on the out-of-bag example since its j-th dimension is modified after the permutation. Since the predictors with the original variable h_k and the permuted variable $h_k^{\tilde{x}_j}$ are individual decision trees, the sum over the various trees where this variable is present represents the ensemble behaviour, respectively from the original variable values and its various permutations.

3 A Statistical Feature Importance Index from RF

While J_a is able to capture individual variable importances conditioned to the other variables used in the forest, it is not easily interpretable. In particular,

it does not define a clear threshold to highlight statistically relevant variables. In the following sections, we propose a statistical feature importance measure closely related to J_a, and then compare it with an existing approach that aims at providing statistical interpretation to feature importance scores.

3.1 Definition

In [2], the authors analyse the convergence properties of ensembles of predictors. Their statistical analysis allows us to determine the number of classifiers needed in an ensemble in order to make the same predictions as an ensemble of infinite size. To do so, they analyse the voting process and have a close look to the class vote distribution of such ensembles.

In the present work, we combine the idea of Breiman's J_a to use a permutation test with the analysis of the tree class vote distribution of the forest. We propose to perform a statistical test that assesses whether permuting a variable significantly influences that distribution. The hypothesis is that removing an important variable signal by permuting it should change individual tree predictions, hence the class vote distribution.

One can estimate this distribution using the OOB data. In a binary classification setting, for each data point in an OOB, the prediction of the corresponding tree can fall into one of the four following cases : correct prediction of class 1 (TP), correct prediction of class 0 (TN), incorrect prediction of class 1 (FP) and incorrect prediction of class 0 (FN). Summing the occurrences of those cases over all the OOBs gives an estimation of the class vote distribution of the whole forest. The same can be performed when permuting a particular feature x_j. This gives an estimation of the class vote distribution of the forest after perturbing this variable. The various counts obtained can be arranged into a 4×2 contingency table. The first variable that can take four different values is the class vote. The second one is an indicator variable to represent whether x_j has been permuted or not. Formally a contingency table is defined as follows for each variable x_j:

$$\begin{array}{c|cc} & x_j & \tilde{x}_j \\ \hline \text{TN} & s(0,0) & s^{\tilde{x}_j}(0,0) \\ \text{FP} & s(0,1) & s^{\tilde{x}_j}(0,1) \\ \text{FN} & s(1,0) & s^{\tilde{x}_j}(1,0) \\ \text{TP} & s(1,1) & s^{\tilde{x}_j}(1,1) \end{array} \tag{2}$$

where

$$s(l_1, l_2) = \sum_{\overline{B}_k \in \overline{\mathcal{B}}} \sum_{i \in \overline{B}_k} I(y_i = l_1 \text{ and } h_k(i) = l_2) \tag{3}$$

and $s^{\tilde{x}_j}(l_1, l_2)$ is defined the same way with $h_k^{\tilde{x}_j}(i)$ instead of $h_k(i)$.

In order to quantify whether the class vote distribution changes when permuting x_j, one can use Pearson's χ^2 test of independence on the contingency table defined above. This test allows to measure if joined occurrences of two variables are independent of each other. Rejecting the null hypothesis that they

are independent with a low p-value $p_{\chi^2}(x_j)$ would mean that x_j influences the distribution and is therefore important. We note that, even on small datasets, there is no need to consider a Fisher's exact test instead of Pearson's χ^2 since cell counts are generally sufficiently large: the sum of all counts is twice the sum of all OOB sizes, which is influenced by the number of trees T.

If the importance of several variables has to be assessed *e.g.* to find out which features are important, one should be careful and correct the obtained p-values for multiple testing. Indeed, if 1000 dimensions are evaluated using the commonly accepted 0.05 significance threshold, 50 variables are expected to be falsely deemed important. To control that false discovery rate (FDR), p-values can be rescaled *e.g.* using the Benjamini-Hochberg correction [3].

Let $p^{fdr}_{\chi^2}(x_j)$ be the value of $p_{\chi^2}(x_j)$ after FDR correction, the new importance measure is defined as

$$J_{\chi^2}(x_j) = p^{fdr}_{\chi^2}(x_j) \tag{4}$$

This statistical importance index is closely related to Breiman's J_a. The two terms inside the innermost sum of Equation (1) correspond to counts of FP et FN for permuted and non permuted variable x_j. This is encoded by the second and third lines of contingency table in Equation (2). However, there are some differences between the two approaches. First, the central term of J_a (eq. (1)) is normalized by each OOB size while the contingency table of J_{χ^2} (eq. (2)) considers global counts. This follows from the fact that J_a estimates an average decrease in accuracy on the OOB samples while J_{χ^2} estimates a distribution on those samples. More importantly, the very nature of those importance indices differ. J_a is an aggregate measure of prediction performances whereas J_{χ^2} (eq. (4)) is a p-value from a statistical test. The interpretation of this new index is therefore much more easy from a statistical significance viewpoint. In particular, it allows one to decide if a variable is significantly important in the voting process of a RF. As a consequence, the lower J_{χ^2} the more important the corresponding feature, while it is the opposite for J_a.

3.2 Additional Related Work

In [4], the authors compare several ways to obtain a statistically interpretable index from a feature relevance score. Their goal is to convert feature rankings to statistical measures such as the false discovery rate, the family wise error rate or p-values. To do so, most of their proposed methods make use of an external permutation procedure to compute some null distribution from which those metrics are estimated. The external permutation tests repeatedly compute feature rankings on dataset variants where some features are randomly permuted.

A few differences with our proposed index can be highlighted. First, even if it can be applied to convert Breiman's J_a to a statistically interpretable measure, the approach in [4] is conceptually more complex than ours: there is an additional resampling layer on top of the RF algorithm. This *external* resampling encompasses the growing of many forests and should not be confused with the

internal bootstrap mechanism at the tree level, inside the forest. This external resampling can introduce some meta-parameters such as the number of external resamplings and the number of instances to be sampled. On the other side, our approach runs on a single RF. There is no need for additional meta-parameters but it is less general: it is restricted to algorithms based on classifier ensembles. The external resampling procedures in [4] implies that those methods are also computationally more complex than J_{χ^2}. Indeed, they would multiply the cost of computing a ranking with J_a by the number of external resamplings whereas the time complexity of computing J_{χ^2} for p variables is exactly the same as with Breiman's J_a. If we assume that each tree node splits its instances into two sets of equal sizes until having one point per leaf, then the depth of a tree is $\log n$ and the time complexity of classifying an instance with one tree is $O(\log n)$. Hence, the global time complexity of computing a ranking of p variables is $O(T \cdot p \cdot n \cdot \log n)$. Algorithm 1 details the time complexity analysis.

$res \leftarrow initRes()$ // $\Theta(p)$
for $x_j \in Variables$ **do** // $\Theta(p)$
  $contTable \leftarrow init()$ // $\Theta(1)$
  for $\overline{B}_k \in \overline{\mathcal{B}}$ **do** // $\Theta(T)$
    $\tilde{x}_j \leftarrow perm(x_j, \overline{B}_k)$ // $\Theta(n)$
    for $i \in \overline{B}_k$ **do** // $O(n)$
      $a \leftarrow h_k(i)$ // $\Theta(depth)$
      $b \leftarrow h_k^{\tilde{x}_j}(i)$ // $\Theta(depth)$
      $contTable \leftarrow update(contTable, a, b, y_i)$ // $\Theta(1)$
    end
  end
  $res[x_j] \leftarrow \chi^2(contTable)$ // $\Theta(1)$
end
return res

Algorithm 1: Pseudo-code for computing the importance of all variables with a forest of $T = |\overline{\mathcal{B}}|$ trees

4 Experiments

The following sections present experiments that highlight properties of the J_{χ^2} importance measure. They show that J_{χ^2} actually provides an interpretable importance index (Section 4.1), and that it is closely related to J_a both in terms of variable rankings (Section 4.2) and predictive performances when used as feature selection pre-filter (Section 4.3). The last experiments in Section 4.4 present some predictive performances when restricting models to only statistically significant variables.

4.1 Interpretability of J_{χ^2}

The main goal of the new feature importance measure is to provide an interpretable index allowing to retrieve variables that are significantly important in the prediction of the forest. In order to check that J_{χ^2} is able to identify those variables, first experiments are conducted on an artificial dataset with a linear decision boundary. This dataset is generated the same way as in [4]. Labels $y \in \{-1, 1\}^n$ are given by $y = \text{sign}(Xw)$ where $w \in \mathbb{R}^p$ and $X \in \mathbb{R}^{n \times p}$. Data values come from a $\mathcal{N}(0, 1)$ distribution. The number p of variables is set to 120. The first 20 weights w_i are randomly sampled from $\mathcal{U}(0, 1)$. The other 100 weights are set to 0 such that relevant variables only belong to the first 20 ones (but all these variables need not be relevant *e.g.* whenever a weight is very small). The number of instances is $n = 500$ such that $X \in \mathbb{R}^{500 \times 120}$. In order to add some noise, 10% of the labels are randomly flipped.

To check that a feature selection technique is able to identify significant variables, we report the observed False Discovery Rate (FDR) as in [4]:

$$FDR = \frac{FD}{FD + TD} \tag{5}$$

where FD is the number of false discoveries (*i.e.* variables which are flagged as significantly important by the feature importance index but that are actually not important) and TD the number of true discoveries. A good variable importance index should yield a very low observed FDR.

A RF, built on the full dataset, is used to rank the variables according to their importance index. In order to decide if a variable is significantly important, we fix the p-value threshold to the commonly accepted 0.05 value after correcting for multiple testing. Figure 1 shows importance indices obtained by forests of various sizes and different numbers m of variables randomly sampled as candidate in each tree node. As we can see, the traditional (decreasing) J_a index does not offer a clear threshold to decide which variables are relevant or not. Similarly to the methods presented in [4], the (increasing) J_{χ^2} index appears to distinguish more clearly between relevant and irrelevant variables. It however requires a relatively large number of trees to gain confidence that a feature is relevant. When computed on small forests (plots on the left), J_{χ^2} may fail to identify variables as significantly important but they are still well ranked as shown by the FDR values. Moreover, increasing the parameter m also tends to positively impact the identification of those variables when the number of trees is low.

4.2 Concordance with J_a

As explained in Section 3.1, J_{χ^2} and J_a share a lot in their computations. Figure 2 compares the rankings of the two importance measures on one sampling of the microarray DLBCL[5] dataset ($p = 7129$, class priors = 58/19). It shows that variable ranks in the top 500 are highly correlated. Spearman's rank correlation coefficient is 0.97 for those variables. One of the main differences between the rankings produced by J_a and J_{χ^2} is that the first one penalizes features whose

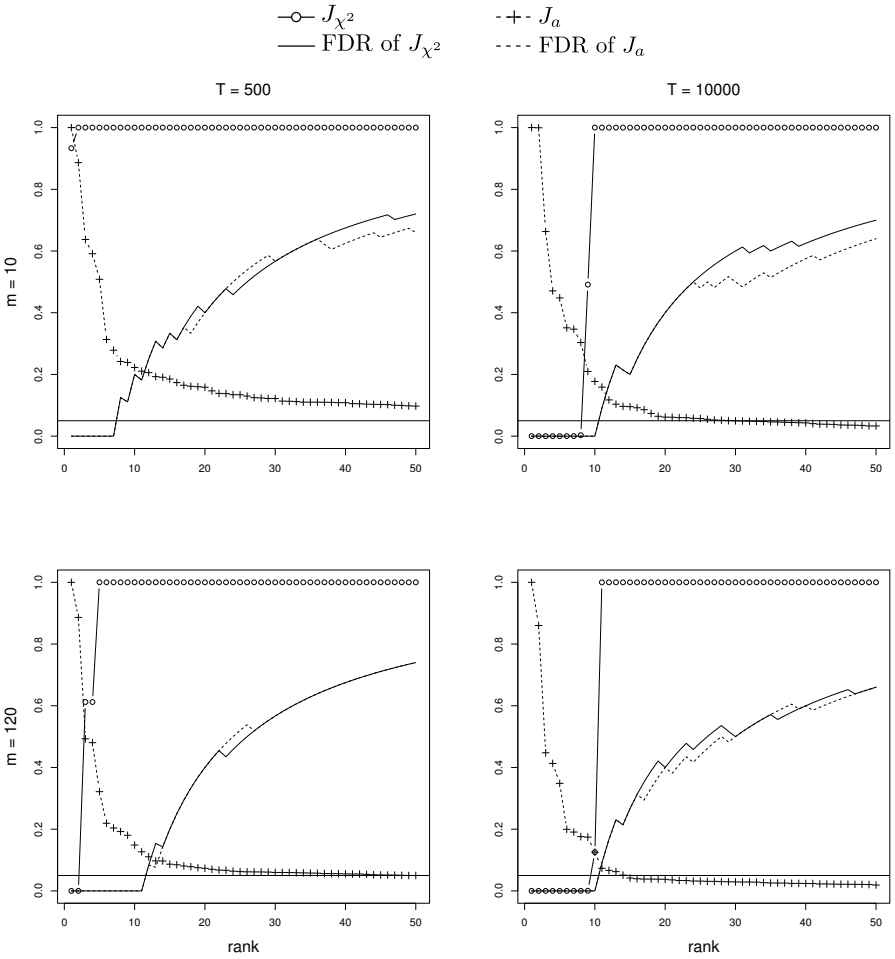

Fig. 1. Importance indices computed on an artificial dataset with a linear decision boundary. For the sake of visibility, J_a has been rescaled between 0 and 1. The horizontal line is set at 0.05. $J_{\chi^2}(x_j)$ below this line are deemed statistically relevant.

permuted versions would increase the prediction accuracy while the second one would favour such a variable since it changes the class vote distribution. That explains why features at the end of J_a's ranking have a better rank with J_{χ^2}. In particular, after rank 1250 on the horizontal axis, features have a negative J_a value for they somehow lower the prediction performance of the forest. But, since they influence the class vote distribution, they are considered more important by J_{χ^2}. Although those differences are quite interesting, the large ranks of those variables indicates that they encode most probably nothing but noise. Furthermore, only top ranked features are generally interesting and selected based on their low corrected p-values.

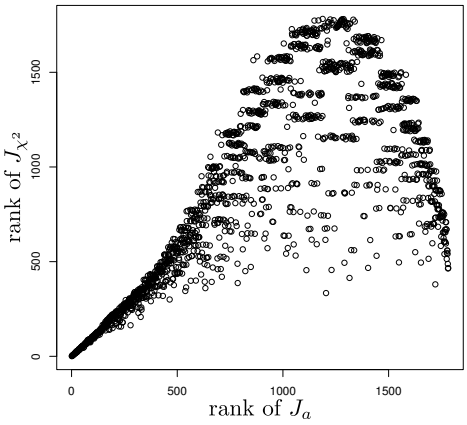

Fig. 2. Rankings produced by J_a and J_{χ^2} on one external sampling of the DLBCL dataset

4.3 Feature Selection Properties

As shown in section 4.2, J_a and J_{χ^2} provide quite correlated variable rankings. The experiments described in this section go a little bit deeper and show that, when used for feature selection, the properties of those two importance indices are also very similar in terms of prediction performances and stability of the feature selection.

In order to measure the predictive performances of a model, the Balanced Classification Rate (BCR) is used. It can be seen as the mean of per-class accuracies and is preferred to accuracy when dealing with non-balanced classes. It also generalizes to multi-class problems more easily than AUC. For two class problems, it is defined as

$$BCR = \frac{1}{2}\left(\frac{TP}{P} + \frac{TN}{N}\right) \tag{6}$$

Stability of feature selection indices aim at quantifying how much selected sets of features vary when little changes are introduced in a dataset. The Kuncheva index (KI) [6] measures to which extent K sets of s selected variables share common elements.

$$KI(\{S_1, ..., S_K\}) = \frac{2}{K(K-1)} \sum_{i=1}^{K-1} \sum_{j=i+1}^{K} \frac{|S_i \cap S_j| - \frac{s^2}{p}}{s - \frac{s^2}{p}} \tag{7}$$

where p is the total number of features and $\frac{s^2}{p}$ is a term correcting the chance to share common features at random. This index ranges from -1 to 1. The greater the index, the greater the number of commonly selected features. A value of 0 is the expected stability for a selection performed uniformly at random.

In order to evaluate those performances and to mimic little changes in datasets, an *external* resampling protocol is used. The following steps are repeated 200 times:

Randomly select a training set Tr made of 90% data. The remaining 10% form the test set Te.

- train a forest of T trees to rank the variables on Tr
- for each number of selected features s
  - train a forest of 500 trees using only the first s features on Tr
  - save the BCR computed on Te and the set of s features

The statistics recorded at each iteration are then aggregated to provide mean BCR and KI.

Figure 3 presents the measurements made over 200-resamplings from the DLBCL dataset according to the number of features kept to train the classifier. It shows that the two indices behave very similarly with respect to the number of features and the number of trees used to rank the features. Increasing the number of trees allows to get more stable feature selection in both cases. This kind of behaviour has also been shown in [7].

4.4 Prediction from Significantly Important Variables

Experiments show that J_{χ^2} ranks features roughly the same way as J_a while providing a statistically interpretable index. One can wonder if it is able to highlight important variables on real-world datasets and furthermore if those variables are good enough to make a good prediction by themselves. Table 1 briefly describes the main characteristics of 4 microarray datasets used in our study.

Using the same protocol as in Section 4.3, experiments show that the number of selected variables increases with the number of trees, which is consistent with the results in Section 4.1. As we can see on Table 2, it is also very dataset dependent with almost no features selected on the DLBCL dataset. Similar results are observed in [4]. When comparing the predictive performances of a model built on only significant variables of J_{χ^2} and a model built using the 50 best ranked

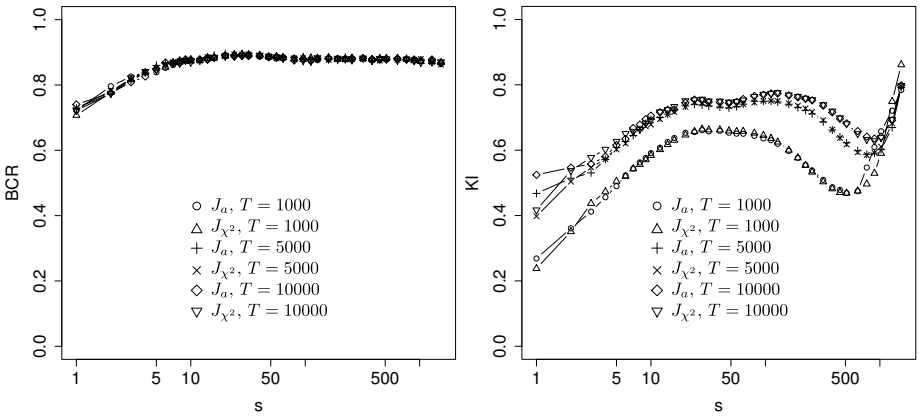

Fig. 3. Average BCR and KI of J_a and J_{χ^2} over 200 resamplings of the DLBCL dataset according to the number of selected features, for various numbers T of trees

Table 1. Summary of the microarray datasets: class priors report the n values in each class, p represents the total number of variables.

Name	Class priors	p
DLBCL [5]	58/19	7129
Lymphoma [8]	22/23	4026
Golub [9]	25/47	7129
Prostate [10]	52/50	6033

variables of J_a, a paired T-test shows significant differences in most of the cases. However, except for the DLBCL dataset, when using 10000 trees, the average predictive performances are quite similar to each other. This confirms that, provided the number of trees is large enough and depending on the dataset, J_{χ^2} is able to select important variables that can be used to build good predictive models.

Table 2. Various statistics obtained over 200-resamplings when keeping only significantly relevant variables. T is the number of trees used to build the forest. *avg*(s^{rel}) (resp. *max*, *min*) is the average (resp. maximum, minimum) number of J_{χ^2} significantly important features used to make the prediction. *BCR* is the average BCR obtained on models for which there is at least one significant feature with J_{χ^2}. BCR^{50} is the average BCR obtained when using the 50 J_a best ranked features in each iteration where J_{χ^2} outputted at least one significant feature.

	T	avg(s^{rel})	min(s^{rel})	max(s^{rel})	BCR	BCR^{50}
DLBCL	5000	0.04	0	1	0.52	0.67
	10000	0.99	0	5	0.69	0.83
golub	5000	5.96	3	10	0.93	0.97
	10000	10.82	8	14	0.96	0.97
lymphoma	5000	0.66	0	6	0.62	0.82
	10000	4.85	2	9	0.93	0.94
prostate	5000	4.95	2	8	0.93	0.94
	10000	7.92	6	11	0.93	0.94

5 Conclusion and Perspectives

This paper introduces a statistical feature importance index for the Random Forest algorithm which combines easy interpretability with the multivariate aspect of embedded feature selection techniques. The experiments presented in Section 4 show that it is able to correctly identify important features and that it is closely related to Breiman's importance measure (mean decrease in accuracy after permutation). The two approaches yield similar feature rankings. In comparison to Breiman's importance measure, the proposed index J_{χ^2} brings the interpretability of a statistical test and allows us to decide which variables are significantly important using a very natural threshold at the same computational cost.

We show that growing forests with many trees increases the confidence that some variables are statistically significant in the RF voting process. This observation may be related to [7] where it is shown that feature selection stability of tree ensemble methods increases and stabilises with the number of trees. The proposed importance measure may open ways to formally analyse this effect, similarly to [2]. We have evaluated J_{χ^2} on binary classification tasks. Although there is a straightforward way to adapt it to the multi-class setting, future work

should assess whether it is practically usable, in particular how many trees would be needed when increasing the number of classes. Finally one should also evaluate the possibility to apply this approach on other ensemble methods, possibly with different kinds of randomization.

References

1. Breiman, L.: Random Forests. Machine Learning **45**(1) (October 2001) 5–32
2. Hernández-Lobato, D., Martínez-Muñoz, G., Suárez, A.: How large should ensembles of classifiers be? Pattern Recognition **46**(5) (2013) 1323 – 1336
3. Benjamini, Y., Hochberg, Y.: Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. Journal of the Royal Statistical Society. Series B (Methodological) **57**(1) (1995) 289–300
4. Huynh-Thu, V.A.A., Saeys, Y., Wehenkel, L., Geurts, P.: Statistical interpretation of machine learning-based feature importance scores for biomarker discovery. Bioinformatics (Oxford, England) **28**(13) (July 2012) 1766–1774
5. Shipp, M.A., Ross, K.N., Tamayo, P., Weng, A.P., Kutok, J.L., Aguiar, R.C., Gaasenbeek, M., Angelo, M., Reich, M., Pinkus, G.S., Ray, T.S., Koval, M.A., Last, K.W., Norton, A., Lister, T.A., Mesirov, J., Neuberg, D.S., Lander, E.S., Aster, J.C., Golub, T.R.: Diffuse large b-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. Nat Med **8**(1) (January 2002) 68–74
6. Kuncheva, L.I.: A stability index for feature selection. In: AIAP'07: Proceedings of the 25th conference on Proceedings of the 25th IASTED International Multi-Conference, Anaheim, CA, USA, ACTA Press (2007) 390–395
7. Paul, J., Verleysen, M., Dupont, P.: The stability of feature selection and class prediction from ensemble tree classifiers. In: ESANN 2012, 20th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. (April 2012) 263–268
8. Alizadeh, A.A., Eisen, M.B., Davis, R.E., Ma, C., Lossos, I.S., Rosenwald, A., Boldrick, J.C., Sabet, H., Tran, T., Yu, X., Powell, J.I., Yang, L., Marti, G.E., Moore, T., Hudson, J., Lu, L., Lewis, D.B., Tibshirani, R., Sherlock, G., Chan, W.C., Greiner, T.C., Weisenburger, D.D., Armitage, J.O., Warnke, R., Levy, R., Wilson, W., Grever, M.R., Byrd, J.C., Botstein, D., Brown, P.O., Staudt, L.M.: Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature **403**(6769) (February 2000) 503–511
9. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., et al.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. science **286**(5439) (1999) 531–537
10. Singh, D., Febbo, P.G., Ross, K., Jackson, D.G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A.A., D'Amico, A.V., Richie, J.P., et al.: Gene expression correlates of clinical prostate cancer behavior. Cancer cell **1**(2) (2002) 203–209

Acknowledgements Computational resources have been provided by the supercomputing facilities of the Université catholique de Louvain (CISM/UCL) and the Consortium des Equipements de Calcul Intensif en Fédération Wallonie Bruxelles (CECI) funded by the Fond de la Recherche Scientifique de Belgique (FRS-FNRS).