Evaluation Method for Feature Rankings and their Aggregations for Biomarker Discovery

Ivica Slavkov IVICA.SLAVKOV@IJS.SI
Department of Knowledge Technologies
Jožef Stefan Institute
Jamova cesta 39, 1000 Ljubljana, Slovenia

Bernard Ženko BERNARD.ZENKO@IJS.SI
Department of Knowledge Technologies
Jožef Stefan Institute
Jamova cesta 39, 1000 Ljubljana, Slovenia

Sašo Džeroski SASO.DZEROSKI@IJS.SI
Department of Knowledge Technologies
Jožef Stefan Institute
Jamova cesta 39, 1000 Ljubljana, Slovenia

Editor: Sašo Džeroski, Pierre Geurts, and Juho Rousu

Abstract

In this paper we investigate the problem of evaluating ranked lists of biomarkers, which are typically an output of the analysis of high-throughput data. This can be a list of probes from microarray experiments, which are ordered by the strength of their correlation to a disease. Usually, the ordering of the biomarkers in the ranked lists varies a lot if they are a result of different studies or methods. Our work consists of two parts. First, we propose a method for evaluating the "correctness" of the ranked lists. Second, we conduct a preliminary study of different aggregation approaches of the feature rankings, like aggregating rankings produced from different ranking algorithms and different datasets. We perform experiments on multiple public Neuroblastoma microarray studies. Our results show that there is a generally beneficial effect of aggregating feature rankings as compared to the ones produced by a single study or single method.

Keywords: feature ranking evaluation, biomarker discovery, ranking aggregation

1. Introduction

In medicine, the progress or presence of some disease is determined by measuring certain biological parameters. These parameters are commonly called biomarkers and can range from blood pressure to the expression of a certain gene. Here, we focus on biomarkers derived from different types of high-throughput data.

We consider the process of biomarker discovery as the process of determining markers which have the strongest correlation to the presence or status of a certain disease. For example, given a microarray experiment, the output would be a list of probes ranked according to their differential expression. The main challenge in biomarker discovery from

high dimensional data arises from having a small number of available biological samples, as well as from the inherent high variability of the data.

In machine learning terminology, biomarker discovery translates into the task of feature ranking and feature selection. Although these two tasks are related, they produce different result. On one hand, feature ranking provides an assessment of the "importance" of individual features to a target concept. On the other hand, feature selection algorithms evaluate the "importance" of a subset of features as a whole. This does not mean that all (or any) of the features in the subset have high individual importance. In the context of biomarker discovery, the task of feature selection would be more appropriate for diagnostic markers while feature ranking would be more useful when searching for individual drug targets.

The estimation of importance in feature selection and feature ranking is different. In feature selection, the feature subsets are evaluated explicitly via a predictive model (classifier), built from just those features. As for feature ranking, there is no direct way of evaluating the "correctness" of the order of the individual features. Therefore, our work in this paper focuses on developing an evaluation methodology for feature rankings.

We present our work as follows: First, in Section 2 we define the problem under consideration. We then propose and describe our evaluation methodology in Section 3, where we also consider different approaches of aggregating feature rankings. In Section 4 we outline the experimental evaluation and provide description of the data used. The outcome of the experiments is presented in Section 5. Finally, we discuss the results and draw some conclusions in Section 6.

2. Problem description

We formalize the problem setting as follows: given is dataset D, consisting of k instances (samples) $D = \{S_1, S_2, ..., S_k\}$. Each sample is a vector of n values, $S_i = (v_{i1}, v_{i2}, ...v_{in})$. Each value of an instance represents a certain property or a so-called feature f of that instance. Each feature has a specific value for a specific sample, i.e., $f_j(S_i) = v_{ij}$. Simply put, each row in a dataset is an instance S_i, and each column is the vector of values of a feature f_j.

In this kind of a setting, a feature of particular interest is called a target feature f_{target}, for example the status of some disease. If we apply on the dataset D a ranking algorithm $R(D, f_{target})$, it outputs a list of features $F = [f_1, ..., f_n]$, ordered by decreasing importance $Imp(f_j)$ with respect to f_{target}. The function $Imp(f_j)$ is different for different ranking methods.

In this paper we would like to evaluate how correct is the ordering of features in the ranked list, considering that we never know the ground truth ranking. We will refer to this problem as a problem of evaluating *feature rankings*. This kind of an evaluation methodology, in terms of biomarker discovery, would help answer the question: Which ranking method and/or which study, produce the most "correct" ranked list of genes?

3. Methodology

In this section we present our proposed methodology for evaluating feature rankings. We begin by more formally describing our approach and we also briefly discuss the issue of aggregating feature rankings.

3.1 Error curve

We approach the problem of evaluating feature rankings by following the idea that the "correctness" of the feature rank is related to predictive accuracy. A good ranking algorithm would put on top of a list a feature that is most important, and at the bottom a feature that is least important w.r.t. some target concept. All the other features would be in-between, ordered by decreasing importance. By following this intuition, we evaluate the ranking by performing a stepwise feature subset evaluation, with which we generate a so-called *error curve*.

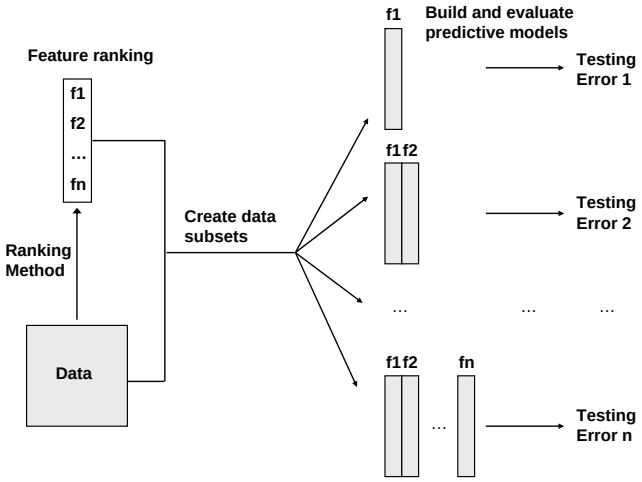


Figure 1: Constructing an error curve

We present the process of generating the error curve on Figure 1. We begin with a dataset D on which we apply an arbitrary ranking algorithm R. This produces a feature ranking $F = [f_1, ..., f_n]$, where f_1 denotes the top-ranking feature and f_n the bottom-ranked one. We then proceed by generating n data subsets $\{D_{f_{1..1}}, D_{f_{1..2}}..., D_{f_{1..n}}\}$ from the original dataset D. We construct the first data subset $D_{f_{1..1}}$ with only the top-ranked feature f_1. We then add to this subset the second ranked feature f_2, denoted by $D_{f_{1..2}}$. This process is continued iteratively until we add the bottom ranked feature f_n to the $D_{f_{1..n-1}}$ subset, thus yielding $D_{f_{1..n}}$. Finally, we build n predictive models from each of the data subsets and we estimate their error. The points of the error curve are each of the n estimated errors $[E_1, ..., E_n]$. This procedure is summarized in Table 1.

3.2 Aggregating feature rankings

We consider aggregating feature rankings an important practical issue when working with high-dimensional data. Considering the plethora of feature ranking methods and datasets

Table 1: Constructing an error curve

Input: Data D, Ranking method R
Output: Error curve E
$E \Leftarrow \emptyset$
$D_{f_{1..0}} \Leftarrow \emptyset$
$F \Leftarrow FeatureRanking(R, D)$
for $i = 1$ to n **do**
$D_{f_{1..i}} \Leftarrow D_{f_{1..i-1}} \cup f_i$
$P_i \Leftarrow BuildPredictiveModel(D_{f_{1..i}})$
$E \Leftarrow E \cup EstimateError(P_i)$
end for
return E

that are available, it is reasonable to assume that it might be beneficial to join the different information (rankings) that they provide.

When aggregating feature rankings, there are two issues to consider. The first one is which base feature rankings to aggregate. There are different ways to generate the base feature rankings: from the same dataset, but by different ranking method; from different datasets but the same ranking method or from different subsamples of the same dataset and the same ranking method. The second issue concerns the type of aggregation function to use. Many functions are available, and we believe that this is a topic worth exploring by itself, which is out of the scope of this paper. For our initial experiments we used simple methods, like taking the mean or median of the ranks.

3.3 Error estimation

An important aspect of the evaluation methodology is how the testing error is estimated. A commonly used approach to estimate model error is the well known cross-validation procedure. In a study by Kohavi (1995), it was concluded that in general, the best error estimate can be obtained from ten fold stratified cross-validation.

Considering that we are working with high-dimensional data and with a small number of data samples (instances), a ten fold cross-validation is not the best solution. With such small number of data instances, the best way of obtaining an unbiased error estimate is to perform a leave-one-out cross validation (LOOCV), where the number of folds equals the number of instances in the data. But, as noted in Efron and Tibshirani (1997) and Kohavi (1995), the error estimate obtained by LOOCV has high variance. In order to account for this the so called ".632+ Bootstrap" method is proposed in Efron and Tibshirani (1997). This method has been previously used for estimating the error from microarray experiments, for example in Ambroise and McLachlan (2002).

In short, the .632+ bootstrap method effectively smooths the variance of the LOOCV error estimate by using bootstrapping. The exact way that this method works is presented in Table 2.

The error estimation starts as an ordinary leave-one-out cross validation. One instance is excluded from the original data which gives the training fold and the left-out instance

Table 2: The .632+ Bootstrap method

Input: Data D, Number of bags b
Output: Testing error Err
  $folds \Leftarrow NumberOfInstances(D)$
  $CV_{err} \Leftarrow \emptyset$
  for $i = 1$ to $folds$ **do**
    $Fold_{train} \Leftarrow D - Instance_i(D)$
    $B_{err,i} \Leftarrow \emptyset$
    for $j = 1$ to b **do**
      $B_{i,j} \Leftarrow BootstrapResampling(Fold_{train})$
      $P_{i,j} \Leftarrow BuildPredictiveModel(B_{i,j})$
      $B_{err,i} \Leftarrow B_{err,i} \cup TestingError(P_{i,j}, Instance_i(D))$
    end for
    $CV_{err} \Leftarrow CV_{err} \cup Average(B_{err,i})$
  end for
  $Err \Leftarrow Calculate632Error(Average(CV_{err}))$
  return Err

is used for testing. We then produce b bootstrap replicates of the training fold by using re-sampling with replacement. We proceed by building predictive models for each bootstrap replicate and we estimate their error $B_{err,i,j}$ on the left-out instance. The errors are then averaged and we obtain estimated error for just one fold $Fold_{err,i}$ of the cross-validation. The average error Err_{cv} is calculated as the mean of all of the fold errors. The final ".632 error" is determined as:

$$Err_{632} = (1 - w) \cdot Err_{train} + w \cdot Err_{cv} \tag{1}$$

where the weight w is calculated as:

$$w = \frac{0.632}{1 - 0.368 \cdot r}, \tag{2}$$

and r is the relative overfitting rate calculated as:

$$r = \frac{Err_{cv} - Err_{train}}{\gamma - Err_{train}}, \tag{3}$$

and γ is the so-called no-information error rate. In theory, γ is the error rate if the features and the target concept of the dataset are independent. If we assume a multi-class classification problem then it is calculated as:

$$\gamma = \sum_{i=1}^{c} p_i \cdot (1 - q_i), \tag{4}$$

where c is the number of classes, while p_i and q_i are the proportions of the original and predicted class distribution (correspondingly) of the i_{th} class.

So far, we have presented the .632+ Bootstrap method when it is used for estimating the error of just a single predictive model. It should be noted that when generating the TE curve, this method is repeatedly used to estimate the error for each point.

4. Experimental setup

This section concerns the experimental setup used in order to demonstrate the application of our evaluation method. We first present the data used for the experiments, and then we proceed with the description of the experimental design. The main idea behind the experimental setup is to evaluate and compare the behavior of different ranking algorithms and different aggregation methods, on single studies and as well across studies. We also investigate the way the error curve changes when we consider different predictive models for estimating the error.

4.1 Data description

We performed our experiments on Neuroblastoma studies. Neuroblastoma is the most common extracranial solid tumor of childhood. We considered the status of relapse/no relapse of a patient, as a target concept of interest. The derived markers could be useful for determining the course of treatment of a patient.

We focus on three Affymetrix microarray studies, namely: De Preter et al. (2007) (17 samples), Schramm et al. (2005) (63 samples) and Wang et al. (2006) (99 samples). For practical purposes when presenting the results we refer to them as the "D", "S" and the "W" study.

4.2 Experimental design

We can divide our experiments in two parts: individual study evaluation and cross-study evaluation.

In the individual study setting, we focus on comparing the performance of different ranking approaches. We considered four different feature ranking methods: a simple method based on information gain and more complex methods like random forests (Breiman (2001)), the ReliefF algorithm (Kononenko (1994)) and SVM (Guyon et al. (2002)). All of these methods, have very different approaches for determining the feature importance and are therefore interesting to compare.

Random forests use ensemble learning for evaluating the feature importance. After each random tree is constructed, its performance (misclassification rate) is evaluated on the original out-of-bag data. Then the values of each feature (one feature at a time) are randomly permuted at the out-of-bag data. On these modified sets, the misclassification rate of the original random trees is evaluated. At the end the importance of the feature is the increase of misclassification rate as compared to the original out-of-bag rate (with all features intact).

The ReliefF algorithm has a very intuitive way of assessing the feature importance. It's main idea is to evaluate features according to how well they distinguish between instances close to each other. The rationale is that for a given instance I of a class C, good features should differentiate between instances of different class(es) I_{-c}, but at the same time have similar values for instances of the same class I_c. The importance of a feature f is then calculated as a sum of differences for each instance I and each class C as:

$$Imp(f) = \sum_{I} (\sum_{I_{-c}} difference(f(I), f(I_{-c})) - \sum_{I_c} difference(f(I), f(I_c))) \qquad (5)$$

Table 3: Cross-study evaluation

$S \Rightarrow D$	$D \Rightarrow S$	$D \Rightarrow W$
$W \Rightarrow D$	$W \Rightarrow S$	$S \Rightarrow W$
$agg\{S, W\} \Rightarrow$ D	$agg\{D, W\} \Rightarrow$ S	$agg\{D, S\} \Rightarrow$ W

The SVM-RFE algorithm couples recursive feature elimination (RFE) with SVMs. It uses the SVMs weight magnitude as a feature importance criterion. Initially it builds a SVM on all the features and finds the feature with the smallest weight. This feature is given the lowest rank and is eliminated from the feature set. Using the new feature set, a SVM is again trained and the procedure of elimination is repeated. This iterative procedure is continued until there are no more features in the feature set, i.e., each feature has an assigned rank (importance).

Furthermore, we also investigated if it is beneficial to aggregate the feature rankings produced by different methods on the same study, intuitively similar to Saeys et al. (2008) and Jong et al. (2004). We considered simple aggregation methods as the Mean rank, Median rank, as well as Min and Max rank.

When investigating the cross-study setting, we considered only one ranking method, namely ReliefF. The idea initially is to compare how feature rankings learned on one study behave if they are tested on another study. Then we examine how that compares to aggregating feature rankings from two different studies and testing on the third.

We summarize the cross-study setting in Table 3. We use "D","S" and "W" to denote different studies and "$A \Rightarrow B$" to signify that we build the feature ranking on study "A" and evaluate it on study "B". When aggregating the feature rankings from the different studies ($agg\{...\}$), we used the previously mentioned aggregation methods.

In both experimental settings, for estimating the error we used the .632+ Bootstrap method, as explained in Section 3.3. In our experiments, we use 20 bags (bootstrap resampling), which was previously empirically estimated.

We have used different predictive models when constructing the error curve, namely: Naïve Bayes, random forests, decision trees and SVMs. We present the comparison of the error curves (for the Wang dataset) in Section 5.3 followed by a short discussion. For the single and cross-study setting we present only the Naïve Bayes error curves.

5. Results

In this section we present the results from the previously described experimental evaluation. We first consider the results from the comparison of different ranking algorithms and different aggregation methods on single datasets. We then presents the effects of combining feature rankings from different studies from the cross-study setting. The results of comparing different predictive models is given as the last subsection of the results.

5.1 Individual studies

We present the testing error curves from the single study experiments on Figure 2. On the left-hand side, we show the comparison between the different ranking algorithms, while

on the right-hand side the error curves of different aggregation methods are shown. The figures are ordered in such a way that the results for the smallest dataset (De Preter) are the first figures in a column, while for the largest one (Wang) the results are the last ones in a column.

If we first consider the comparison of different ranking algorithms, it is not immediately obvious which one performs the best. However, it seems that SVM-RFE and ReliefF seem to produce the best ranking, according to the error curves. Also, there is a noticeable effect of the dataset size, where the biggest difference in the curves is for the smallest (De Preter) dataset. Furthermore, if we take a look at the comparisons between the different ranking aggregation methods, the median method has an overall "better" error curve. The median error curve is comparable to the individual ranking algorithms, but it is noticeably less variable.

5.2 Cross studies

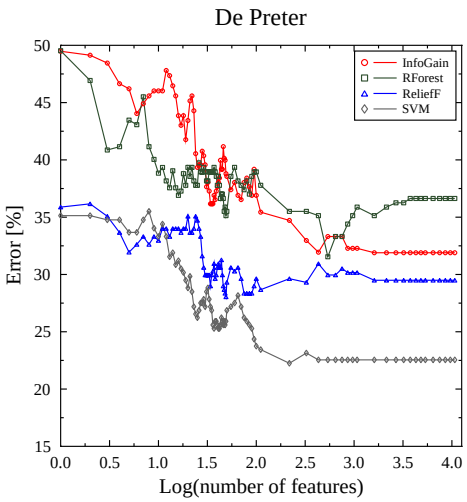
In a similar fashion, we present the results from the cross-studies experiments on Figure 3. The results from the different aggregation methods that are used for combining the feature rankings from the different studies are on the right-hand side figures ((b), (d) and (f)). The comparison between the single study feature ranking and the best aggregated feature ranking, tested on a separate study, are presented on the left-hand side ((a), (c) and (e)). The ordering according to dataset size, also applies here.

The comparison between the different aggregation methods, does not reveal a noticeable difference, although when testing on smaller studies there is great variability of the error curves as compared to testing on bigger studies.

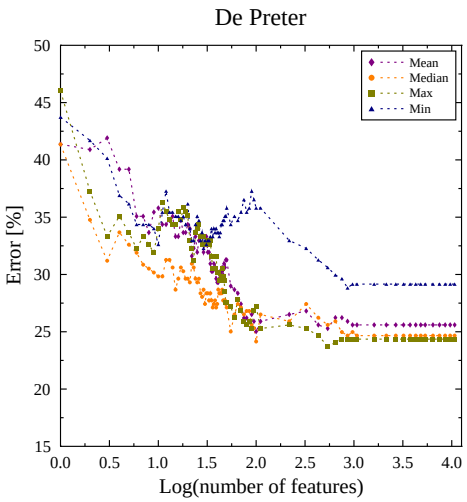
If we take a look at Figure 3(a), it compares between three different feature rankings tested on the De Preter dataset. The feature ranking from the biggest dataset (Wang) is better, but it is worse than the feature ranking produced by aggregating the two different rankings from the Schramm and Wang datasets.

When testing on the Schramm dataset (Figure 3(c)), the feature ranking from the smallest dataset (De Preter), performs obviously much worse than the one derived from the biggest dataset (Wang). However, aggregating the feature rankings also does not produce a better ranking. We believe that this is due to the fact that when combining the feature rankings from the two studies, the De Preter derived one is of much worse quality and therefore it has a detrimental effect on the overall aggregated rank.

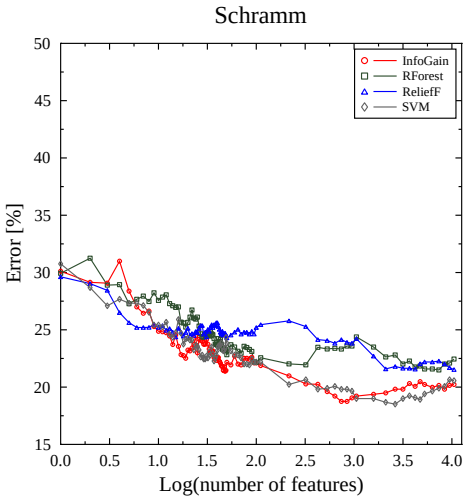
Finally, we show the error curves, when testing on the Wang dataset (Figure 3(e)). On first look, the error curve of the feature ranking derived from the aggregation, seems to be somewhat better than the others. Although a little after the beginning of the curves the error seems to be the same, the curve from the aggregated feature rankings is much less variable than the others. Also it seems that at a very later stage it improves, which we think is due to aggregating an unreliable feature ranking derived from a particularly small dataset.



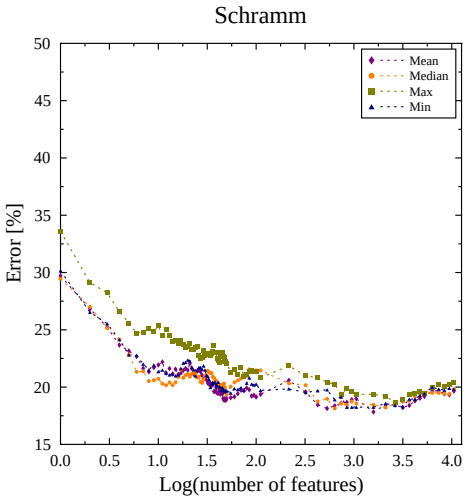
(a) Different ranking methods



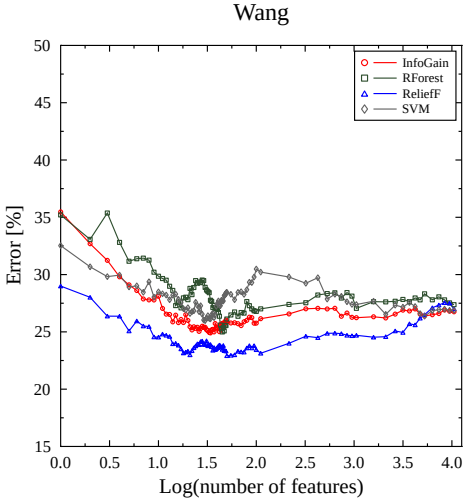
(b) Different aggregation methods



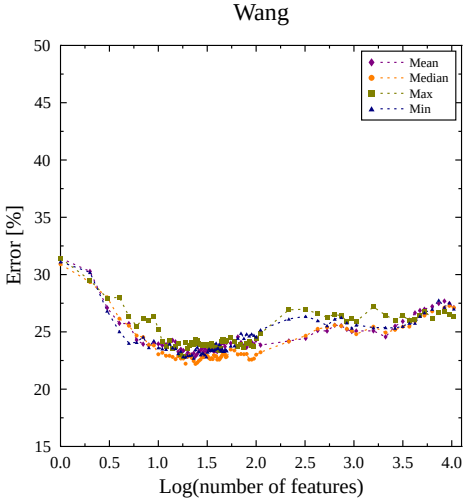
(c) Different ranking methods



(d) Different aggregation methods

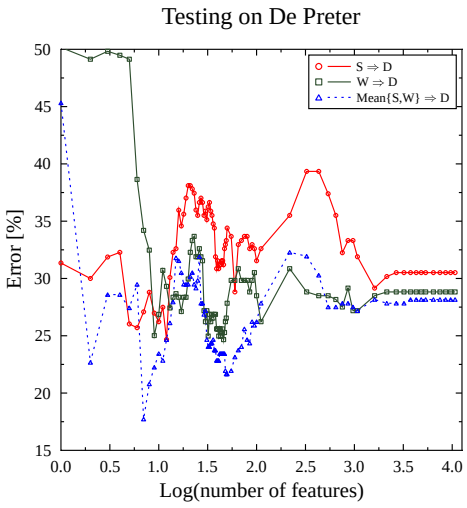


(e) Different ranking methods

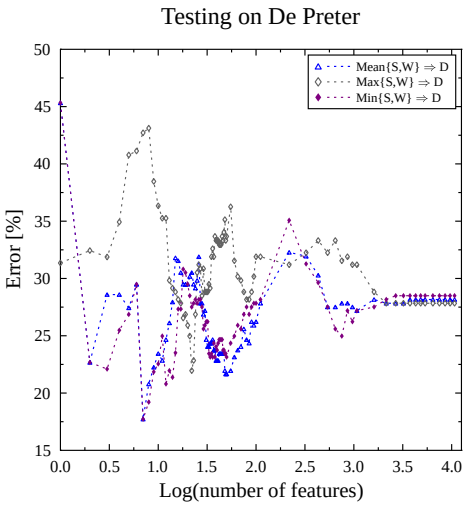


(f) Different aggregation methods

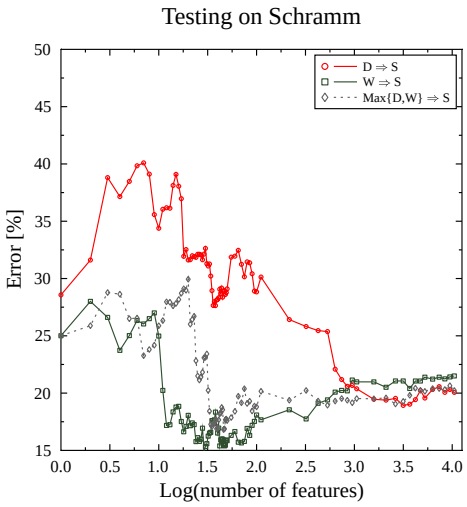
Figure 2: Single study comparisons



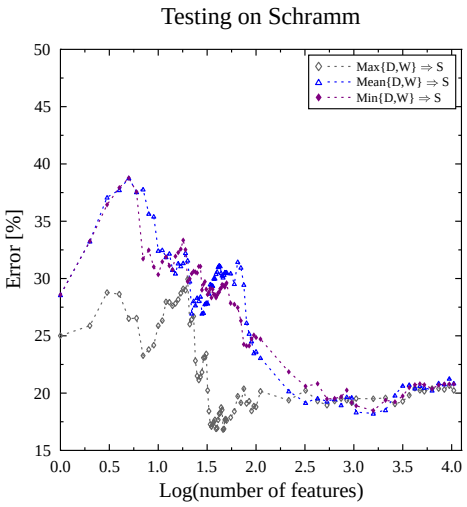
(a) Single vs. combined studies



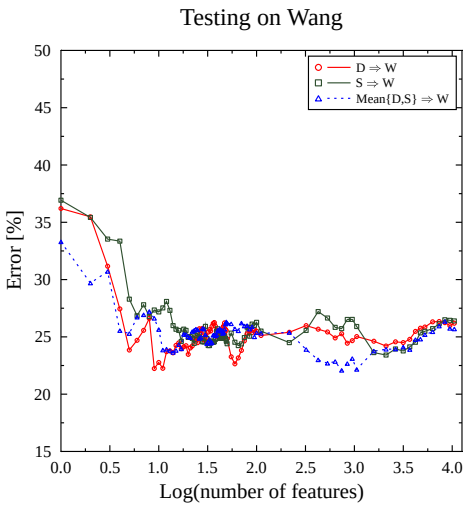
(b) Different aggregation methods



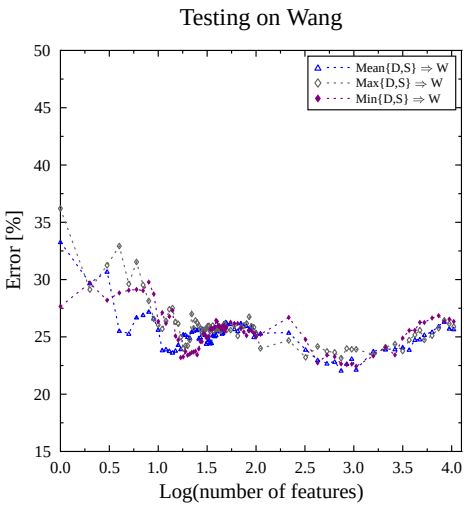
(c) Single vs. combined studies



(d) Different aggregation methods



(e) Single vs. combined studies

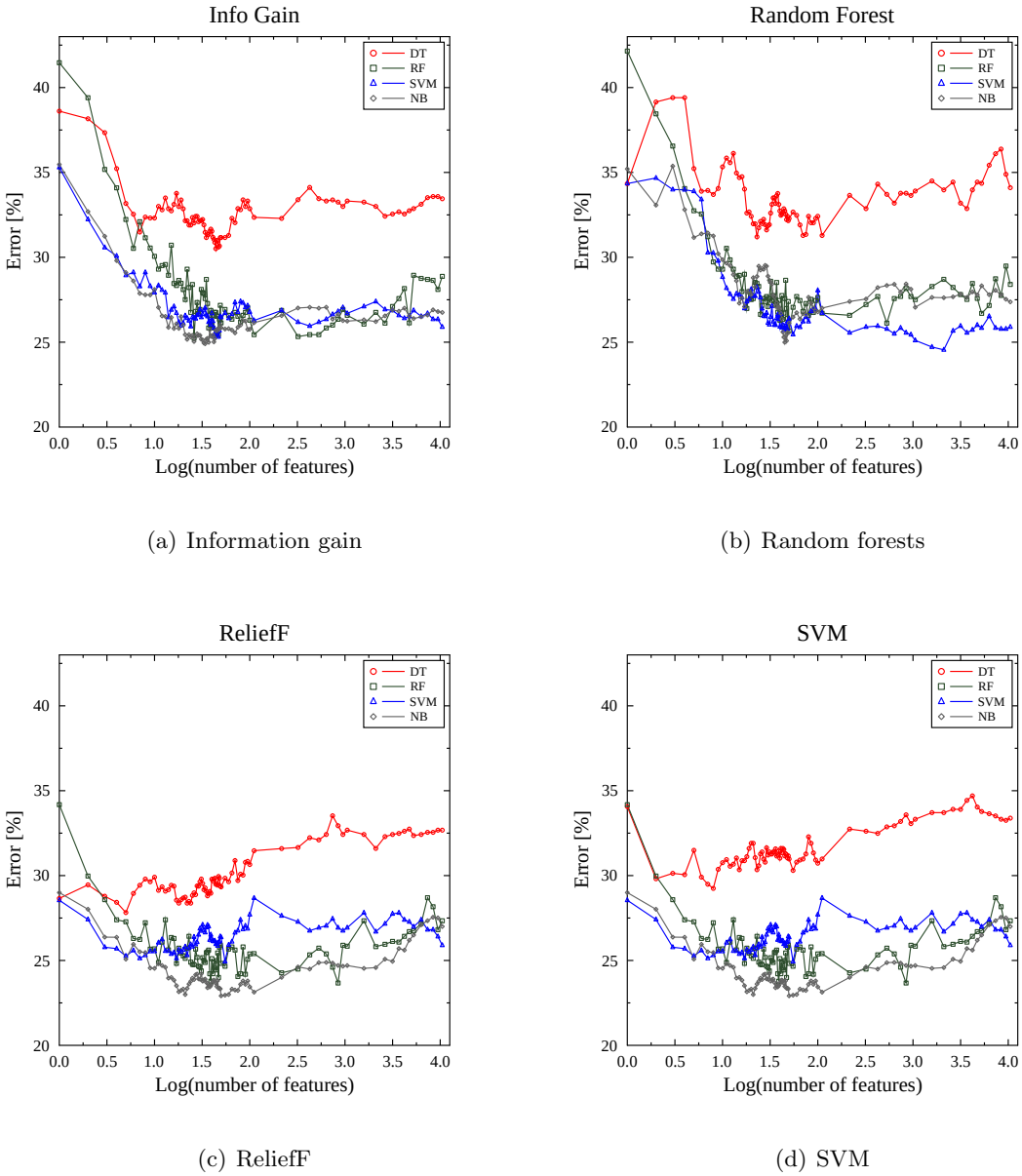


(f) Different aggregation methods

Figure 3: Cross study comparisons

5.3 Comparing different predictive models

Here we present the error curves with the error values estimated by using different predictive models (Figure 4). The four different graphs are for four different ranking methods. Each graph contains error curves for the previously mentioned predictive models (decision trees, Naïve Bayes, random forests and SVMs).



(a) Information gain

(b) Random forests

(c) ReliefF

(d) SVM

Figure 4: Comparison of different predictive models

Concerning the underlying classifiers we can notice the unusual error curve with values estimated by using decision trees. The error seems to be always higher, as compared to the error estimated by other predictive models and it is increasing rather then decreasing when using ReliefF and SVMs as underlying ranking methods (Figure 4. (c) and (d)). The error curve estimates from the other predictive models seem to be behaving similarly and their error size is comparable.

The four ranking methods can be clearly divided into two groups. Information gain and random forests rankings produce error curves which never go below 25%, while ReliefF and SVMs produce rankings which go down below this value. Note that the ReliefF ranking algorithm is much faster than SVM-RFE and the lowest error is produced as a combination of all the different ranking methods with the Naïve Bayes classifier.

6. Conclusions and further work

In this paper we presented a methodology for evaluating feature rankings. The method relates the "correctness" of the feature ranking to the notion of error of predictive models. We use the so-called error curve, constructed as described in Section 3.1, as an indicator for the quality of the produced feature rankings.

Furthermore, the developed method is used for comparing different ranking approaches and different aggregation approaches for combining feature rankings. From the results presented in Section 5 we can discern two interesting points.The first is related to the size of the error of the curves and the second is related to the variability of the error curves.

Concerning the error size, it is difficult to say with certainty which one is the best feature ranking method or aggregation approach. However, for the ranking methods, it seems that ReliefF and SVMs have the lowest errors. When aggregating feature rankings from different methods, the median aggregation function seems to have the lowest error. The differences in error are very much related to the study size, where bigger differences between ranking algorithms appear for smaller dataset sizes.

The aggregation function used when aggregating feature rankings from different studies seems not to have a particular effect on the testing error. However, when comparing the error curves of feature rankings produced by a single study and the aggregated ones, there is an obvious decrease in the error size. This is especially visible when combining bigger with smaller datasets, although sometimes a too small dataset might have detrimental effect on the aggregated ranking. This is very intuitive, and as a part of our further work we plan to take this into account when performing the aggregation by putting different weights of the base feature rankings related to dataset size and ranking quality.

Another important aspect of the error curve is its variability. One general pattern which can be noticed is that when aggregation of the feature rankings is performed (multiple ranking algorithms or multiple studies), the curve is much less variable than the curves of the base feature rankings. Although the variability does not directly represent feature ranking stability as described in Jurman et al. (2008) and Kalousis et al. (2007), we believe that it is indicative of it.

In our further work we plan to go beyond the visual inspection of the error curves. The first step would be to use the "area under the error curve" as a numerical way of assessing the quality of the curves. Also, we plan to include a correlation based indicator of stability

of the feature rankings, which combined with the area under the curve would provide an insight into the overall quality of the feature ranking.

Acknowledgments

This work was supported by the FP6, E.E.T.-Pipeline project, under the contract LifeSciHealth-2005-037260.

References

C. Ambroise and G. J. McLachlan. Selection bias in gene extraction on the basis of microarray gene-expression data. *Proc Natl Acad Sci U S A*, 99(10):6562–6566, May 2002.

L. Breiman. Random forests. *Machine Learning*, V45(1):5–32, October 2001.

K. De Preter, J. Vandesompele, P. Heimann, N. Yigit, S. Beckman, A. Schramm, A. Eggert, R. L. Stallings, Y. Benoit, M. Renard, A. De Paepe, G. Laureys, S. Påhlman, and F. Speleman. Correction: Human fetal neuroblast and neuroblastoma transcriptome analysis confirms neuroblast origin and highlights neuroblastoma candidate genes. *Genome Biology*, 8:401+, January 2007.

B. Efron and R. Tibshirani. Improvements on cross-validation: The .632+ bootstrap method. *Journal of the American Statistical Association*, 92(438):548–560, 1997.

I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1-3):389–422, 2002.

K. Jong, J. Mary, A. Cornuéjols, E. Marchiori, and M. Sebag. Ensemble feature ranking. In *Conf. on Principles and Practice of Knowledge Discovery in Databases (PKDD)*, Lecture Notes In Computer Science; Vol. 3202, pages 267–278, 2004.

G. Jurman, S. Merler, A. Barla, S. Paoli, A. Galea, and C. Furlanello. Algebraic stability indicators for ranked lists in molecular profiling. *Bioinformatics*, 24(2):258–264, January 2008.

A. Kalousis, J. Prados, and M. Hilario. Stability of feature selection algorithms: a study on high-dimensional spaces. *Knowledge and Information Systems*, 12(1):95–116, May 2007.

R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *IJCAI*, pages 1137–1145, 1995.

I. Kononenko. Estimating attributes: Analysis and extensions of relief, 1994.

Y. Saeys, T. Abeel, and Y. Van de Peer. Robust feature selection using ensemble feature selection techniques. In *ECML/PKDD (2)*, pages 313–325, 2008.

A. Schramm, J. H. Schulte, L. Klein-Hitpass, W. Havers, H. Sieverts, B. Berwanger, H. Christiansen, P. Warnat, B. Brors, J. Eils, R. Eils, and A. Eggert. Prediction of

clinical outcome and biological characterization of neuroblastoma by expression profiling. *Oncogene*, aop(current), 2005.

Q. Wang, S. Diskin, E. Rappaport, E. Attiyeh, Y. Mosse, D. Shue, E. Seiser, J. Jagannathan, S. Shusterman, M. Bansal, D. Khazi, C. Winter, E. Okawa, G. Grant, A. Cnaan, H. Zhao, N. Cheung, W. Gerald, W. London, K. K. Matthay, G. M. Brodeur, and J. M. Maris. Integrative Genomics Identifies Distinct Molecular Classes of Neuroblastoma and Shows That Multiple Genes Are Targeted by Regional Alterations in DNA Copy Number. *Cancer Res*, 66(12):6050–6062, 2006.