Digging into acceptor splice site prediction: an iterative feature selection approach

Yvan Saeys, Sven Degroeve, and Yves Van de Peer

Department of Plant Systems Biology, Ghent University, Flanders Interuniversity Institute for Biotechnology (VIB), K.L. Ledeganckstraat 35, Ghent, 9000, Belgium
{yvan.saeys,sven.degroeve,yves.vandepeer}@psb.ugent.be

Abstract. Feature selection techniques are often used to reduce data dimensionality, increase classification performance, and gain insight into the processes that generated the data. In this paper, we describe an iterative procedure of feature selection and feature construction steps, improving the classification of acceptor splice sites, an important subtask of gene prediction.

We show that acceptor prediction can benefit from feature selection, and describe how feature selection techniques can be used to gain new insights in the classification of acceptor sites. This is illustrated by the identification of a new, biologically motivated feature: the AG-scanning feature.

The results described in this paper contribute both to the domain of gene prediction, and to research in feature selection techniques, describing a new wrapper based feature weighting method that aids in knowledge discovery when dealing with complex datasets.

1 Introduction

During the past decades, feature selection techniques have increasingly gained importance, allowing researchers to cope with the massive amounts of data that have emerged in domains like image processing, text classification and bioinformatics. Within the framework of dimensionality reduction techniques, feature selection techniques are referred to as those techniques that select a (minimal) subset of features with "best" classification performance. In that respect, they differ from other reduction techniques like projection and compression, as they do not transform the original input features, but merely select a subset of them. The reduction of data dimensionality has a number of advantages: attaining good or even better classification performance with a restricted subset of features, faster and more cost-effective predictors, and the ability to get a better insight in the processes described by the data. An overview of feature selection techniques can be found in [11] and [7].

In many classification problems in bioinformatics, the biological processes that are modelled are far from being completely understood. Therefore, these classification models are mostly provided with a plethora of features, hoping that the truly important features are included. As a consequence, most of the features will be irrelevant to the classification task, and will act as noise. This can degrade the performance of a classifier, and obfuscates the interpretation by a human expert, motivating the use of feature selection techniques for knowledge discovery.

An important machine learning task in bioinformatics is the annotation of DNA sequences: given a genome sequence and a set of example gene structures, the goal is to predict all genes on the genome [18]. An important subtask within gene prediction is the correct identification of boundaries between coding regions (exons) and intervening, non coding regions (introns). These border sites are termed splice sites: the transition from exon to intron is termed the *donor* splice site, and the transition from intron to exon is termed the *acceptor* splice site. In higher organisms, the majority of the donor splice sites are characterized by a GT subsequence occurring in the intron part, while the acceptor splice sites have an AG subsequence in the intron part. As a result, each splice site prediction task (donor prediction, acceptor prediction) can be formally stated as a two-class classification task: given a GT or AG subsequence, predict whether it is a true donor/acceptor splice site or not. In this paper we will focus on the prediction of acceptor splice sites in the plant model species *Arabidopsis thaliana*.

The paper is organised as follows. We start by introducing the different classification algorithms and feature selection techniques that were used in the experiments. Then we discuss the application of these techniques to the acceptor prediction task, describing an iterative procedure of feature selection and feature construction steps. We conclude by summarizing the new ideas that emerged from this work, and some suggestions for future work.

2 Methods

It is generally known that there is no such thing as the best classification algorithm. Likewise, there is no best feature selection method, and a comparative evaluation of classification models and feature selection techniques should thus be conducted to find out which combination performs best on the dataset at hand. In this section we discuss the classifiers, feature selection algorithms, and combinations of both techniques that were evaluated in our experiments, followed by a description of the datasets and the implementation.

2.1 Classification algorithms

For our experiments, we chose two distinct types of classifiers that are widely used in machine learning research : a Bayesian classifier and a linear discriminant function. For the Bayesian classifier, the Naive Bayes method (NBM) was chosen because of its ability to cope with high-dimensional feature spaces and its robustness [5]. For the linear discriminant function, the linear Support Vector Machine (SVM) [3] was chosen because of its good performance on a wide range of classification tasks, and its ability to deal with high-dimensional feature spaces and large datasets.

2.2 Feature selection techniques

Techniques for feature selection are traditionally divided into two classes: filter approaches and wrapper approaches [14]. Filter approaches usually compute a feature relevance score such as the feature-class entropy, and remove low-scoring features. As such, these methods only look at the intrinsic properties of the dataset, providing a mechanism that is independent of the classification algorithm to be used afterwards. In the wrapper approach, various subsets of features are generated, and evaluated using a specific classification model. A heuristic search through the space of all subsets is then conducted, using the classification performance of the model as a guidance to find promising subsets. In addition to filter and wrapper approaches, a third class of feature selection methods can be distinguished: embedded feature selection techniques [2]. In embedded methods, the feature selection mechanism is built into the classification model, making direct use of the parameters of the induction model to include or reject features. In our experiments, a representative sample of each type of feature selection techniques was chosen, providing the basis for a comparative evaluation.

Filter techniques In general, filter techniques compute a feature relevance score in a single pass over the data, requiring a computational cost that is linear in the number of features. However, this comes at the expense of ignoring feature dependencies, which may result in suboptimal feature sets. To include feature dependencies, more complex filter approaches have been suggested such as the Markov blanket filter [15], or correlation based feature selection [8, 26].

In our work, we used the Markov blanket filter, introduced by Koller and Sahami, as an example of an advanced filter method that deals well with high-dimensional feature spaces [15]. This algorithm eliminates features whose information content is subsumed by some number of the remaining features. An approximate algorithm then solves the task of identifying the Markov blanket for each feature, and - starting from the full feature set - iteratively removes the feature with the "best" Markov blanket. The result of this algorithm is a ranking of all features from least relevant to most relevant. This ranking can then be used by any classification algorithm in a subsequent step. The parameter k of the algorithm determines the size of the Markov blanket, and exponentially increases running time as k gets larger. Typical values for K are $\{0,1,2\}$. In our experiments we choose $k = 1$.

Wrapper techniques Wrapper based methods combine a specific classification model with a strategy to search the space of all feature subsets. Commonly used methods are sequential forward or backward selection [13], and stochastic iterative sampling methods like genetic algorithms (GA) or estimation of distribution algorithms (EDA) [16, 10]. In our experiments we used an extension of the Univariate Marginal Distribution Algorithm (UMDA, [20]), the most simple EDA. This approach is very similar to the compact GA [9] or to a GA with uniform crossover. However, instead of using the single best feature subset that results from an iterative process like a GA or EDA, we used the frequencies with which the features are present in the final distribution of the UMDA as feature relevance scores. This gives a more dynamic view of the feature selection process, as it allows to derive feature weights. More details about this approach are discussed in [23].

Embedded techniques Two embedded feature selection techniques were used in the experiments: a weighted version of the Naive Bayes classifier, to which we will refer as WNBM, and recursive feature elimination using the weights of a linear SVM, further referred to as WLSVM.
The WNBM technique is based on the observation that the Naive Bayes method can be reformulated as a linear classifier when the classification task involves two classes, and all features are binary [5]. In this case, the feature weights can be calculated as

$$w_i = \ln \frac{p_i(1-p_i)}{q_i(1-q_i)} \ \forall\ i = 1, \cdots, n$$

where $p_i = Pr(x_i = 1|c_1)$ and $q_i = Pr(x_i = 1|c_2)$ are the class conditional probabilities of feature x_i being 1. These feature weights can then be sorted, providing a feature ranking. In principle, any classification task can be reduced to a set of two-class classification tasks, and arbitrary features can be converted to binary ones by discretization and sparse vector encoding.
In a similar fashion, the weights w_i of the decision boundary of a linear SVM can be used as feature weights to derive a feature ranking. However, better results can be obtained by recursively discarding one feature at the time, and retraining the SVM on the remaining features, as described in [6]. This is the approach that we adopted. A method, equivalent to this recursive feature elimination (RFE) approach, for feature selection with SVM is described in [24].

2.3 Data sets and implementation

The *Arabidopsis thaliana* data set was generated from sequences that were retrieved from the EMBL database, and only experimentally validated genes (i.e. no genes that resulted from a prediction) were used to build the dataset. Redundant genes were excluded, and splice site datasets were constructed from 1495 genes. More details on how these datasets were generated can be found in [4].

Because in real sequences, the number of true acceptor sites is largely outnumbered by the number of false acceptor sites, we chose to enforce a *class imbalance* in our datasets for feature selection. We constructed a dataset of 6000 positive instances and 36,000 negative instances. To obtain stable solutions for feature selection, a 10-fold cross-validation of this dataset was used to test all feature selection methods. This was done by doing 5 replications of a two-fold cross-validation, maintaining the same class imbalance of 1 positive versus 6 negative instances in every partition. For the EDA-based wrapper approach, the internal evaluation of classification performance was obtained by doing a 5-fold cross-validation on the training set.

The methods for feature selection were all implemented in C++, making use of the SVM^{light} implementation for Support Vector Machines [12]. The EDA-based wrapper method is a suitable candidate for parallellization, providing a linear speedup of the selection process. This was done using the MPI libraries, available at `http://www-unix.mcs.anl.gov/mpi/mpich`. However, due to other processes running on our servers, timing comparisons of the different algorithms fall outside the scope of this article.

3 An iterative feature selection approach

As already discussed in the introduction, many biological processes are still far from being understood. This greatly influences the design of the features that are to be used for a classification model that tries to model this process. In this section we describe an iterative feature construction and feature selection approach, resulting in increasingly more complex features and datasets, the design of which is guided by the feature selection techniques.

We start from the knowledge that the discrimination between true and false acceptor sites is determined by the part of the sequence where the site is located, more precisely the *local context* of the acceptor site. Therefore, the nucleotides A,T,C and G occurring on either side of the acceptor constitute a basic feature set.

3.1 A simple dataset: position dependent nucleotides

A local context of 100 nucleotides (50 to the left, 50 to the right) around the acceptor sites was chosen, having at each position one of the four nucleotides {A,T,C,G}. These features were extracted for the positive and negative instances, resulting in a dataset of 100 4-valued features, which were converted into binary format using sparse vector encoding (A=1000,T=0100,C=0010,G=0001). This results in a dataset described by 400 binary features. For this dataset, the following combinations of classifiers and feature selection algorithms were evaluated: the Koller-Sahami filter method (further abbreviated as KS) for both NBM and linear SVM (LSVM), the EDA-based wrapper approach (EDA-R) for both NBM and LSVM, and both the embedded methods WNBM and WLSVM. For EDA-R the distribution size and the number of iterations were tuned to 500 and 20 respectively. For the SVM, the C-parameter was tuned to 0.05, using the full feature set.

A comparative evaluation of the results of our experiments is shown in Table 1. The classifier/selection approach combinations are tabulated row wise, and the results on different feature subsets are shown in the columns. Apart from the results using the full feature set (100%), the results using only 50%, 25%, 10% and 5% of the features are shown. The numbers represent the average F-measure [19]

Table 1. *F test comparisons for the dataset of 400 features.*

Method	**100%**	**50%**	**25%**	**10%**	**5%**
KS NBM	80.87 ± 0.31	80.85 ± 0.37$^{=}$	78.77 ± 0.45	74.67 ± 0.79	72.14 ± 0.70
EDA-R NBM	80.87 ± 0.31	82.32 ± 0.32*	80.65 ± 0.37$^{=}$	76.70 ± 0.73	69.49 ± 2.46
WNBM	80.87 ± 0.31	80.80 ± 0.42$^{=}$	76.84 ± 0.41	67.52 ± 0.52	60.39 ± 1.73
KS LSVM	84.45 ± 0.30	82.75 ± 0.28	80.66 ± 0.49	75.05 ± 0.42	71.00 ± 0.49
EDA-R LSVM	84.45 ± 0.30	84.17 ± 0.38$^{=}$	81.62 ± 0.46	76.32 ± 0.67	68.73 ± 2.09
WLSVM	84.45 ± 0.30	84.00 ± 0.40	81.87 ± 0.39	76.73 ± 0.43	71.23 ± 0.40

over the 10 cross-validation folds, and the standard deviation. For each of the reduced feature sets, the result was compared to the results on the full feature set, using the combined 5x2 cv F test, introduced in [1]. Statistically significant improvements at confidence intervals of 0.9 and 0.99 were denoted respectively by ‡ and *, statistically equivalent results compared to the full feature set were denoted by $^{=}$ and statistically worse results were not marked.

On a global scale, the only method that achieves better results is NBM combined with EDA-R feature selection, using only half of the features. For NBM, the wrapper method thus seems to produce the best results. The filter method KS produces the second best results, and WNBM performs worst. For the linear SVM, no significant gain in classification performance could be obtained. This can be explained by the fact that the SVM already implicitly uses a feature weighting scheme. For the LSVM, the embedded method WLSVM achieves the best results overall, followed by EDA-R and KS.

For knowledge discovery, the only method in our experiments that is able to derive feature weights is the EDA-R method. Using the results of the EDA-R LSVM combination, we can thus use the derived weights to visualize the relevance of the features. This can be done by color coding the normalized feature weights, as is shown in Figure 1. Part a in this figure shows the results of this color coding, where a gradient ranging from blue (unimportant) to red (important) shows the feature weights. For each of the nucleotides (rows), the nucleotide positions (columns) are shown for both parts of the local context, the acceptor site being in the middle. Several patterns can be observed. The nucleotides bordering the acceptor site are of primary importance, representing binding information. Furthermore the nucleotides T in the left part of the context are highly important, representing the well-known pyrimidine-stretch (an excess of nucleotides T and C). A last pattern that can be observed is the three-base periodicity in the right part of the context, especially for nucleotides T and G, capturing the fact that this part of the sequence is the coding part (exon), and that nucleotides in this part are organized in triplets.

3.2 Adding position invariant features

Position dependent features are not the best solutions when trying to capture information like coding potential and composition in the sequence. Therefore, a

Table 2. *F test comparisons for the dataset of 528 features.*

Method	**100%**	**50%**	**25%**	**10%**	**5%**
KS NBM	78.21 ± 0.50	$78.40 \pm 0.50^{=}$	$77.96 \pm 0.64^{=}$	77.26 ± 0.46	74.21 ± 0.58
EDA-R NBM	78.21 ± 0.50	$84.48 \pm 0.30^{*}$	$83.52 \pm 0.36^{*}$	$80.79 \pm 0.57^{\ddagger}$	75.93 ± 0.87
WNBM	78.21 ± 0.50	77.17 ± 0.51	$77.85 \pm 0.37^{=}$	74.06 ± 0.32	67.96 ± 0.68
KS LSVM	87.52 ± 0.49	87.15 ± 0.32	86.03 ± 0.41	82.05 ± 0.46	77.03 ± 0.80
EDA-R LSVM	87.52 ± 0.49	86.72 ± 0.54	85.64 ± 0.59	82.34 ± 0.43	77.02 ± 1.10
WLSVM	87.52 ± 0.49	87.20 ± 0.49	86.40 ± 0.50	84.35 ± 0.48	78.34 ± 0.92

second dataset was constructed as an extension of the dataset in the previous experiment. In addition to the position dependent nucleotide features, we also added a number of position invariant features. These features capture the occurrence of 3-mers (words of length 3) in the sequence flanking the acceptor. An example of such a feature is the occurrence of the word “GAG” in the left part of the context. This results in another 128 binary features (64 for each part of the context), a 1 decoding the presence, a 0 the absence of the specific word in the context. Together with the position dependent features, this yields a dataset consisting of 528 binary features. The same parameters for EDA-R and SVM were used as with the previous dataset.

The results of the feature selection experiments on this dataset are shown in Table 2. Comparing the results for NBM to the previous dataset, the classification performance on the full feature set is lower than on the first dataset. However, using feature selection, better classification results than on the first dataset can be obtained. Again, the best results were obtained with the EDA-R wrapper method. Using only 10% of the features, this method still obtains significantly better results than using the full feature set. The KS filter method performs second best, WNBM performs worst. For the SVM, a significant gain of 3% in classification performance was obtained by adding the position invariant features. Similar to the previous dataset, the performance could not be improved using feature selection methods, and the embedded method WLSVM obtains the best results.

The visualization of the feature weights, obtained by the EDA-R LSVM approach, is shown in part b of Figure 1. While the same patterns as in the case of dataset 1 can be observed, it is clear that some information is translated from position dependent features into position invariant features. An example of this is the pyrimidine stretch, which is somewhat shortened compared to the results on the previous dataset, together with the fact that T-rich 3-mers in the left part of the context show up as very important. Another example is the fact that the 3-base periodicity on the coding side is less pronounced, yet some 3-mers are shown to be highly relevant. The results from the feature weighting, combined with the improved classification results explain that indeed position invariant features contribute to the prediction of acceptor sites.

Table 3. *F test comparisons for the dataset of 2096 features.*

Method	**100%**	**50%**	**25%**	**10%**	**5%**
KS NBM	79.21 ± 0.33	79.46 ± 0.30$^{=}$	79.08 ± 0.39$^{=}$	79.07 ± 0.57$^{=}$	78.03 ± 0.97$^{=}$
EDA-R NBM	79.21 ± 0.33	85.29 ± 0.36*	83.81 ± 0.69*	79.90 ± 0.62$^{=}$	76.51 ± 0.99
WNBM	79.21 ± 0.33	79.90 ± 0.44‡	79.52 ± 0.34$^{=}$	77.36 ± 0.50	75.61 ± 0.58
KS LSVM	88.24 ± 0.51	87.56 ± 0.41$^{=}$	85.62 ± 0.64	83.10 ± 0.49	79.88 ± 1.06
EDA-R LSVM	88.24 ± 0.51	87.90 ± 0.36	86.66 ± 0.43	84.07 ± 0.74	81.73 ± 0.56
WLSVM	88.24 ± 0.51	88.22 ± 0.44$^{=}$	88.08 ± 0.35$^{=}$	87.10 ± 0.32	85.86 ±0.34

3.3 Adding more complex features: dependencies between adjacent nucleotides

It is known that correlations exist between nucleotides in the vicinity of splice sites. To detect these dependencies, higher-order (i.e. non-linear) classification methods can be used, like polynomial SVMs. However, these methods have the disadvantage of being quite slow to train, rendering the feature selection process more computationally intensive. Here, we describe another approach to deal with nucleotide dependencies, having the advantage that linear (and thus fast) classification models can be used. We do this by constructing more complex features, capturing the nucleotide dependencies at the feature level. Another important advantage is that the combination with feature selection techniques allows us to select those dependencies that are of primary importance, and visualize them.

To this end, we created complex features that capture dependencies between two adjacent nucleotides. These features are represented as position dependent 2-mers (words of length 2). At each position i of the local context, these features represent the word appearing at position i and $i + 1$. This results in an a set of 1568 binary features (49x16x2). Together with the position dependent nucleotides and the position invariant features, this results in a dataset described by 2096 features. For this dataset, the C-parameter of the SVM was tuned to 0.005.

The results of the feature selection experiments for this dataset are shown in part c of Figure 1. Compared to the results on the previous datasets, similar trends can be observed. The NBM classifier performs worse than dataset 1 on the full feature set, but outperforms the results on dataset 1 and 2, when EDA-R is used with only 50% of the features. For the SVM, an increase in classification performance is noted, compared to dataset 1 and 2. Again, the result on the full feature set cannot be improved using feature selection methods.

Visualizing the weights derived from the EDA-R LSVM combination (Figure 1, part c) reveals some remarkable, new patterns. In addition to the previous patterns, three new patterns, related to the inclusion of dependencies between adjacent nucleotides, can be observed. Firstly, it is observed that nucleotide dependencies immediately neighbouring the acceptor site are of great importance. Furthermore two patterns, related to the 2-mers AG and TG emerge in the left part of the context.

It should be noted that the only result that can be drawn from this visualization is the fact that these are important features for classification. The method does not tell if e.g. AG occurs more or less at these positions in true acceptor sites than in false sites. In order to reveal this information, inspection of the classification model or the datasets is needed. In the case of the AG-feature, an analysis of our dataset shows that there is a strong selection against AG dinucleotides in the left part of the context. This can be explained by looking into more detail to the biological process involved in the recognition of acceptor sites. In this process, a protein binds to a part of the sequence to the left of the acceptor (the so-called branch point) and then scans the sequence until an AG is encountered (usually the first AG encountered is the splice site). As a result, our feature selection method discovers this "AG-scanning" as very important in the distinction between true and false acceptors, as false acceptors will usually have more AG dinucleotides in the left part of the sequence. The second pattern (TG) was identified as being more abundant in true acceptor sites than in false sites, and is probably related to the T-abundance of the pyrimidine stretch.

Comparing the results of all feature selection combinations on the three dataset reveals some general trends for these acceptor datasets. For the NBM classifier, classification performance could be significantly improved using feature selection, especially using the EDA-R wrapper method, which achieves the best results when half of the features have been eliminated. Using feature selection on the most complex dataset achieves an F-measure of 85%, which is about 5% better than using all features of the simplest dataset. Overall, the EDA-R method gives the best results for NBM, followed by the KS filter method, and the embedded method WNBM.

For the linear SVM, classification performance could not be improved upon using feature selection. At least equivalent results could be obtained using feature selection methods. However, we showed the potential of feature selection techniques as a useful tool to gain more insight into the underlying biological process, using the combination of EDA-R with SVM. The iterative approach of feature selection and feature construction shows that also the classification performance of SVM could be improved. On the most complex dataset, SVM achieves an F-measure of about 88%, an increase by 3% compared to using the most simple dataset. Additionally, the complex dataset allowed us to distinguish a new feature: AG-scanning.

4 Related work

The use of feature selection techniques for splice site prediction was first described in [4]. More recent work on splice sites includes work on maximum entropy modelling [25], where the authors only consider a very short local context, and the work of Zhang et al. [27] where SVMs are used to model human splice sites. Related work on using EDAs for feature selection can be found in [10, 22], and some recent developments to use feature selection in combination with SVMs are described in [24]

5 Conclusions and future work

In this paper, we described an iterative feature selection approach for the classification of acceptor splice sites, an important problem in the context of gene prediction. A comparative evaluation of various feature selection and classification algorithms was performed, demonstrating the usefulness of feature selection techniques for this problem. Furthermore, we proposed a new feature weighting scheme (derived from the EDA-R method) that deals well with datasets described by a large number of features. We showed how this technique can be used to guide the feature construction process, arriving at more complex feature sets with better classification performance. Using these feature weights, and the design of complex features that capture dependencies at the feature level, we demonstrated how important nucleotide correlations could be visualised. Such a visualisation facilitates the discovery of knowledge for human experts, and led to the discovery of a new, biologically motivated feature: the AG-scanning feature.

For the best-scoring method of our experiments, the linear SVM, preliminary experiments with a more complex version of the AG-scanning feature have been designed, incorporating the predictor in a complete gene prediction system. These experiments show promising results with respect to state-of-the-art splice site predictors, like GeneSplicer [21]. Another line of future research is motivated by the use of linear classifiers in combination with more complex features, capturing higher order, or non-adjacent nucleotide dependencies at the feature level, or taking into account secondary structure information. Combining such complex feature sets with linear classifiers and feature selection techniques can be useful to learn and visualise more complex dependencies.

References

1. Alpaydin, E. A Combined 5x2 cv F Test for Comparing Supervised Classification Learning Algorithms. Neural Computation **11(8)** (1999) 1885–1892
2. Blum, A.I., Langley, P.: Selection of relevant features and examples in machine learning. Artificial Intelligence **97** (1997) 245–271
3. Boser, B., Guyon, I., Vapnik, V.N.: A training algorithm for optimal margin classifiers. Proceedings of COLT (Haussler,D. ,ed.), ACN Press (1992) 144–152
4. Degroeve, S., De Baets, B., Van de Peer, Y., Rouzé, P.: Feature subset selection for splice site prediction Bioinformatics **18 Supp.2** (2002) 75–83
5. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern classification. New York, NY, Wiley, 2nd edition (2000)
6. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.N.: Gene Selection for Cancer Classification using Support Vector Machines. Machine Learning **46(1-3)** (2000) 389–422
7. Guyon, I., Elisseeff, A.: An Introduction to Variable and Feature Selection. Journal of Machine Learning Research **3** (2003) 1157–182

8. Hall, M.A., Smith, L.A.: Feature Selection for Machine Learning : Comparing a Correlation-based Filter Approach to the Wrapper. Proc. of the Florida Artificial Intelligence Symposium (1999)
9. Harik, G.R., Lobo, G.G., Goldberg, D.E.: The compact genetic algorithm. Proc. of the International Conference on Evolutionary Computation 1998 (ECEC '98), Piscataway, NJ: IEEE Service Center (1998) 523–528
10. Inza, I.,Larrañaga, P., Sierra. B. Feature Subset Selection by Estimation of Distribution Algorithms. In *Estimation of Distribution Algorithms. A new tool for Evolutionary Computation.* (2001) P. Larrañaga, J.A. Lozano (eds.)
11. Jain, A.K., Duin, R.P.W., Mao, J. Statistical Pattern Recognition: A Review. IEEE Transactions on Pattern Analysis and Machine Intelligence **22(1)** (2000) 4–37
12. Joachims, T.: Making large-scale support vector machine learning practical. B. Schölkopf, C. Burges, A. Smola. Advances in Kernel Methods: Support Vector Machines, MIT Press, Cambridge, MA (1998)
13. Kittler, J.: Feature set search algorithms. In *Pattern Recognition and Signal Processing* (1978) 41–60
14. Kohavi, R., John, G.: Wrappers for feature subset selection. Artificial Intelligence **97(1-2)** (1997) 273–324
15. Koller, D., Sahami, M.: Toward optimal feature selection. Proceedings Thirteenth International Conference on Machine Learning (1996) 284-292
16. Kudo, M., Sklansky, J.: Comparison of algorithms that select features for pattern classifiers. Pattern Recognition **33** (2000) 25–41
17. Larrañaga, P., Lozano, J.A.: Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation. Kluwer Academic Publishers (2001)
18. Mathé, C., Sagot, M.F., Schiex, T., Rouzé, P.: Current methods of gene prediction, their strengths and weaknesses. Nucleic Acids Research **30** (2002) 4103–4117
19. Mladenić, D., Grobelnik, M.: Feature selection on hierarchy of web documents. Decision Support Systems **35** (2003) 45–87
20. Mühlenbein, H., Paass, G.: From recombination of genes to the estimation of distributions. Binary parameters. Lecture Notes in Computer Science 1411 : Parallel Problem Solving from Nature, PPSN IV (1996) 178–187
21. Pertea, M., Lin, X., Salzberg, S.: GeneSplicer: a new computational method for splice site prediction. Nucleic Acids Research **29** (2001) 1185–1190
22. Saeys, Y., Degroeve, S., Aeyels, D., Van de Peer, Y., Rouzé, P.: Fast feature selection using a simple Estimation of Distribution Algorithm: A case study on splice site prediction. Bioinformatics **19-2** (2003) 179–188
23. Saeys, Y., Degroeve, S., Van de Peer, Y.: Feature ranking using an EDA-based wrapper approach. In *Towards a new evolutionary computation: advances in Estimation of Distribution Algorithms* J.A. Lozano et al. (eds), In press
24. Weston, J., Elisseeff, A., Schoelkopf, B., Tipping, M.: Use of the Zero-Norm with Linear Models and Kernel Methods. Journal of Machine Learning Research **3** (2003) 1439–1461
25. Yeo, G., Burge, C.B.: Maximum entropy modelling of short sequence motifs with applications to RNA splicing signals. Proceedings of RECOMB 2003 (2003) 322–331
26. Yu, L., Liu,H.: Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution. Proceedings ICML 2003 (2003) 856–863
27. Zhang, X., Heller, K., Hefter, I., Leslie, C., Chasin L.: Sequence Information for the Splicing of Human pre-mRNA Identified by Support Vector Machine Classification Genome Research **13** (2003) 2637–2650

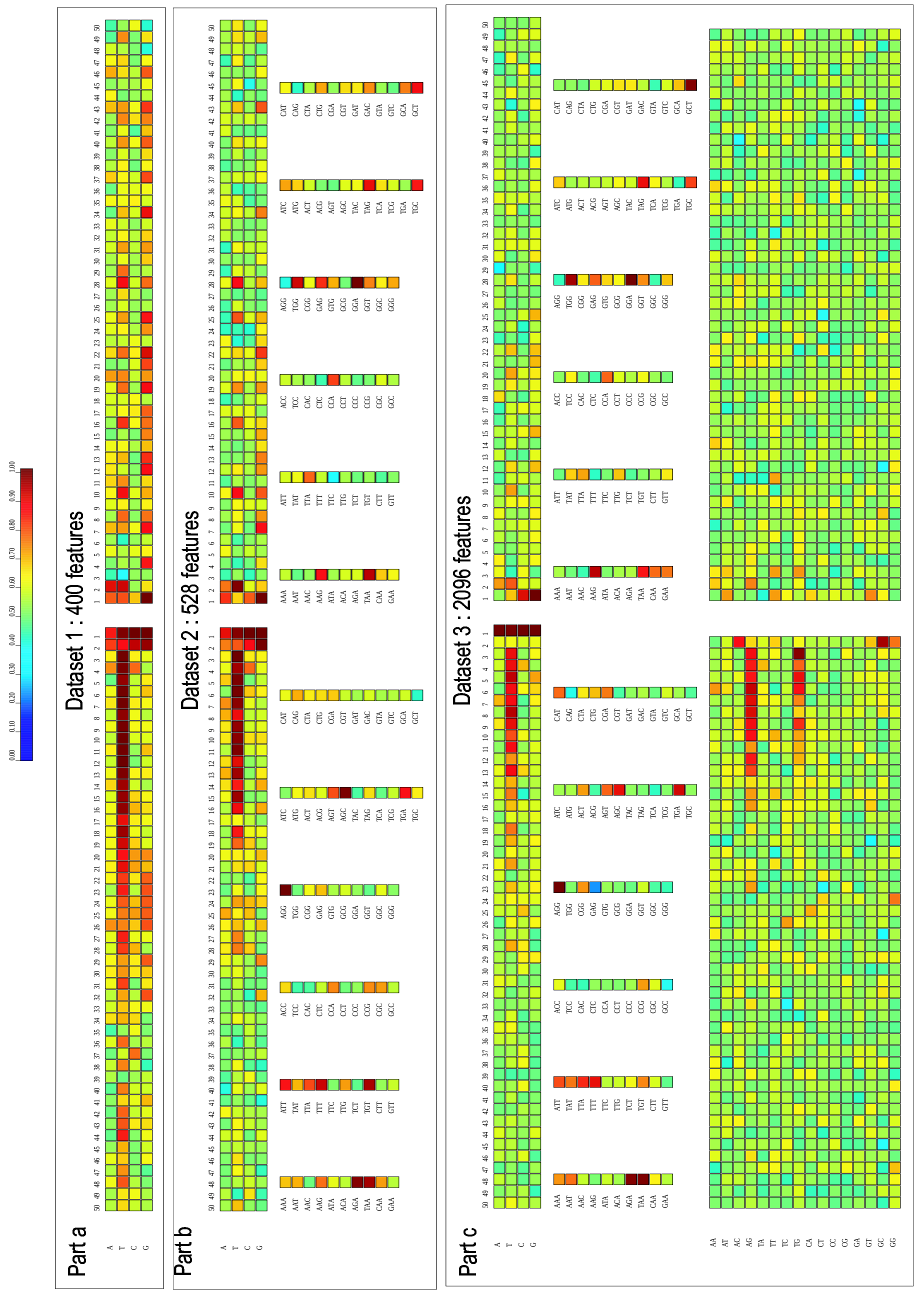

Fig. 1. *Visualization of the feature weights for datasets with increasing complexity, obtained by the EDA-R LSVM approach: a simple dataset of 400 binary features (part a), an extended dataset also including position invariant features (part b), and a complex dataset also capturing adjacent feature dependencies (part c).*