Evolutionary Optimization of Sequence Kernels for Detection of Bacterial Gene Starts

Britta Mersch[1], Tobias Glasmachers[2], Peter Meinicke[3], and Christian Igel[2]

[1] German Cancer Research Center, 69120 Heidelberg, Germany
b.mersch@dkfz.de
[2] Institut für Neuroinformatik, Ruhr-Universität Bochum, 44780 Bochum, Germany
{Tobias.Glasmachers, Christian.Igel}@neuroinformatik.rub.de
[3] Institut für Mikrobiologie und Genetik, Abteilung für Bioinformatik,
Georg-August-Universität Göttingen, 37077 Göttingen, Germany
pmeinic@gwdg.de

Abstract. Oligo kernels for biological sequence classification have a high discriminative power. A new parameterization for the K-mer oligo kernel is presented, where all oligomers of length K are weighted individually. The task specific choice of these parameters increases the classification performance and reveals information about discriminative features. For adapting the multiple kernel parameters based on cross-validation the covariance matrix adaptation evolution strategy is proposed. It is applied to optimize the trimer oligo kernel for the detection of prokaryotic translation initiation sites. The resulting kernel leads to higher classification rates, and the adapted parameters reveal the importance for classification of particular triplets, for example of those occurring in the Shine-Dalgarno sequence.

1 Introduction

Kernel-based learning algorithms have been successfully applied to a variety of sequence classification tasks within the field of bioinformatics [1]. Recently, *oligo kernels* were proposed [2] for the analysis of biological sequences. Here the term oligo (-mer) refers to short, single stranded DNA/RNA fragments. Oligo kernels compare sequences by looking for matching fragments. They allow for gradually controlling the level of position-dependency of the representation, that is, how important the exact position of an oligomer is. In addition, decision functions based on oligo kernels are easy to interpret and to visualize and can therefore be used to infer characteristic sequence features.

In the standard oligo kernel, all oligomers are weighted equally. Thus, all oligomers are considered to have the same importance for classification. In general this assumption is not reasonable. In this study, we therefore propose the K-weighted oligo kernel considering all oligomers of length K (K-mers), in which the relative importance of all K-mers can be controlled individually. A task specific choice of the weighting parameters can potentially increase the classification performance. Moreover, appropriate weights for a particular classification task may reveal sequence characteristics with high discriminative power and biological importance.

The question arises how to adjust the weighting parameters for the K-mers for a given task. In practice, appropriate hyperparameter combinations are usually determined by

S. Kollias et al. (Eds.): ICANN 2006, Part II, LNCS 4132, pp. 827–836, 2006.

grid search. This means that the hyperparameters are varied with a fixed step size through a wide range of values and the performance of every combination is assessed using some performance measure. Because of the computational complexity, grid search is only suitable for the adjustment of very few parameters. Hence, it is not applicable for the adjustment of the 4^K weights of the K-weighted oligo kernel. Perhaps the most elaborated systematic technique for choosing multiple hyperparameters are gradient descent methods [3, 4, 5]. If applicable, these methods are highly efficient. However, they have significant drawbacks. In particular, the score function for assessing the performance of the hyperparameters (or at least an accurate approximation of this function) has to be differentiable with respect to all hyperparameters. This excludes reasonable measures such as the (exact) cross-validation error. Further, the considered space of kernels has to have an appropriate differentiable structure.

We propose a method for hyperparameter selection that does not suffer from the limitations described above, namely using the covariance matrix adaptation evolution strategy (CMA-ES, [6]) to search for appropriate hyperparameter vectors [7, 8].

As an application of our approach to kernel optimization we consider the prediction of bacterial gene starts in genomic sequences. Although exact localization of gene starts is crucial for correct annotation of bacterial genomes, it is difficult to achieve with conventional gene finders, which are usually restricted to the identification of long coding regions. The prediction of gene starts therefore provides a biologically relevant signal detection task, well-suited for the evaluation of our kernel optimization scheme.

We therefore apply the CMA-ES to the tuning of weighted oligo kernels for detecting prokaryotic translation initiation sites, that is, for classifying putative gene starts in bacterial RNA. The performance measure for the hyperparameter optimization is based on the mean classification rate of five-fold cross-validation.

In the following we introduce the oligo kernel and our new parameterization. Section 3 deals with the adaptation of kernel parameters using evolutionary optimization methods. Section 4 presents the experiments demonstrating the performance of the kernel and the optimization of the hyperparameters.

2 Oligo Kernels

The basic idea of kernel methods for classification is to map the input data, here biological sequences, to a feature space endowed with a dot product. Then the data is processed using a learning algorithm in which all operations in feature space can be expressed by dot products. The trick is to compute these inner products efficiently in input space using a kernel function (e.g., see [9]). Here the feature space can be described in terms of *oligo functions* [2]. These functions encode occurrences of oligomers in sequences with an adjustable degree of positional uncertainty. This is in contrast to existing methods, which provide either position-dependent [10] or completely position-independent representations [11]. For an alphabet $\mathscr{A}$ and a sequence $\mathbf{s}$, which contains K-mer $\omega \in \mathscr{A}^K$ at positions $S_\omega^\mathbf{s} = \{p_1, p_2, \dots\}$, the oligo function is given by

$$\mu_\omega^\mathbf{s}(t) = \sum_{p \in S_\omega^\mathbf{s}} \exp\left(-\frac{1}{2\sigma_K^2}(t-p)^2\right)$$

for $t \in \mathbb{R}$. The smoothing parameter σ_K adjusts the width of the Gaussians centered on the observed oligomer positions and determines the degree of position-dependency of the function-based feature space representation. While small values for σ_K imply peaky functions, large values imply flatter functions. For a sequence $\mathbf{s}$ the occurrences of all K-mers contained in $\mathscr{A}^K = \{\omega_1, \omega_2, \ldots, \omega_m\}$ can be represented by a vector of m oligo functions. This yields the final feature space representation $\Phi^K(\mathbf{s}) = [\mu^{\mathbf{s}}_{\omega_1}, \mu^{\mathbf{s}}_{\omega_2}, \ldots, \mu^{\mathbf{s}}_{\omega_m}]'$ of that sequence. The feature space objects are vector-valued functions. This can be stressed using the notation

$$\phi^K_{\mathbf{s}}(t) = [\mu^{\mathbf{s}}_{\omega_1}(t), \mu^{\mathbf{s}}_{\omega_2}(t), \ldots, \mu^{\mathbf{s}}_{\omega_m}(t)]' \ .$$

This representation is well-suited for the interpretation of discriminant functions and visualization [2]. To make it practical for learning, we construct a kernel function to compute the dot product in the feature space efficiently. The inner product of two sequence representations ϕ^K_i and ϕ^K_j, corresponding to the oligo kernel $k_K(\mathbf{s}_i, \mathbf{s}_j)$, can be defined as

$$\langle \phi^K_i, \phi^K_j \rangle \equiv \int \phi^K_i(t) \cdot \phi^K_j(t) \mathrm{d}t \propto \sum_{\omega \in \mathscr{A}^K} \sum_{p \in S^i_\omega} \sum_{q \in S^j_\omega} \exp\left(-\frac{1}{4\sigma^2_K}(p-q)^2\right) \equiv k_K(\mathbf{s}_i, \mathbf{s}_j)$$

using $\phi_i \equiv \phi_{\mathbf{s}_i}$. The feature space representations of two sequences may have different norms. In order to improve comparability between sequences of different lengths, we compute the normalized oligo kernel

$$\tilde{k}_K(\mathbf{s}_i, \mathbf{s}_j) = \frac{k_K(\mathbf{s}_i, \mathbf{s}_j)}{\sqrt{k_K(\mathbf{s}_i, \mathbf{s}_i) k_K(\mathbf{s}_j, \mathbf{s}_j)}} \ . \tag{1}$$

From the above definition of the oligo kernel, the effect of the smoothing parameter σ_K becomes obvious. For the limiting case $\sigma_K \to 0$ with no positional uncertainty, only oligomers which occur at the same positions in both sequences contribute to the sum. In general it is not appropriate to represent oligomer occurrences without positional uncertainty. This would imply zero similarity between two sequences if no K-mer appears at *exactly* the same position in both sequences. For $\sigma_K \to \infty$ position-dependency of the kernel completely vanishes. In this case, all terms of oligomers occurring in both sequences contribute equally to the sum, regardless of their distance and the oligo kernel becomes identical to the spectrum kernel [11].

2.1 Weighted Oligo Kernel

So far, the different K-mers are weighted equally in the K-mer oligo kernel. However, some K-mers may be more discriminative than others. Therefore, we introduce new parameters w_i, $i = 1, \ldots, 4^K$, for their weighting and define the K-weighted oligo kernel $\tilde{k}_{K\text{-weighted}}$ in analogy to equation (1) with

$$k_{K\text{-weighted}}(\mathbf{s}_i, \mathbf{s}_j) = \sum_{\omega \in \mathcal{A}^K} |w_i| \sum_{p \in S^{\mathbf{s}_i}_\omega} \sum_{q \in S^{\mathbf{s}_j}_\omega} \exp\left(-\frac{1}{4\sigma^2_K}(p-q)^2\right) \ .$$

The parameterization ensures a valid oligo kernel for $w_1, \ldots, w_{4^K}, \sigma \in \mathbb{R}$. This makes unconstrained optimization methods directly applicable to the $1 + 4^K$ kernel parameters.

3 Evolutionary Model Selection

Evolutionary algorithms are iterative, direct, randomized optimization methods inspired by principles of neo-Darwinian evolution theory. They have proven to be suitable for hyperparameter and feature selection for kernel-based learning algorithms [12,7,13,14, 8,15,16,17,18,19]. Evolution strategies (ES, [20]) are one of the main branches of evolutionary algorithms. Here the highly efficient covariance matrix ES (CMA-ES, [21,6]) for real-valued optimization is applied, which learns and employs a variable metric by means of a covariance matrix for the search distribution. The CMA-ES has successfully been applied to tune Gaussian kernels for SVMs considering a cross-validation error as optimization criterion [7, 8]. The visualization of the objective function in [7] depicts an error surface that shows a global trend superimposed by local minima, and ES are usually a good choice for such kind of problems.

In the CMA-ES, a set of μ individuals forming the parent population is maintained. Each individual has a genotype that encodes a candidate solution for the optimization problem at hand, here a real-valued vector containing the hyperparameter combination of the kernel parameters to be optimized. The fitness of an individual is equal to the objective function value—here the five-fold cross-validation error—at the point in the search space it represents. In each iteration of the algorithm, $\lambda > \mu$ new individuals, the offspring, are generated by partially stochastic variations of parent individuals. The fitness of the offspring is computed and the μ best of the offspring form the next parent population. This loop of variation and selection is repeated until a termination criterion is met. The object variables are altered by global intermediate recombination and Gaussian mutation. That is, the genotypes $\mathbf{g}_k^{(t)}$ of the offspring $k = 1, \ldots, \mu$ created in iteration t are given by $\mathbf{g}_k^{(t)} = \langle \tilde{\mathbf{g}} \rangle^{(t)} + \xi_k^{(t)}$, where $\langle \tilde{\mathbf{g}} \rangle^{(t)}$ is the center of mass of the parent population in iteration t, and the $\xi_k^{(t)} \sim \mathcal{N}(0, \mathbf{C}^{(t)})$ are independent realizations of an m-dimensional normally distributed random vector with zero mean and covariance matrix $\mathbf{C}^{(t)}$. The matrix $\mathbf{C}^{(t)}$ is updated online using the covariance matrix adaptation method (CMA). Roughly speaking, the key idea of the CMA is to alter the mutation distribution in a deterministic way such that the probability to reproduce steps in the search space that led to the actual population—i.e., produced offspring that were selected—is increased. The search path of the population over the past generations is taken into account, where the influence of previous steps decays exponentially. The CMA does not only adjust the mutation strengths in m directions, but also detects correlations between object variables. The CMA-ES is invariant against order-preserving transformations of the fitness function and in particular against rotation and translation of the search space—apart from the initialization. If either the strategy parameters are initialized accordingly or the time needed to adapt the strategy parameters is neglected, any affine transformation of the search space does not affect the performance of the CMA-ES. For details of the CMA-ES algorithm, we refer to the literature [21,6].

4 Detection of Prokaryotic Translation Initiation Sites

We apply 1-norm soft margin SVMs with 3-mer weighted oligo kernels to the detection of prokaryotic translation initiation sites [22]. We first introduce the problem and then the locality improved kernel, which we consider for comparison. Then the experimental setup is described. Finally the results are presented.

4.1 Problem Description

To extract protein-encoding sequences from nucleotide sequences is an important task in bioinformatics. For this purpose it is necessary to detect locations at which coding regions start. These locations are called translation initiation sites (TIS). A TIS contains the start codon ATG or rarely GTG or TTG (there is one known case where also ATT serves as a start codon). The start codon marks the position at which the translation starts. The codon ATG codes for the amino acid methionine, and not every ATG triplet is a start codon. Therefore it must be decided whether a particular ATG corresponds to a start codon or not. This classification problem can be solved automatically using machine learning techniques, in which the neighborhood of nucleotides observed around potential TISs is used as input pattern to a classifier.

In contrast to prediction of eukaryotic TIS (e.g., see [23]) there is no biological justification for using a general learning machine across different species for prediction of prokaryotic TIS. For this reason, learning of prokaryotic TISs is always restricted to a limited amount of species-specific examples and model selection methods have to cope with small data sets.

As in previous studies, we tested our approach on *E. coli* genes from the EcoGene database [24]. Only those entries with biochemically verified N-terminus were considered and the neighboring nucleotides were looked up in the GenBank file U00096.gbk [25]. From the 730 positive examples we created associated negative examples. For the negative examples we extracted sequences centered around a codon from the set $\{\mathsf{ATG}, \mathsf{GTG}, \mathsf{TTG}\}$. Such a sequence is used as a negative example if the codon is in-frame with one of the correct start sites used as a positive case, its distance from a real TIS is less than 80 nucleotides, and no in-frame stop codon occurs in between. This procedure generates a difficult benchmark data set, because the potential TISs in the neighborhood of the real start codon are the most difficult candidates in TIS discrimination. We created 1243 negative examples. The length of each sequence is 50 nucleotides, with 32 located upstream and 15 downstream with respect to the potential start codon.

To minimize random effects, we generated 40 different partitionings of the data into training and test sets. Each training set contained 400 sequences plus the associated negatives, the corresponding test set 330 sequences plus the associated negatives.

Measuring the performance of a TIS classifier by the standard classification rate on test sets leads to over-optimistic results. In a process of annotation, one normally obtains a window with several possible TISs. The goal is to detect the position of a real TIS—if there is one—within this window. If there are several positions marked as TISs, one has to select one of them. In practice, the position with the highest score (i.e., decision function value) is chosen. Thus, although a real TISs was classified as a TIS, the classification can be overruled by a wrong classification in the neighborhood. Therefore, when the SVM categorizes a location with corresponding sequence $\mathbf{s}$ as being a TIS, we consider a frame of 160 nucleotides centered at that position. The score of every potential TIS within this frame is computed. Only if $\mathbf{s}$ corresponds to a real TIS and

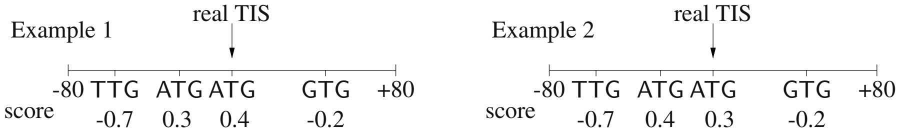

Fig. 1. Performance assessment: Example 1 shows a correct positive classification of a TIS. In example 2, the classification is not correct: The real TIS is classified as a TIS, but its score is not the largest in the neighborhood.

the score for **s** is the largest of all potential TIS locations, the pattern **s** is considered to be classified correctly, see Figure 1.

4.2 Locality Improved Kernel

For comparison, we consider the locality improved kernel [1,23]. It counts matching nucleotides and considers local correlations within local windows of length $2l+1$. Given two sequences $\mathbf{s}_i$, $\mathbf{s}_j$ of length L the locality improved kernel is given by

$$k_{\text{locality}}(\mathbf{s}_i, \mathbf{s}_j) = \sum_{p=l+1}^{L-l} \left(\sum_{t=-l}^{+l} v_{t+l} \cdot \text{match}_{p+t}(\mathbf{s}_i, \mathbf{s}_j) \right)^d$$

with $\text{match}_t(\mathbf{s}_i, \mathbf{s}_j)$ equal to one if $\mathbf{s}_i$ and $\mathbf{s}_j$ have the same nucleotide at position t and zero otherwise. The weights v_t allow to emphasize regions of the window which are of special importance. They were fixed to $v_t = 0.5 - 0.4|l-t|/l$ [1]. The hyperparameter d determines the order to which local correlations are considered.

4.3 Experiments

In our experiments, we considered trimer oligo kernels with hyperparameter σ, locality improved kernels with hyperparameters l and d, and weighted trimer oligo kernels with adjustable σ and 64 weights. For each of the 40 partitionings into training and test data and each sequence kernel independent optimizations of the kernel parameters were conducted. In the end, we evaluate the median of the 40 trials.

For the SVM using the oligo kernel without individually weighting of the K-mers we adjusted the smoothing parameter σ by one-dimensional grid-search. After narrowing the possible values down, the grid search varied $\sigma \in \{0.1 + 0.2 \cdot k \mid 0 \leq k < 10\}$. The parameters l and d of the locality improved kernel were also optimized using two-dimensional grid-search. After determining an interval of parameters leading to well generalizing classifiers, the grid-search varied $l, d \in \{2, 3, 4\}$ [23]. For both kernels, independent grid-searches were performed for each of the 40 partitionings.

The $1 + 4^3 = 65$ parameters of the weighted trimer oligo kernels were optimized using the CMA-ES with randomly chosen starting points in the interval $[0, 1]$. For each of the 40 partitionings an independent optimization trial was started. The offspring population size was $\lambda = 16$ (e.g., a default choice for this dimensionality, see [21]) and each trial lasted 100 generations.

The optimization criterion in the grid-searches and the evolutionary optimization was the five-fold cross-validation error based on the error measure described above. The training data set is partitioned into five disjoint subsets. For each of the subsets, the classifier is trained using the union of the four other sets and a test error is computed on the left-out subset. The final cross-validation error is the average of the five test errors.

4.4 Results

We first interpret the outcome of the optimization of the parameters of the weighted oligo kernel. Then we compare the classification performance of the weighted oligo kernel, the trimer oligo kernel with equal weights, and the locality improved kernel.

The results of the optimization of the smoothing parameter σ are shown in Table 1. The optimized values are rather small, that is, the position of the triplets is very important. However, the smoothing parameter for the oligo and the weighted oligo kernel do not differ much.

To analyze the relevance of particular oligomers, the 64 triplets were sorted according to the median of the corresponding evolved weighting parameters. The weight values indeed vary, and a group of a few oligomers with comparatively high weight values can be identified. These triplets on the first 10 ranks are given in Table 2. Additionally to the start codon ATG the triplets GAG, AGG, and GGA were assigned the largest weight values. These triplets are all contained in the sequence TAAGGAGGT, which is known to be of importance for translation initiation sites because it is the sequence that will bind to the 16S rRNA 3' terminal sequence of the ribosome. This sequence is called Shine-Dalgarno Sequence [26, 27]. Obviously the kernel uses the presence of triplets occurring in the Shine-Dalgarno sequence for discrimination.

The medians of the weights for the potential start codons were 5.6 for ATG, 3.58 for TTG, and 2.45 for GTG. That is, the presence of ATG appears to be a relevant feature, whereas GTG and TTG are not as important as ATG. In all positive as well as negative sequence patterns there is a potential start codon at the positions 33–35. Still, the frequency of ATG at this position is considerably higher in positive than in negative

Table 1. Optimized smoothing parameter for the oligo and the weighted oligo kernel

	oligo kernel (grid search)	weighted oligo kernel (CMA-ES)
Mean	1.86	1.71
Median	1.9	1.83
0.25 quantile	1.5	1.34
0.75 quantile	2.3	2.12

Table 2. The 3-mers of major importance for classification

3-mer	GAG	ATG	AGG	GGA	GGC	GCT	CAA	TTG	TCC	GGG
weight	6.23	5.6	5.36	5.29	5	4.81	4.05	3.58	3.45	3.41

Table 3. Classification performance in percent for 40 trials with different partitionings of the data

	oligo kernel (grid search)	locality improved kernel (grid search)	weighted oligo kernel (CMA-ES)
Mean	84.86	85.60	86.02
Median	85.01	86.01	86.41
0.25 quantile	83.90	84.63	84.79
0.75 quantile	86.42	86.78	87.08

examples. The initiation codon of more than 90 % of prokaryotic genes is ATG [22]. The rule of thumb "a pattern is positive if the start codon is ATG and negative otherwise", which would lead to a classification accuracy of about 72% when applied to our data, can be implemented with the evolved kernel weights. However, more sophisticated features based on the triplets with large weights in Table 2 can overrule the presence or absence of ATG.

The classification results are given in Table 3. The median of the classification performance of the 3-mer oligo kernel with equal weighting is 85.01%. Introducing the weights for the individual 3-mers in the oligo kernel and optimizing them using CMA-ES leads to an increase of the classification performance to 86.41%. The results achieved by the weighted oligo kernel are significantly better than those of the oligo kernel with equal weights and the smoothing parameter as only adjustable variable (Wilcoxon rank-sum test, $p < 0.01$).

The median of the locality improved kernel parameters adjusted by grid search was two for both l and d. That is, the nucleotides were only compared within a small window. This is in accordance with the results for σ in the oligo kernels. The median of the classification performance reached by the locality improved kernel is 86.01%, that is, between the 3-mer oligo kernel with equal weights and the evolutionary optimized 3-weighted oligo kernel. However, the differences are not statistically significant (Wilcoxon rank-sum test, $p > 0.05$).

5 Conclusion and Outlook

A task specific choice of the kernel can significantly improve kernel-based machine learning. Often a parameterized family of kernel functions is considered so that the kernel adaptation reduces to real-valued optimization. Still, the adaptation of complex kernels requires powerful optimization methods that can adapt multiple parameters efficiently. When the considered space of kernel functions lacks a differentiable structure or the model selection criterion is non-differentiable, a direct search method is needed. The covariance matrix adaptation evolution strategy (CMA-ES) is such a powerful, direct algorithm for real-valued hyperparameter selection.

In biological sequence analysis, the CMA-ES allows for a more task specific adaptation of sequence kernels. Because multiple parameters can be adapted, it is possible to adjust new weighting variables in the oligo kernel to control the influence of every oligomer individually. Further, the cross-validation error can directly be optimized (i.e., without smoothening).

We demonstrated the discriminative power of the oligo kernel and the benefits of the evolutionary model selection approach by applying them to prediction of prokaryotic translation initiation sites (TISs). The adapted weighted oligo kernel leads to improved results compared to kernel functions with less adaptable parameters, which were optimized by grid-search. Furthermore, it is possible to reveal biologically relevant information from analyzing the evolved weighting parameters. For the prediction of TISs, for example, triplets referring to the Shine-Dalgarno sequence are used for discrimination.

Acknowledgments

The authors thank Nico Pfeifer for his work on the original implementation of the oligo kernel and the TIS classification.

References

1. Schölkopf, B., Tsuda, K., Vert, J.P., eds.: Kernel Methods in Computational Biology. Computational Molecular Biology. MIT Press (2004)
2. Meinicke, P., Tech, M., Morgenstern, B., Merkl, R.: Oligo kernels for datamining on biological sequences: A case study on prokaryotic translation initiation sites. BMC Bioinformatics **5** (2004)
3. Chapelle, O., Vapnik, V., Bousquet, O., Mukherjee, S.: Choosing multiple parameters for support vector machines. Machine Learning **46** (2002) 131–159
4. Glasmachers, T., Igel, C.: Gradient-based adaptation of general Gaussian kernels. Neural Computation **17** (2005) 2099–2105
5. Keerthi, S.S.: Efficient tuning of SVM hyperparameters using radius/margin bound and iterative algorithms. IEEE Transactions on Neural Networks **13** (2002) 1225–1229
6. Hansen, N., Ostermeier, A.: Completely derandomized self-adaptation in evolution strategies. Evolutionary Computation **9** (2001) 159–195
7. Friedrichs, F., Igel, C.: Evolutionary tuning of multiple SVM parameters. Neurocomputing **64** (2005) 107–117
8. Igel, C., Wiegand, S., Friedrichs, F.: Evolutionary optimization of neural systems: The use of self-adaptation. In: Trends and Applications in Constructive Approximation. Number 151 in International Series of Numerical Mathematics. Birkhäuser Verlag (2005) 103–123
9. Schölkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press (2002)
10. Degroeve, S., Beats, B.D., de Peer, Y.V., Rouzé, P.: Feature subset selection for splice site prediction. Bioinformatics **18** (2002) 75–83
11. Leslie, C., Eskin, E., Noble, W.S.: The spectrum kernel: A string kernel for SVM protein classification. In Altman, R.B., et al., eds.: Proceedings of the Pacific Symposium on Biocomputing, World Scientific (2002) 564–575
12. Eads, D.R., et al.: Genetic algorithms and support vector machines for time series classification. In Bosacchi, B., Fogel, D.B., Bezdek, J.C., eds.: Applications and Science of Neural Networks, Fuzzy Systems, and Evolutionary Computation V. Volume 4787 of Proceedings of the SPIE. (2002) 74–85
13. Fröhlich, H., Chapelle, O., Schölkopf, B.: Feature selection for support vector machines using genetic algorithms. International Journal on Artificial Intelligence Tools **13** (2004) 791–800

14. Igel, C.: Multi-objective model selection for support vector machines. In Coello, C.A.C., Zitzler, E., Aguirre, A.H., eds.: Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005). Volume 3410 of LNAI., Springer-Verlag (2005) 534–546
15. Jong, K., Marchiori, E., van der Vaart, A.: Analysis of proteomic pattern data for cancer detection. In Raidl, G.R., et al., eds.: Applications of Evolutionary Computing. Number 3005 in LNCS, Springer-Verlag (2004) 41–51
16. Miller, M.T., Jerebko, A.K., Malley, J.D., Summers, R.M.: Feature selection for computer-aided polyp detection using genetic algorithms. In Clough, A.V., Amini, A.A., eds.: Medical Imaging 2003: Physiology and Function: Methods, Systems, and Applications. Volume 5031 of Proceedings of the SPIE. (2003) 102–110
17. Pang, S., Kasabov, N.: Inductive vs. transductive inference, global vs. local models: SVM, TSVM, and SVMT for gene expression classification problems. In: International Joint Conference on Neural Networks (IJCNN). Volume 2., IEEE Press (2004) 1197–1202
18. Runarsson, T.P., Sigurdsson, S.: Asynchronous parallel evolutionary model selection for support vector machines. Neural Information Processing – Letters and Reviews **3** (2004) 59–68
19. Shi, S.Y.M., Suganthan, P.N., Deb, K.: Multi-class protein fold recognition using multi-objective evolutionary algorithms. In: IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, IEEE Press (2004) 61–66
20. Beyer, H.G., Schwefel, H.P.: Evolution strategies: A comprehensive introduction. Natural Computing **1** (2002) 3–52
21. Hansen, N., Kern, S.: Evaluating the CMA evolution strategy on multimodal test functions. In Yao, X., et al., eds.: Parallel Problem Solving from Nature (PPSN VIII). Volume 3242 of LNCS., Springer-Verlag (2004) 282–291
22. Gualerzi, C.O., Pon, C.L.: Initiation of mRNA translation in procaryotes. Biochemistry **29** (1990) 5881–5889
23. Zien, A., et al.: Engineering support vector machine kernels that recognize translation initiation sites. Bioinformatics **16** (2000) 799–807
24. Rudd, K.E.: Ecogene: a genome sequence database for *Escherichia coli* K-12. Nucleic Acids Research **28** (2000) 60–64
25. Blattner, F.R., et al.: The complete genome sequence of Escherichia coli K-12. Science **277** (1997) 1453–1462
26. Kozak, M.: Initiation of translation in prokaryotes and eukaryotes. Gene **234** (1999) 187–208
27. Shine, J., Dalgarno, L.: The 3'-terminal sequence of *Escherichia coli* 16S ribosomal RNA: Complementarity to nonsense triplets and ribosome binding sites. PNAS **71** (1974) 1342–1346