

POIMs: positional oligomer importance matrices—understanding support vector machine-based signal detectors

Sören Sonnenburg^{1,†,*}, Alexander Zien^{1,2,3,†}, Petra Philips² and Gunnar Rätsch²

¹Fraunhofer Institute FIRST, Department IDA, Kekuléstr. 7, 12489 Berlin, ²Friedrich Miescher Laboratory, Max Planck Society, Spemannstr. 39 and ³Max Planck Institute for Biological Cybernetics, Spemannstr. 38, 72076 Tübingen, Germany

ABSTRACT

Motivation: At the heart of many important bioinformatics problems, such as gene finding and function prediction, is the classification of biological sequences. Frequently the most accurate classifiers are obtained by training support vector machines (SVMs) with complex sequence kernels. However, a cumbersome shortcoming of SVMs is that their learned decision rules are very hard to understand for humans and cannot easily be related to biological facts.

Results: To make SVM-based sequence classifiers more accessible and profitable, we introduce the concept of *positional oligomer importance matrices* (POIMs) and propose an efficient algorithm for their computation. In contrast to the raw SVM feature weighting, POIMs take the underlying correlation structure of k -mer features induced by overlaps of related k -mers into account. POIMs can be seen as a powerful generalization of sequence logos: they allow to capture and visualize sequence patterns that are relevant for the investigated biological phenomena.

Availability: All source code, datasets, tables and figures are available at <http://www.fml.tuebingen.mpg.de/raetsch/projects/POIM>.

Contact: Soeren.Sonnenburg@first.fraunhofer.de

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

For many sequence classification problems, support vector machines (SVMs) (Schölkopf and Smola, 2002; Vapnik, 1995) with the right choice of sequence kernels perform better than other state-of-the-art methods, as exemplified in Table 1. While this success can in part be attributed to the SVM algorithm and the statistical learning theory that underlies it (Vapnik, 1995), it is essential to use appropriate kernels and features. In order to achieve the best prediction results, it typically pays off to rather include many, potentially weak features than to manually pre-select a small set of discriminative features. For instance, the SVM-based translation initiation start (TIS) signal detector *Startscan* (Saeys et al., 2007), which relies on a relatively small set of carefully designed features, shows a considerably higher error rate than an SVM with a standard kernel that implies a very high-dimensional feature space (cf. Table 1).

The best methods in Table 1 are all based on SVMs that work in feature spaces that exhaustively represent the incidences of all

k -mers up to a certain maximum length K . There are two cases: the k -mers are either (i) summarized over all positions or (ii) considered separately for each position. For (i) there are the popular *spectrum kernel* without (Leslie et al., 2002) and with mismatches (Leslie et al., 2003); for (ii) we use the *weighted degree kernel* without (WD; Rätsch and Sonnenburg, 2004) and with shift (WDS; Rätsch et al., 2005).

Nowadays, SVMs with string kernels can be trained efficiently on millions of DNA sequences even for large orders K (e.g. Sonnenburg et al., 2007a), thereby inducing enormous feature spaces (for instance, $K=30$ gives rise to more than $4^{30} > 10^{18}$ k -mers). Such feature spaces supply a solid basis for accurate predictions as they allow to capture complex relationships (e.g. binding site requirements). From an application point of view, however, they are yet unsatisfactory as they offer little scientific insight about the nature of these relationships. The reason is that SVM classifiers $\hat{y} = \text{sign}(f(\mathbf{x}))$ employ a kernel expansion,

$$f(\mathbf{x}) = \sum_{i=1}^N \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b, \quad (1)$$

where $(\mathbf{x}_i, y_i)_{i=1, \dots, N}$, with $y_i \in \{+1, -1\}$, are the N training examples (Schölkopf and Smola, 2002). Thus, SVMs use a weighting α over training examples that only indirectly relate to features. One idea to remedy this problem is to characterize input variables by their correlation with the weight vector α (Üstün et al., 2007); however, the importance of features in the induced feature space remains unclear.

Partial relief is offered by multiple kernel learning (MKL; e.g. Lanckriet et al., 2004; Rätsch et al., 2006). In MKL, convex combinations of M kernels are considered, i.e. $k(\mathbf{x}_i, \mathbf{x}_j) := \sum_{m=1}^M \beta_m k_m(\mathbf{x}_i, \mathbf{x}_j)$ with $\beta_m \geq 0$ and $\sum_{m=1}^M \beta_m = 1$. For appropriately designed sub-kernels k_m , the optimized combination coefficients β can then be used to highlight which parts of an input sequence are important for discrimination (Rätsch et al., 2006). The use of the l_1 -norm constraint ($\sum_{m=1}^M \beta_m = 1$) causes the resulting β to be sparse, however at the price of discarding relevant features, which may lead to inferior performance.

An alternative approach is to keep the SVM decision function unaltered, and to find adequate ways to ‘mine’ the decision boundary for good explanations of its high accuracy. A natural way is to compute and analyze the normal vector of the separation in feature space, $\mathbf{w} = \sum_{i=1}^N \alpha_i y_i \Phi(\mathbf{x}_i)$, where Φ is the feature mapping associated to the kernel k . This has been done, for instance, in cognitive sciences to understand the differences in human perception of pictures showing male and female faces. The resulting normal vector \mathbf{w} was relatively easy to understand for humans since it can

* To whom correspondence should be addressed.

† The authors wish to be known that, in their opinion, the first two authors should be regarded as joint First Authors.

Table 1. Comparison of SVM performance (second column) versus competing state-of-the-art classifiers (third column) on six different DNA signal detection problems

Signal detection problem to be solved	SVM performance	SVM-based approach and string kernel names	Performance of competitor	Competing approach
Transcription start	26.2% auPRC	WDS and spectrum (ARTS, Sonnenburg <i>et al.</i> , 2006b)	11.8% auPRC	RVM (Eponine, Down and Hubbard, 2002)
Acceptor splice site	54.4% auPRC	WDS (Sonnenburg <i>et al.</i> , 2007b)	16.2% auPRC	IMC (Sonnenburg <i>et al.</i> , 2007b)
Donor splice site	56.5% auPRC	WDS (Sonnenburg <i>et al.</i> , 2007b)	25.0% auPRC	IMC (Sonnenburg <i>et al.</i> , 2007b)
Alternative splicing	89.7% auROC	WDS (RASE, Ratsch <i>et al.</i> , 2005)	-	-
Trans-splicing	95.2% auROC	WD (mGene, Schweikert <i>et al.</i> , manuscript in preparation)	-	-
Translation initiation	10.7% Se80	WD (mGene, Schweikert <i>et al.</i> , manuscript in preparation)	12.5% Se80	PWM, ICM (Startscan, Saeys <i>et al.</i> , 2007)

The chosen best SVMs employ spectrum, WD or WDS kernels. Note that performance measures differ. AuROC denotes the area under the receiver operator characteristic curve and auPRC the area under the precision recall curve; for both, larger values correspond to better performance. Se80 is the false positive rate at a true positive rate of 80% (lower values are better). Transcription start site (TSS): among the best TSS recognizers is the relevance vector machine (RVM) based Eponine, which is clearly outperformed by our ARTS TSS detector. Acceptor and donor splice sites: the best existing splice site detectors are SVM based (Sonnenburg *et al.*, 2007b); we therefore deliberately compare our methods to the popular inhomogeneous Markov chains (IMC), which achieve less than half of the auPRC on human splice sites. Alternative and trans-splice sites in *Caenorhabditis elegans*: to the best of our knowledge no other *ab initio* approaches are available. Translation initiation sites: for TIS recognition we compare with Startscan (SMB, 2007) which is based on positional weight matrices (PWM) and interpolated context models (ICM). Our WD-kernel SVM, trained using default settings $C=1$, $d=20$ (no model selection) on the dataset from Saeys *et al.* (2007), already performs favorably.

be represented as an image (Graf *et al.*, 2006). Such approach is only feasible if there exists an explicit and manageable representation of Φ for the kernel at hand. Fortunately, for most string kernels we can also compute such weight vector which leads to weightings over all possible k -mers. However, it seems considerably more difficult to represent such weightings in a way humans can easily understand. There have been first attempts in this direction (Meinicke *et al.*, 2004), but the large number of k -mers and their dependence due to overlaps at neighboring positions still remain an obstacle in representing complex SVM decision boundaries.

In this work, we address this problem by considering new measures for k -mer-based scoring schemes (such as SVMs with string kernels) useful for the understanding of complex local relationships that go beyond the well-known sequence logos. For this, we first compute the *importance* of each k -mer (up to a certain length K) at each position as its expected contribution to the total score $f(\mathbf{x})$. The resulting *Positional Oligomer Importance Matrices* (POIMs) can be used to rank and visualize k -mer-based scoring schemes. Note that a ranking based on \mathbf{w} is not necessarily meaningful: due to the dependencies of the features there exist $\mathbf{w}' \neq \mathbf{w}$ that implement the same classification, but yield different rankings. In contrast, our importance values are well-defined and have the desired semantics. The lowest order POIM ($k=1$) essentially conveys the same information as is represented in a sequence logo. However, unlike sequence logos, POIMs naturally generalize to higher order nucleotide patterns.

The article is structured as follows. In Section 2 we introduce POIMs for visualization and feature extraction. In Section 3 we use artificial data to show that POIMs easily out-compete MKL and the SVM weight \mathbf{w} . We then analyze POIMs of state-of-the-art SVM-based signal detectors for recognizing acceptor splice, transcription start and *trans*-splicing sites (TRSSs). We show that POIMs recover many known motifs: they exactly pin-point length, location and typical sequences of motifs. We close the article with a discussion and an outlook on future work (Section 4).

2 METHODS

First we introduce the necessary background, then define POIMs, provide recursions for efficient computation, and finally describe visualization and analysis.

2.1 Linear positional oligomer scoring systems

Given an alphabet Σ , here the DNA nucleotides $\Sigma = \{A, C, G, T\}$, let $\mathbf{x} \in \Sigma^L$ be a sequence of length L . A sequence $\mathbf{y} \in \Sigma^k$ is called a k -mer or oligomer of length k . A *positional oligomer* (PO) is defined by a pair $(\mathbf{y}, i) \in \mathcal{I} := \bigcup_{k=1}^K (\Sigma^k \times \{1, \dots, L-k+1\})$, where \mathbf{y} is the subsequence of length k and i is the position at which it begins within the sequence of length L . We consider scoring systems of order K (that are based on POs of lengths $k \leq K$) defined by a *weighting function* $w: \mathcal{I} \rightarrow \mathbb{R}$. Let the *score* $s(\mathbf{x})$ be defined as a sum of PO weights:

$$s(\mathbf{x}) := \sum_{k=1}^K \sum_{i=1}^{L-k+1} w(\mathbf{x}[i]^k, i) + b, \quad (2)$$

where b is a constant offset (bias), and we write $\mathbf{x}[i]^k := x_i x_{i+1} \dots x_{i+k-1}$ to denote the substring of \mathbf{x} that starts at position i and has length k . Many classifiers implement such a scoring system as will be shown below.

2.1.1 The WD kernel The WD kernel (Ratsch and Sonnenburg, 2004) of order K compares two sequences \mathbf{x} and \mathbf{x}' of equal length L by counting k -mer matches of lengths $k \in \{1, \dots, K\}$ with predefined weights β_k :

$$k(\mathbf{x}, \mathbf{x}') = \sum_{k=1}^K \beta_k \sum_{i=1}^{L-k+1} \mathbb{I}\{\mathbf{x}[i]^k = \mathbf{x}'[i]^k\},$$

where $\mathbb{I}\{\cdot\}$ is the indicator function (see also Fig. 1 for illustration). A feature mapping $\Phi(\mathbf{x})$ is defined by a vector representing each PO of length $\leq K$: if it is present in \mathbf{x} then the vector entry is $\sqrt{\beta_k}$ and 0 otherwise (Ratsch and Sonnenburg, 2004). It can be easily seen that $\Phi(\mathbf{x})$ is an explicit representation of the WD kernel, i.e. $k(\mathbf{x}, \mathbf{x}') = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}')$.

When training any kernel method (e.g. an SVM or a kernel regression method) with this kernel, the resulting function is a weighted sum of kernel evaluations (1). If there exist an explicit feature map, the function $f(\mathbf{x})$ can

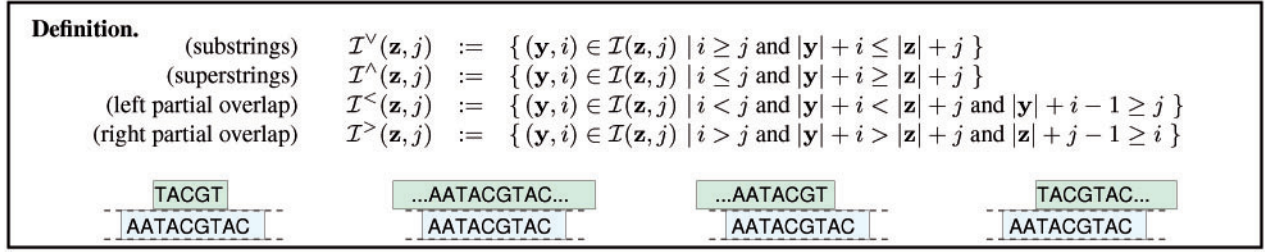


Fig. 2. Substrings, superstrings, left partial overlaps and right partial overlaps: definition and examples for the string AATACGTAC.

Theorem. Let $\mathbf{z} \in \Sigma^k$ and $Q(\mathbf{z}, j)$ be defined as in (3). Then

$$Q(\mathbf{z}, j) = u(\mathbf{z}, j) - \sum_{\mathbf{z}' \in \Sigma^k} Pr(\mathbf{x}[j] = \mathbf{z}') u(\mathbf{z}', j), \quad (8)$$

where u decomposes as $u(\mathbf{z}, j) = u^\vee(\mathbf{z}, j) + u^\wedge(\mathbf{z}, j) + u^<(\mathbf{z}, j) + u^>(\mathbf{z}, j) - w(\mathbf{z}, j)$ and $u^\vee, u^\wedge, u^<, u^>$ are computed recursively by

$$u^\vee(\sigma\mathbf{z}\tau, j) = w_{(\sigma\mathbf{z}\tau, j)} + u^\vee(\sigma\mathbf{z}, j) + u^\vee(\mathbf{z}\tau, j+1) - u^\vee(\mathbf{z}, j+1) \quad \text{for } \sigma, \tau \in \Sigma$$

$$u^\wedge(\mathbf{z}, j) = w_{(\mathbf{z}, j)} - \sum_{(\sigma, \tau) \in \Sigma^2} Pr(\mathbf{x}[j-1] = \sigma) Pr(\mathbf{x}[j+k] = \tau) u^\wedge(\sigma\mathbf{z}\tau, j-1) \\ + \sum_{\sigma \in \Sigma} Pr(\mathbf{x}[j-1] = \sigma) u^\wedge(\sigma\mathbf{z}, j-1) + \sum_{\tau \in \Sigma} Pr(\mathbf{x}[j+p] = \tau) u^\wedge(\mathbf{z}\tau, j)$$

$$u^<(\mathbf{z}, j) = \sum_{\sigma \in \Sigma} Pr(\mathbf{x}[j-1] = \sigma) \sum_{l=1}^{\min\{k, K\}-1} L(\sigma(\mathbf{z}[1]^l), j-1)$$

$$u^>(\mathbf{z}, j) = \sum_{\tau \in \Sigma} Pr(\mathbf{x}[j+k] = \tau) \sum_{l=1}^{\min\{k, K\}-1} R(\mathbf{z}[k-l+1]^l \tau, j+p-l),$$

$$L(\mathbf{z}, j) = w_{(\mathbf{z}, j)} + \sum_{\sigma \in \Sigma} Pr(\mathbf{x}[j-1] = \sigma) L(\sigma\mathbf{z}, j-1) \quad \text{and} \quad R(\mathbf{z}, j) = w_{(\mathbf{z}, j)} + \sum_{\tau \in \Sigma} Pr(\mathbf{x}[j+p] = \tau) R(\mathbf{z}\tau, j).$$

Fig. 3. The theorem which enables efficient POIM computation for zeroth-order Markov chains (Zien *et al.*, 2007).

for all k -mers \mathbf{z} up to length P at all positions $1, \dots, L-p+1$, which takes advantage of shared intermediate terms. This results in a recursive algorithm operating on string prefix data structures and therefore is efficient enough to be applied to real biological analysis tasks.

The crucial idea is to treat the POs in $\mathcal{I}(\mathbf{z}, j)$ separately according to their relative position to (\mathbf{z}, j) . To do so, we subdivide the set $\mathcal{I}(\mathbf{z}, j)$ into substrings, superstrings, left partial overlaps and right partial overlaps of (\mathbf{z}, j) , as illustrated in Figure 2. The function u can be decomposed correspondingly; see Equation (8) in Figure 3. This figure also summarizes all the previous observations in the central POIM computation theorem. The complete derivation can be found in Zien *et al.* (2007).

Once \mathbf{w} is available as a suffix trie (Sonnenburg *et al.*, 2007a), the required amounts of memory and computation time for computing POIMs are dominated by the size of the output, i.e. $\mathcal{O}(|\Sigma|^k \cdot L)$. The recursions are implemented in the freely available toolbox SHOGUN² (which also offers a non-positional version for the spectrum kernel) and applicable online.³

²available from <http://www.shogun-toolbox.org>


³via GALAXY at <http://galaxy.fml.tuebingen.mpg.de/>

2.3 Ranking features and condensing information for visualization

Given the POIMs, we can analyze POs for their contributions to the scoring function. Now we discuss several methods to visualize the information in POIMs and to find relevant POs. In the following, we will use the term motif either synonymously for PO, or for a set of POs that are similar (e.g. that share the same oligomer at a small range of neighboring positions).

2.3.1 Ranking tables A simple analysis of a POIM is to sort all POs by their importance $Q(\cdot)$ or absolute importance $|Q(\cdot)|$. As argued above, the highest ranking POs are likely to be related to relevant motifs. Examples of such tables can be found in the Supplementary Material.

2.3.2 POIM Plots We can visualize an entire POIM of a fixed order $k \leq 3$ in the form of a heat map: each PO (\mathbf{z}, j) is represented by a cell, indexed by its position j on the x -axis and the k -mer \mathbf{z} on the y -axis (e.g. in lexical ordering), and the cell is colored according to the importance value $Q(\mathbf{z}, j)$. This allows to quickly overview the importance of all possible POs at all positions. An example for $k=1$ can be found in Figure 6. There also the corresponding sequence logo is shown.



$$q_{\max}^{k,j} := \max_{\mathbf{z} \in |\Sigma|^k} |Q(\mathbf{z}, j)|$$

$$D(k, j) := q_{\max}^{k,j} - \max\{q_{\max}^{k,j}, q_{\max}^{k,j+1}\} \quad (9)$$

Fig. 4. Two $(k-1)$ -mers are covered by a k -mer.

However, for $k > 3$ such visualization ceases to be useful due to the exploding number of k -mers. Similarly, ranking lists may become too long to be accessible to human processing. Therefore we need views that aggregate over all k -mers, i.e. that show PO importance just as a function of position j and order k . Once a few interesting areas (j, k) are identified, the corresponding POs can be further scrutinized (e.g. by looking at local rank lists) in a second step. In the following paragraphs we propose three approaches for the first, summarizing step.

2.3.3 POIM weight mass At each position j , the total importance can be computed by summing the absolute importance of all k -mers at this position, $\text{weight_mass}_k(j) = \sum_{\mathbf{z} \in \Sigma^k} |Q(\mathbf{z}, j)|$. Several such curves for different k can be shown simultaneously in a single graph. An example can be found in Figure 7B.

2.3.4 Differential POIMs Here we start by taking the maximum absolute importance over all k -mers for a given length k and at a given position j . We do so for each $k \in \{1, \dots, P\}$ and each $j \in \{1, \dots, L\}$. Then we subtract the maximal importance of the two sets of $(k-1)$ -mers covered by the highest scoring k -mer at the given position (Fig. 4). This results in the importance gain by considering longer POs. As we demonstrate in the next section, this allows to determine the length of relevant motifs. Figure 5A–D shows such images.

2.3.5 POIM diversity A different way to obtain an overview over a large number of k -mers for a given length k is to visualize the distribution of their importances. To do so, we approximate the distribution at each position by a mixture of two normal distributions, thereby dividing them into two clusters. We set the values in the lower cluster to zero, and sort all importances in descending order. The result is depicted in a heat map just like a POIM plot. Here we do not care that individual importance values cannot be visually mapped to specific k -mers, as the focus is on understanding the distribution. An example is given in Figure 7C.

2.3.6 k -mer scoring overview Previously, the scoring and visualization of features learned by SVMs was performed according to the values of the weight vector \mathbf{w} (Meinicke et al., 2004; Sonnenburg et al., 2007a). In order to show the benefit of our POIM techniques in comparison to such techniques, we also display matrices visualizing the position-wise maximum of the absolute value of the raw weight vector \mathbf{w} over all possible k -mers at this position, $\text{KS}(p, j) = \max_{\mathbf{z} \in \Sigma^k} |(\mathbf{z}, j)|$. We will also display the *Weight Plots* and the *Weight Mass* for the weight matrices SVM- \mathbf{w} in the same way as for the importance matrices. Differential plots for SVM- \mathbf{w} do not make sense, as the weights for different orders are not of the same order of magnitude. Figure 5E–H shows such overview images.

3 RESULTS

3.1 POIMs reveal motifs which remain hidden in sequence logos and SVM weights

As a first step we demonstrate our method on two simulations with artificial data.

3.1.1 Two motifs at fixed positions In our first example we reuse a toy data set of Sonnenburg et al. (2005): two motifs of length seven, with consensus sequences GATTACA and AGTAGTG, are planted at two fixed positions into random sequences (Sonnenburg et al., 2005). To simulate motifs with different degrees of conservations we also mutate these consensus sequences, and then compare how well different techniques can recover them. More precisely, we generate a data set with 11 000 sequences of length $L=50$ with the following distribution: at each position the probability of the symbols $\{A, T\}$ is $1/6$ and for $\{C, G\}$ it is $2/6$. We choose 1000 sequences to be positive examples: we plant our two POs, (GATTACA, 10) and (AGTAGTG, 30), and randomly replace s symbols in each PO with a random letter. We create four different versions by varying the mutation level $s \in \{0, 2, 4, 5\}$. Each data set is randomly split into 1000 training examples and 10 000 validation examples. For each s , we train an SVM with WD kernel of degree 20 exactly as in Sonnenburg et al. (2005). We also reproduce the results of Sonnenburg et al. (2005) obtained with a WD kernel of degree 7, in which MKL is used to obtain a sparse set of relevant pairs of position and degree.

The results can be seen in Figure 5. We start with the first column, which corresponds to the unmutated case $s=0$. Due to its sparse solution, MKL (I–L) characterizes the positive class by a single 3mer in the GATTACA. Thus MKL fails to identify the complete consensus sequences. The K -mer scoring matrices (E–H) are able to identify two clusters of important oligomers at and after positions 10 and 30; however they assign highest impact to shorter oligomers. Only the differential POIMs (A–D) manage to identify the exact length of the embedded POs: they do not only display that 7mers up to $k=7$ are important, but also that exactly 7mers at position 10 and 30 are most important.

Moving through the columns to the left, the mutation level increases. Simultaneously the performance deteriorates, as expected. In the second column (2/7 mutations), differential POIMs still manage to identify the exact length of the embedded motifs, unlike the other approaches. For higher mutation rates even the POIMs assign more importance to shorter POs, but they continue to be closer to the truth than the two other approaches. In Figure 6 we show the POIM plot for 1mer versus sequence logos for this level of mutation. As one can see, 1mer POIMs can capture the whole information contained in sequence logos.

3.1.2 Mutated motif at varying positions In order to make our experiment more realistic, we consider motifs with positional shift. First 5000 training and test sequences are created by drawing uniformly from $\{A, C, G, T\}^{100}$. For half of the sequences, the positive class, we randomly insert the 7mer GATTACA, with one mutation at a random position. The position j of insertion follows a normal distribution with mean 0 and standard deviation 7 (thus for 50% of the cases, $j \in [-5, +5]$). We train an SVM with WDS kernel of degree 10 with constant shift 30 to discriminate between the inseminated sequences and the uniform ones; it achieves an accuracy of 80% on the test set. As displayed in Figure 7, both the sequence logo and the SVM- \mathbf{w} fail to make the consensus and its length apparent, whereas the POIM-based techniques identify length and positional distribution. Please note that Gibbs sampling methods have been used to partially solve this problem for PWMs. Such methods can also be used in conjunction with SVMs.

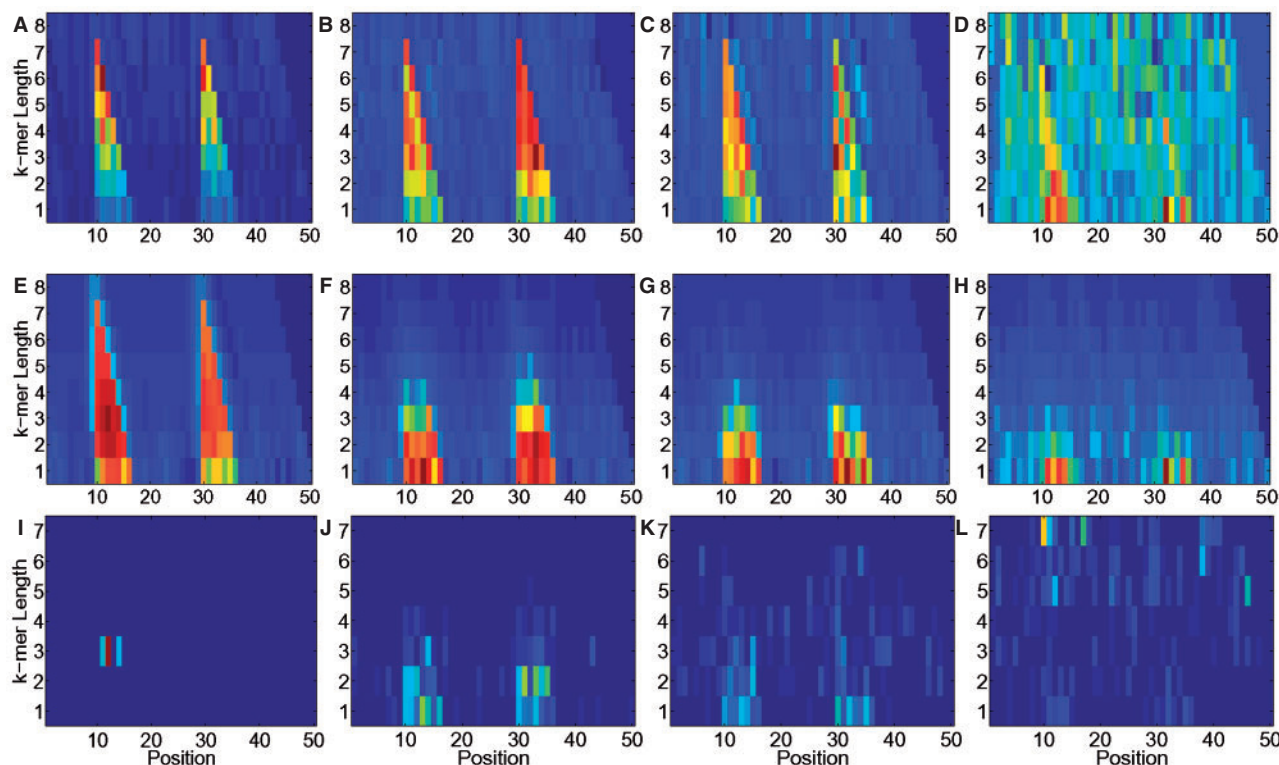


Fig. 5. Comparison of different visualization techniques for the fixed-position-motifs experiment. Motifs GATTACA and AGTAGTG were inserted at positions 10 and 30 respectively with growing level of mutation (i.e. number of nucleotides randomly substituted in the motifs) from left to right. SVMs classifiers were trained to distinguish random sequences from sequences with the (mutated) motifs GATTACA and AGTAGT inserted. (A–D) We computed Differential POIMs [Equation (9)] for up to 8mers, from a WD-kernel SVM of order 20. Here each figure displays the importance of k -mer lengths (y-axis) for $k = 1 \dots 8$ at each position (x -axis) ($i = 1 \dots 50$) as a heat map. Red and yellow color denotes relevant motifs, dark blue corresponds to motifs not conveying information about the problem. 1mers are at the bottom of the plot, 8mers at the top. (E–H) K -mer scoring overview (SVM-w) was computed using the same setup as for differential POIMs. The SVM-w is again displayed as a heat map. (I–L) It was obtained using MKL (averaged weighting obtained using 100 bootstrap runs, Rättsch *et al.*, 2006). Again the result is displayed as a heat map, but for 1-to 7mers only. For a more detailed discussion see text.

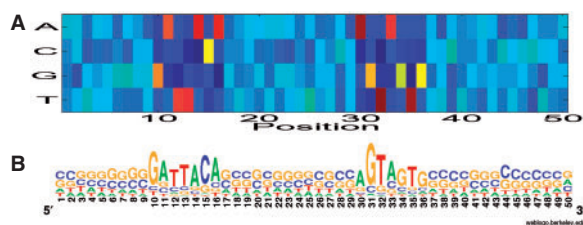


Fig. 6. 1mer POIM plot (A) versus sequence logo (B) for motifs GATTACA and AGTAGTG at positions 10 and 30, respectively, with 4-out-of-7 mutations in the motifs.

3.2 Analysis of biological data

We now demonstrate the power of our technique on three real biological tasks, namely the recognition of splice sites, TSSs and TRSSs. Note that the SVMs investigated here are among the most accurate existing sensors for these signals (cf. Table 1). It is thus of highest interest which clues the SVM uses to distinguish true sites from decoys: the most important POs can be suspected to be the biological traits that determine the cellular events. However, the number of relevant POs may not be small, because (unlike common

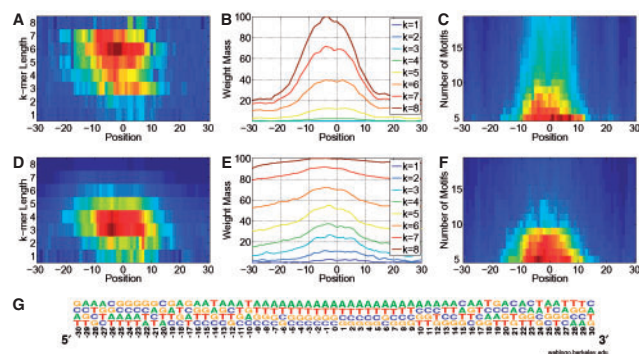


Fig. 7. Comparison of different visualization techniques for the varying-positions-motif experiment. The mutated motif GATTACA was inserted at positions $0 + -13$ in uniformly distributed sequences. (A–C) shows the Differential POIM matrices [cf. Equation (9)] as a heat map, the POIM weight mass for different $k = 1 \dots 8$ and the POIM k -mer diversity for $k = 3$ as a heat map; (D–F) shows the SVM-w overview plot as a heat map, the SVM-w weight mass also for $k = 1 \dots 8$ and the k -mer diversity for $k = 3$ as a heat map; (G) sequence logo.

motif finders) we do not yet pool them. Due to space constraints we have to omit many detailed figures and lists of the extracted

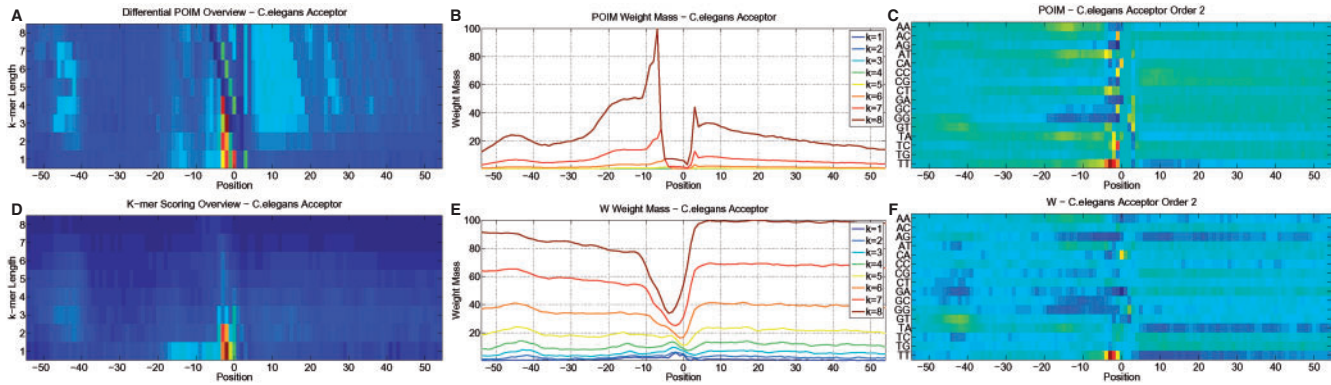


Fig. 8. Comparison of different visualization techniques for the *C. elegans* splice data set based on (A–C) POIM matrices versus (D–F) weight matrices. Position 0 is the splice site.

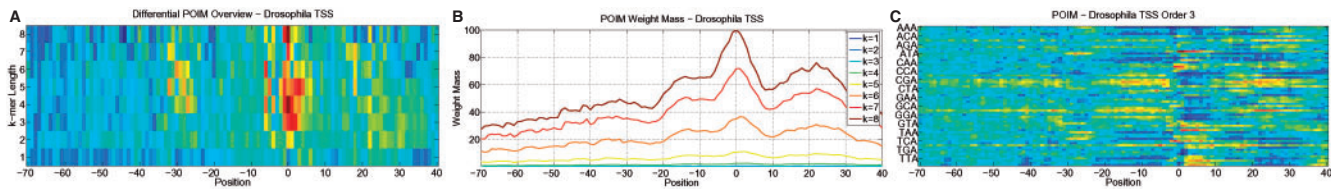


Fig. 9. POIM visualization for the TSSs of *D. melanogaster*. (A–C): differential POIMs, POIM weight mass for $k = 1 \dots 8$ and POIMs for $k = 3$ are displayed.

motifs for the following analyses, which however can be found in the supplementary material.

3.2.1 Splice site analysis First, we apply our techniques to a acceptor splice site recognition task derived from mRNA and genomic sequences of *C.elegans*. The data set contains 262 421 DNA sequences of length 141 nucleotides, each anchored at a AG, which is the acceptor splice site consensus [see Rättsch et al. (2006) for details]. We train an SVM ($C = 1$) with a WD kernel ($K = 20$) on the first 100 000 examples; the resulting classifier achieves 99.7% auROC on the remaining 162 421 examples. POIM results are depicted in Figure 8.

Upstream: A relatively weak, but surprising signal can be seen around 43 nt upstream of the acceptor splice site. Looking at the dinucleotide weighting w (Fig. 8F), one can see an increased weighting for the GT dinucleotide. This leads to the discovery of the upstream donor splice site. In *C.elegans* this site is indeed often located 40–50nt upstream of the acceptor splice site. Zooming into the -55 nt to -35 nt window in the differential POIM figure (Fig. 8C and p.1 in Supplementary Material) one may notice strong signals for up to 7mers. Extracting the three highest scoring 7mers one detect the motifs **GTAAGT**, **AGTAAG**, **GTAGGT** which have high importances in this whole region, especially around -43 nt. This perfectly matches parts of the known donor consensus **AGGTAAGT**. Upstream in the interval -18 nt to -14 nt we find in the 6mer POIMs that **GGGGG** has a strong negative score, which makes it a potential silencer. Furthermore in the same region one also finds **TAAT** which is one of the known branch site signals (Harris and Senapathy, 1990).

Central: one can observe a very strong and localized signal in front of the acceptor splice site, which is located at position 0 and followed by the **AG** consensus at positions 1, 2 [see Rättsch et al. (2006)

for more details]. Extracting the highest scoring 7mers in the window from -9 nt to $+5$ nt one detects several sequences ending on **...TTTC** directly in front of the splice site. Following this known strong T-rich region are the motifs **TTTCAGG** and **TTTCAGA** scoring highest at -3 nt, recovering the known acceptor consensus **TTTTCAG** (A/G).

Downstream: finally, one also recognizes the strong penalty against T's downstream of the splice site ($+6$ nt to $+20$ nt), where the motif **TTTTTTT** scores most negative (Supplementary Material).

3.2.2 Promoter regulatory elements In the following we apply these techniques to the search for promoter regulatory elements. We train a classifier with the WDS kernel—the core ingredient of our SVM-based human TSS finder (ARTS; Sonnenburg et al., 2006b)—*Drosophila melanogaster* TSS detection. We use the same data that have been used for training and evaluation McPromoter (Ohler et al., 2002) and follow the same cross-validation procedure for training and evaluation (optimal SVM parameters after model selection: $C = 10$, $K = 10$, shift = 40). Our cross-validation performance is auROC = 96.2%, and comparable to the one achieved by McPromoter (version 2: 95.8% and version 3: 98.1%, (Ohler, 2006)). The POIM analysis is displayed in Figure 9. Inspection of POIM tables in the discriminative regions from -33 to -21 upstream, around position 0, and downstream $+15$ to $+35$ retrieve the following known motifs:

Upstream: extracting the highest scoring 7mers from the POIM tables in -70 nt to -16 nt let us find several variants of the TATA-box's core motif **TATAAA** (e.g. Burke and Kadonaga, 1997): **TATAAAA**, **GTATAAA**, **ATATAAA**, **TATATAA**, **TATAAAG**,

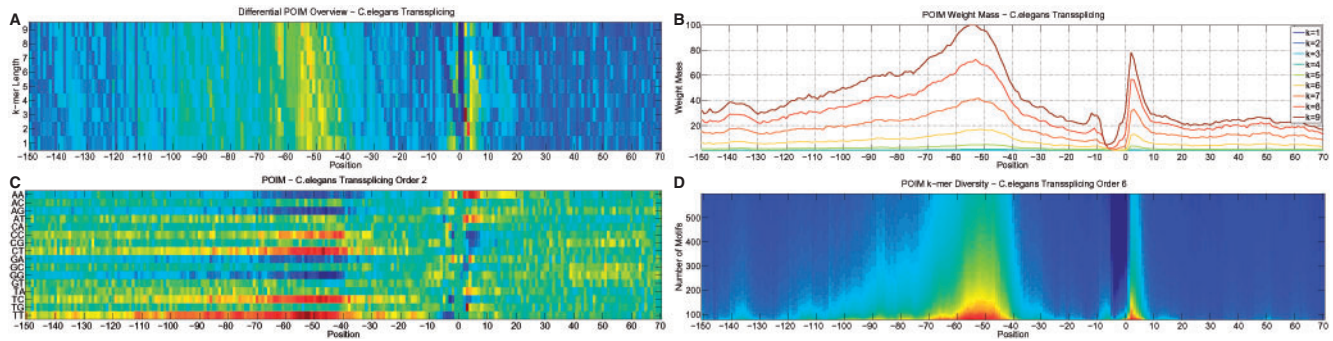


Fig. 10. POIM visualization for the *trans*-splicing control elements of *C. elegans*. (A, B) (A) displays Differential POIMs and the (B) the POIM weight mass for $k = 1 \dots 8$. (C, D) POIMs for $k = 2$ and POIM k -mer diversity.

GTATATA etc., with a peak score at -29 (see Supplemental Files at <http://www.fml.tuebingen.mpg.de/raetsch/projects/POIM>).

Central: one finds the motif CAGT at positions -1 to $+3$ (and its repeated version CAGTCAGT as a high scoring 8mer); as the next highest scoring in the 8mer POIM at -15 to $+9$ TCAGTTGT at -1 , CGTCAGTT at -3 , GTCAGTT at -2 , GGTCAGTT at -3 and AGTCAGTT at 0 are detected. These all match the known signal TCAGTT from the initiator (Inr) element consensus $TCA \frac{G}{T} T \frac{T}{C}$ (cf. Arkhipova, 1995; Burke and Kadonaga, 1997).

Downstream: many CG-rich motifs score high for 2-, 4-, 6-, 8mers and peak at $\approx +23$ nt, which is consistent with CpG-overrepresentation downstream of the TSS. At positions $+15$ to $+22$ nt, the oligomer CGTCGCG and its mostly GC-rich variations score highest while ATATTAT (and AT-dominated variants) in this region have a negative effect. A specific search for the downstream promoter element (DPE) $\frac{A}{G} \frac{G}{C} \frac{T}{T} \frac{C}{G} T G$ —as reported in Arkhipova (1995) and Burke and Kadonaga (1997)—in the POIM 7mer rankings finds AGACGTG ranked 1024 (top 8%) with an importance weight half of the highest scoring CGTCGCG. This shows that the DPE is hidden by the C/G-richness as dominating effect. Clustering of POs may help to identify such occluded motifs.

3.2.3 Detection of *trans*-splicing control elements We finally apply our methods to the *trans*-splice detector as used in the mGene gene finder (Schweikert *et al.*, manuscript in preparation). The data set has been constructed using the Wormbase WS170 annotation and by mapping annotated TRSSs to known, i.e. EST/cDNA confirmed genes (if possible). The remaining TRSSs were used as positive examples. Negative sites were generated from the remaining successfully mapped interior acceptor sites, leading to a total of 69 808 sequences (4970 positive, 64 838 negative) sites. We used a window -190 to $+70$ nt around the splice site and train an SVM classifier ($K = 30$) on 60% of the data. On the validation set this classifier achieves auROC of 95.2%, indicating that TRSSs can be distinguished well from other splice site acceptors. Figure 10 displays the POIMs computed based on this classifier.

Upstream: in the region -150 nt to -71 nt TTTT is the highest scoring 4mer, peaking at -78 nt. We then extract the top 40 scoring 8mers in the -70 nt to -21 nt window and find that they all contain one to three Cs within a poly-T motif. Finally, for comparison of our findings using POIMs with recent work by Graber *et al.* (2007), we extract the top N scoring 5mers separately for each of the windows $[-60, -50]$, $[-50, -40]$, $[-40, -30]$, $[-30, -20]$, $[-10, 0]$,

$[0, 10]$; we then verify how many of the motifs we determined are identical to the ones found in these in Graber *et al.* (2007). In Graber *et al.* (2007), two classes of TRSS leader sequences were separately analyzed: the SL1 class (*trans*-spliced to individual or operon-contained genes) and the SL2 class (nearly exclusively found attached to the downstream genes in operons). We do not split the data into two sets, and therefore compute the overlap for both parts (Graber *et al.*, 2007); part 3 – SL1, part 4 – SL2, table 2). For $N = 40$, the agreement is 15 of 25 and 18 of 25 in SL1 and SL2, respectively. For $N = 10$ it is 9 of 25 and 14 of 25. The agreeing motifs are AAATG, AGAAT, AGATG, CTTTT, GAATG, GATGA, GATGG, GATGT, TAAA, TCTTT, TTCTC, TTCTT, TTTCC, TTTCT, TTTC and TTTTT (made unique). Please note that such large overlaps are very unlikely to happen by chance. The remaining differences can be attributed to the fact that in Graber *et al.* (2007) SL1 and SL2 were separately compared to vanilla acceptor sites, whereas we do compare SL1 + SL2 with acceptor sites simultaneously, and thus we expect weaker (overlapping) motifs to be harder to detect. And indeed, when looking at the $[-60, -50]$ region, we find 7/8 of the SL1 motifs but only 3/8 of the motifs in the SL2 category; similarly we find 5/5 motifs in the $[0, 10]$ interval for SL2 but only 4/8 for SL1. This suggests that one motif class is more dominant in a certain region.

Central: one can clearly observe the high importance of positions around the TRSS and the region ~ -50 nt upstream (UR). The POIM plots of order 2 display the dominance for TT, CT and TCs in the UR, and the dominance of AAs around the TRSS. By extracting 3mers around the TRSS, we find the high ranking ATG start codon. This finding again perfectly matches the prior reported fact that the distance between TRSSs and the TIS is often very short (often 0 nt) (Graber *et al.*, 2007).

4 DISCUSSION

Modern kernel methods with complex, oligomer-based sequence kernels are very powerful for biological sequence classification. However, until now no satisfactory tool for visualization was available that helps to understand their complex decision surfaces. In this work we close this gap by introducing a method which efficiently computes the *importance* of POs defined as their expected contribution to the score. Different from the discrimination normal vector w , the importance takes into account the correlation structure of all features. We illustrated on simulated data how the

visualization of POIMs can help to identify even vaguely localized motifs where pure sequence logos will fail. On three genomic signal detection applications we demonstrated that many previously known regulatory patterns can be recovered with POIM-based ranking and visualization tools.

Note that the structure of the feature spaces of the considered *string kernels* allows us to accurately learn complex *local* correlations as opposed to long range correlations which are for instance modeled in Bayesian networks (Barash et al., 2003; Ben-Gal et al., 2005; Chen et al., 2005). It therefore seems surprising that Bayesian networks fail to achieve state-of-the-art results for tasks like the splice site recognition (Chen et al., 2005; Sonnenburg et al., 2007b). This suggests that for such tasks explicit modeling of long range relationships is not mandatory. Also, many other motif discovery algorithms and discrimination methods have been proposed before, but they typically come at the price of decreased classification accuracy as compared to SVMs with exhaustive kernels. It therefore seems promising to thoroughly analyze the SVM's decision boundary to understand the nature of the differences between the classes. Different from our previous MKL approach (Rätsch et al., 2006), we propose here to leave the SVM untouched for classification to retain its high accuracy, and to defer motif extraction to subsequent steps. This way motif finding can take advantage of the SVMs power: it ensures that the truly most important POs are identified.

Even though we have shown the usefulness of POs and methods of their visualization, for instance our analysis of promoter regulatory elements for *D. melanogaster* reveals several obstacles to systematic motif discovery with POIMs. First, we realize that the list of POs can be prohibitively large for manual inspection. Hence, if several discriminative motifs occur in the same sequence region, we currently only find the strongest ones. Second, if a motif \mathbf{z} has positional variance exceeding its length, we also find its duplicate \mathbf{zz} . This effect gets even stronger for self-overlapping motifs (like AAA, where appending a single further A already yields a second motif occurrence). We currently work on clustering approaches to summarize POs scoring high in similar regions to obtain motifs like PWMs with positional preferences. We believe that such approaches will solve both problems.

Additional work in progress includes computational techniques to efficiently determine the highest scoring 'consensus' sequence $\mathbf{x}^* := \arg\max_{\mathbf{x}} s(\mathbf{x})$, which can be of interest for sequence design applications. Finally, note that protein sequences are known to show dependencies of XOR type, e.g. a pair of a positively and a negatively charged amino acid that can swap their positions. As such dependencies escape sequence logos, but can be modeled by POIMs, an implementation of protein POIMs seems desirable.

We believe that our new POIM-based ranking and visualization algorithms are easy to use yet very helpful analysis tools. It is freely available as part of the SHOGUN toolbox (Sonnenburg et al., 2006a) at <http://www.shogun-toolbox.org>. Finally, note that it seems possible to transfer the underlying concept to other kernels and feature spaces to aid understanding of SVMs that are used for other biological tasks.

ACKNOWLEDGEMENTS

The authors would like to thank G. Schweikert for providing the trans-splice data set for *C. elegans*.

Funding: This work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence (IST-2002-506778), and by the Learning and Inference Platform of the Max Planck and Fraunhofer Societies.

Conflict of Interest: none declared.

REFERENCES

- Arkhipova, I.R. (1995) Promoter elements in *D. melanogaster* revealed by sequence analysis. *Genetics*, **139**, 1359–1369.
- Barash, Y. et al. (2003) Modeling depend. in protein-DNA binding sites. In *Proceedings of the 7th International Conference in Computational Molecular Biology (RECOMB)*.
- Ben-Gal, I. et al. (2005) Identification of transcription factor binding sites with variable-order bayesian networks. *Bioinformatics*, **21**, 2657–2666.
- Burke, T. and Kadonaga, T. (1997) The downstream core promoter element, DPE, is conserved from *Drosophila* to humans and is recognized by TAFII60 of *Drosophila*. *Genes Dev.*, **11**, 3020–3031.
- Chen, T.-M. et al. (2005) Prediction of splice sites with dependency graphs and their expanded bayesian networks. *Bioinformatics*, **21**, 471–482.
- Down, T. and Hubbard, T. (2002) Computational detection and location of transcription start sites in mammalian genomic DNA. *Genome Res.*, **12**, 458–461.
- Graber, J. et al. (2007) *C. elegans* sequences that control trans-splicing and operon pre-mRNA processing. *RNA*, **13**, 1–18.
- Graf, A. et al. (2006) Classification of faces in man and machine. *Neural Comput.*, **18**, 143–165.
- Harris, N.L. and Senapathy, P. (1990) Distribution and consensus of branch point signals in eukaryotic genes: a computerized statistical analysis. *Nucleic Acids Res.*, **18**, 3015–3019.
- Lanckriet, G.R.G. et al. (2004) Learning the kernel matrix with semidefinite programming. *J. Mach. Learn. Res.*, **5**, 27–72.
- Leslie, C. et al. (2002) The spectrum kernel: a string kernel for SVM protein classification. In Altman, R. et al. (eds) *Proceedings of the PSB*. World Scientific, River Edge.
- Leslie, C. et al. (2003) Mismatch string kernels for SVM protein classification. In Becker, S.T.S. and Obermayer, K. (eds) *Advances in Neural Information Processing System 15*. MIT Press, Cambridge, pp. 1417–1424.
- Meinicke, P. et al. (2004) Oligo kernels for datamining on biological sequences: a case study on prokaryotic translation initiation sites. *BMC Bioinf.*, **5**, 169.
- Ohler, U. (2006) Identification of core promoter modules in *Drosophila* and their application in accurate transcription start site prediction. *Nucleic Acids Res.*, **34**, 5943–5950.
- Ohler, U. et al. (2002) Computational analysis of core promoters in the *drosophila* genome. *Genome Biol.*, **3**.
- Rätsch, G. and Sonnenburg, S. (2004) Accurate splice site detection for *C. elegans*. In Schölkopf, B. et al. (eds) *Kernel Methods in Computational Biology*. MIT Press, pp. 277–298.
- Rätsch, G. et al. (2005) RASE: recognition of alternatively spliced exons in *C. elegans*. *Bioinformatics*, **21**(Suppl. 1), i369–i377.
- Rätsch, G. et al. (2006) Learning interpretable SVMs for biological sequence classification. *BMC Bioinformatics*, **7**(Suppl. 1), S9.
- Saeyns, Y. et al. (2007) Translation initiation site prediction on a genomic scale: beauty in simplicity. *Bioinformatics*, **23**, i418–i423.
- Schölkopf, B. and Smola, A.J. (2002) *Learning with Kernels*. MIT Press, Cambridge.
- Sonnenburg, S. et al. (2005) Learning interpretable SVMs for biological sequence classification. In Miyano, S. et al. (eds), *RECOMB 2005, LNBI 3500*. Springer-Verlag Berlin Heidelberg, pp. 389–407.
- Sonnenburg, S. et al. (2006a) Large scale multiple kernel learning. *J. Mach. Learn. Res.*, **7**, 1531–1565.
- Sonnenburg, S. et al. (2006b) ARTS: accurate recognition of transcription starts in human. *Bioinformatics*, **22**, e472–e480.
- Sonnenburg, S. et al. (2007a) Large scale learning with string kernels. In Bottou, L. et al. (eds), *Large Scale Kernel Machines*. MIT Press, Cambridge, MA, pp. 73–103.
- Sonnenburg, S. et al. (2007b) Accurate splice site prediction. *BMC Bioinformatics Special Issue from NIPS Workshop on New Problems and Methods in Computational Biology, Whistler, Canada, December 18, 2006*, **8** (Suppl. 10), S7.
- Üstün, B. et al. (2007) Visualisation and interpretation of support vector regression models. *Anal. Chim. Acta*, **595**, 299–309.
- Vapnik, V.N. (1995) *The Nature of Statistical Learning Theory*. Springer.
- Zien, A. et al. (2007) Computing positional oligomer importance matrices (POIMs). Research Report, Electronic Publishing 2, Fraunhofer FIRST.