Large Scale Machine Learning for Genomic Sequence Analysis

(Support Vector Machine Based Signal Detectors)

Sören Sonnenburg

Friedrich Miescher Laboratory, Tübingen

joint work with

Alexander Zien, Jonas Behr, Gabriele Schweikert, Petra Philips and Gunnar Rätsch

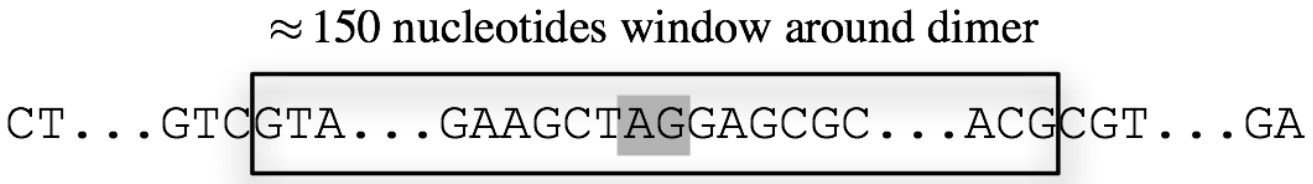
Friedrich Miescher Laboratory

of the Max Planck Society

Outline

Recognizing Genomic Signals

Discriminate true signal positions against all other positions



- True sites: fixed window around a true site
- Decoy sites: all other consensus sites

```
AAACAAATAAGTAACTAATCTTTTAGGAAGAACGTTTCAACCATTTTGAG
AAGATTAAAAAAAAACAAATTTTTAGCATTACAGATATAATAATCTAATT
CACTCCCCAAATCAACGATATTTTAGTTCACTAACACATCCGTCTGTGCC
TTAATTTCACTTCCACATACTTCCAGATCATCAATCTCCAAAACCAACAC
```

Examples: Transcription start site finding, splice site prediction, alternative splicing prediction, trans-splicing, polyA signal detection, translation initiation site detection

Types of Signal Detection Problems I

Vague categorization

(based on positional variability of motifs)

Position Independent

→ Motifs may occur anywhere,

```
AAACAAAAACGTAACTAATCTTTTAGAGAGAACGTTTCAACCATTTTGAG
AAGATTAACTCATCACAGATTTCATTACATACAGATATAATTCAAAAATT
CACTCCCCAAATCAACGATATTTAAAAATCACTAACACATCCGTCTGTGC
```

e.g. tissue classification using promotor region

Types of Signal Detection Problems II

Vague categorization

(based on positional variability of motifs)

Position Dependent

→ Motifs very stiff, almost always at same position,

e.g. Splice Site Classification

Types of Signal Detection Problems III

Vague categorization

(based on positional variability of motifs)

Mixture Position Dependent/Independent

→ variable but still positional information

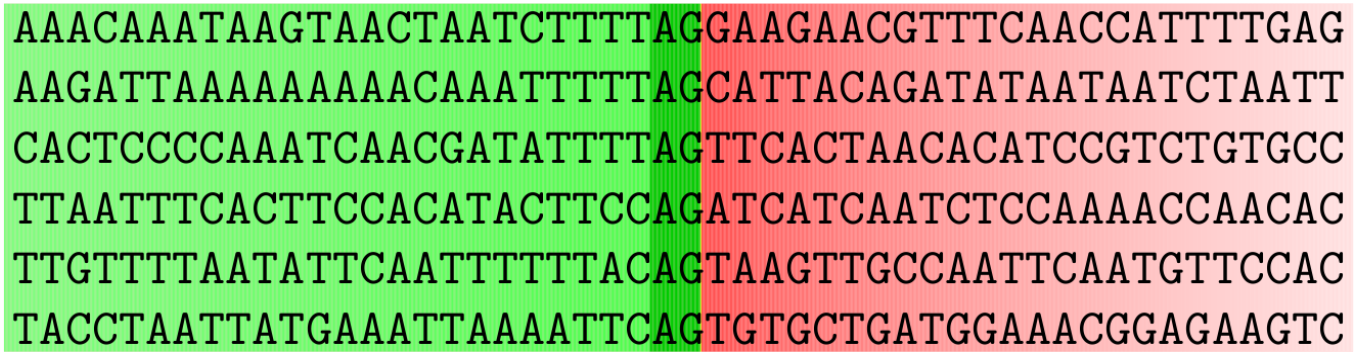
```
AAACAAATAAGTAACTAATCTTTTAAAGAGAACGTTTCAACCATTTTGAG
AAGATTAAAAAAAAACAAATTTCATTAAATACAGATATAATAATCTAATT
CACTCCCCAAATCAACGATATTTAAATTTCACTAACACATCCGTCTGTGC
```

e.g. Promoter Classification

Classification - Learning based on examples

Given:

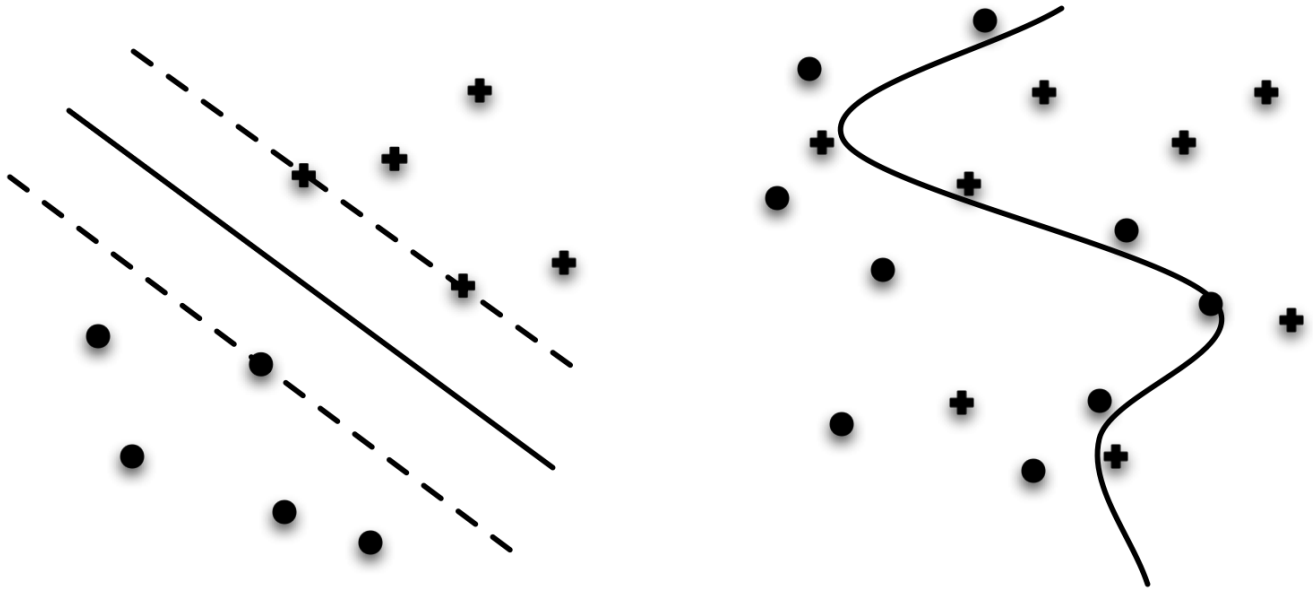
Training examples $(\mathbf{x}_i, y_i)_{i=1}^N \in (\{A, C, G, T\}^L, \{-1, +1\})^N$



Wanted:

Function (Classifier) $f(\mathbf{x}) : \{A, C, G, T\}^L \mapsto \{-1, +1\}$

Support Vector Machines (SVMs)



- **Support Vector Machines** learn weights $\alpha \in \mathbb{R}^N$ over training examples in kernel feature space $\Phi : \mathbf{x} \mapsto \mathbb{R}^D$,

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

with kernel $k(\mathbf{x}, \mathbf{x}') = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}')$

The Spectrum Kernel

Support Vector Machine

$$f(\mathbf{x}) = \text{sign}\left(\sum_{i=1}^{N} y_i \alpha_i \mathsf{k}(\mathbf{x}, \mathbf{x}_i) + b\right),$$

Spectrum Kernel (with mismatches, gaps)

$$K(\mathbf{x}, \mathbf{x}') = \Phi_{sp}(\mathbf{x}) \cdot \Phi_{sp}(\mathbf{x}')$$

AAACAAAAACGTAACTAATCTTTTAGAGAGAACGTTTCAACCATTTTGAG
AAGATTAACTCATCACAGATTTCATTACATACAGATATAATTCAAAAATT
CACTCCCCAAATCAACGATATTTAAAAATCACTAACACATCCGTCTGTGC

The Weighted Degree Kernel

Support Vector Machine

$$f(\mathbf{x}) = \text{sign}\left(\sum_{i=1}^{N} y_i \alpha_i \mathsf{k}(\mathbf{x}, \mathbf{x}_i) + b\right),$$

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

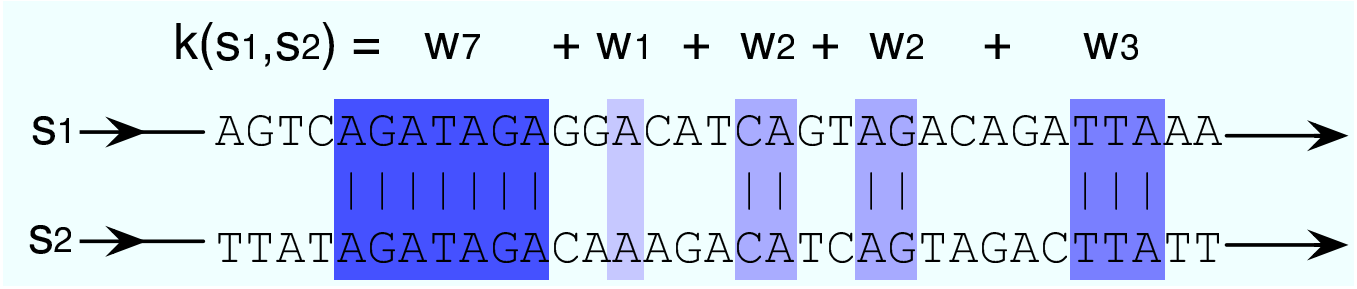
```
      x  AAACAAATAAGTAACTAATCTTTTAGGAAGAACGTTTCAACCATTTTGAG
#1-mers  .|.|.|||.|..||.|.|..|||.||...|....|...|||......|..
#2-mers  .....||.....|.......||..|............||.........
#3-mers  .....|.............|................|..........
      y  TACCTAATTATGAAATTAAATTTCAGTGTGCTGATGGAAACGGAGAAGTC
```

Example: $K = 3$: $\mathsf{k}(\mathbf{x}, \mathbf{x}') = \beta_1 \cdot 21 + \beta_2 \cdot 8 + \beta_3 \cdot 3$

The Weighted Degree Kernel with *shifts*

Support Vector Machine

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



Accelerating String-Kernel-SVMs

1. Linear run-time of the kernel
2. Accelerating linear combinations of kernels

Idea of the `Linadd` Algorithm:

Store $\mathbf{w}$ and compute $\mathbf{w} \cdot \Phi(\mathbf{x})$ *efficiently*

$$f(\mathbf{x}_j) = \sum_{i=1}^{N} \alpha_i y_i \, \mathrm{k}(\mathbf{x}_i, \mathbf{x}_j) = \underbrace{\sum_{i=1}^{N} \alpha_i y_i \Phi(\mathbf{x}_i)}_{\mathbf{w}} \cdot \Phi(\mathbf{x}_j) = \mathbf{w} \cdot \Phi(\mathbf{x}_j)$$

Possible for low-dimensional or sparse $\mathbf{w}$

Effort: $\mathcal{O}(NL) \Rightarrow$ speedup of factor N

$\Rightarrow$ Training on millions of examples, evaluation on billions.

Accelerating String-Kernel-SVMs II

Recent work:

Further drastic speedup using advances of primal SVMs solvers

Acceleration using fast primal SVMs

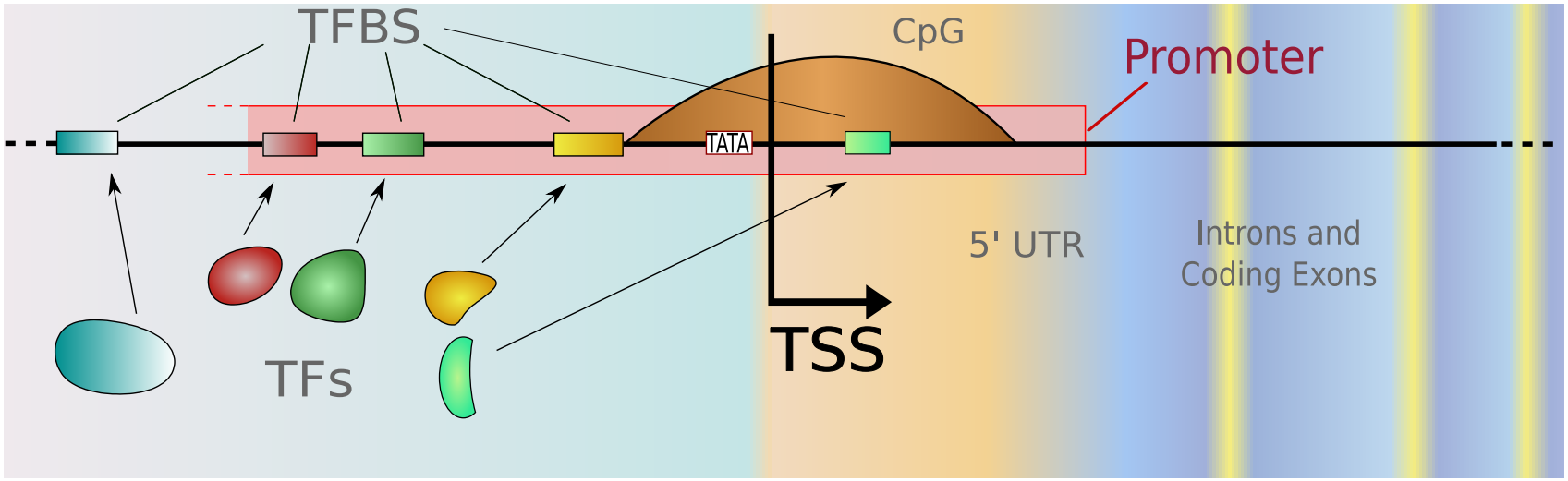
- Idea: Train SVM in primal using kernel feature space
- Problem: $>$ 12 million dims; 50 million examples
- Only $\mathbf{w} \leftarrow \mathbf{w} + \alpha\Phi(\mathbf{x})$ and $\mathbf{w} \cdot \Phi(\mathbf{x})$ required.
- Compute $\Phi(\mathbf{x})$ on-the-fly and parallelize!

Results

$\Rightarrow$ Computations are simple "table lookups" of $k-$mers weights

$\Rightarrow$ **Allows training on 50 million examples**

Detecting Transcription Start Sites



- POL II indirectly binds to a rather vague region of $\approx [-20, +20]$ bp
- Upstream of TSS: promoter containing transcription factor binding sites
- Downstream of TSS: 5' UTR, and further downstream coding regions and introns (different statistics)
- 3D structure of the promoter must allow the transcription factors to bind

Several weak features $\Rightarrow$ Promoter prediction is non-trivial

Features to describe the TSS

- TFBS in Promoter region
- condition: DNA should not be too twisted
- CpG islands (often over TSS/first exon; in most, but not all promoters)
- TSS with TATA box (≈ -30 bp upstream)
- Exon content in UTR 5" region
- Distance to first donor splice site

Idea:

Combine weak features to build strong promoter predictor

$$k(\mathbf{x}, \mathbf{x}') = k_{TSS}(\mathbf{x}, \mathbf{x}') + k_{CpG}(\mathbf{x}, \mathbf{x}') + k_{coding}(\mathbf{x}, \mathbf{x}') + k_{energy}(\mathbf{x}, \mathbf{x}') + k_{twist}(\mathbf{x}, \mathbf{x}')$$

The 5 sub-kernels

1. TSS signal (including parts of core promoter with TATA box)

   – use **Weighted Degree Shift kernel**

2. CpG Islands, distant enhancers and TFBS upstream of TSS

   – use **Spectrum kernel** (large window upstream of TSS)

3. Model coding sequence TFBS downstream of TSS

   – use another **Spectrum kernel** (small window downstream of TSS)

4. Stacking energy of DNA

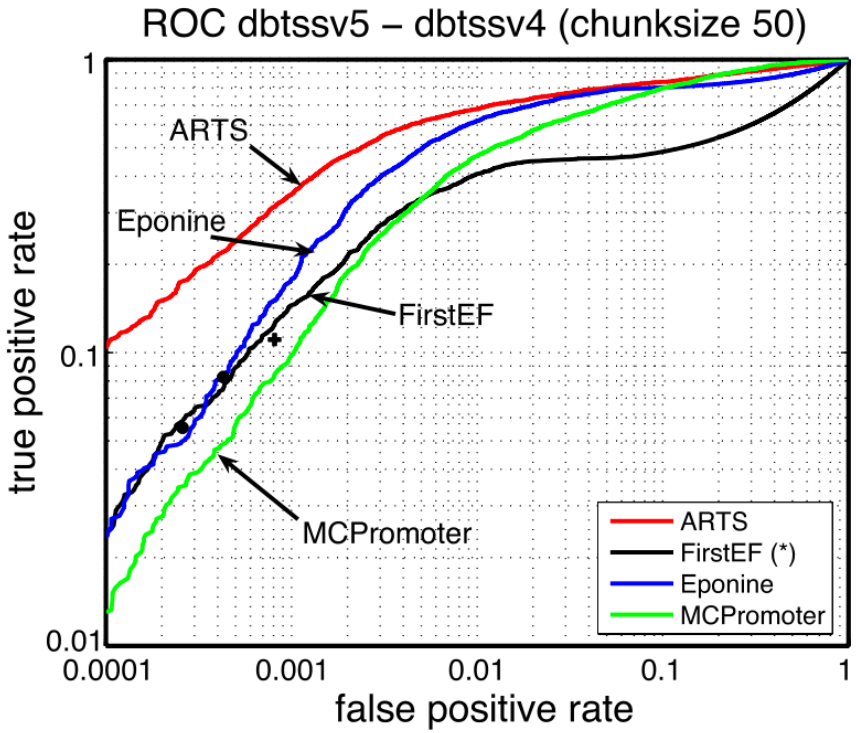
   – use *btwist* energy of dinucleotides with **Linear kernel**

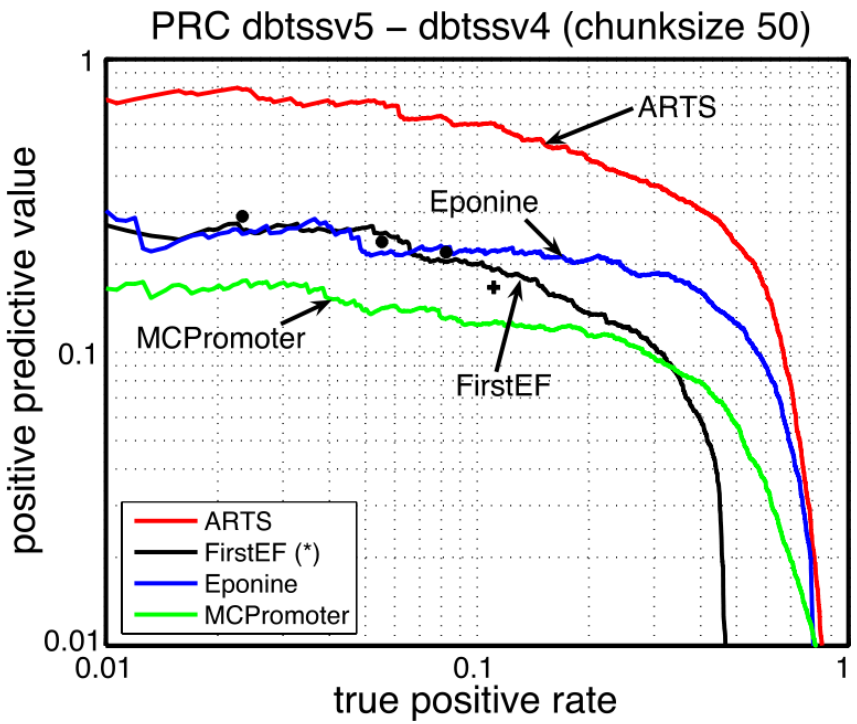
5. Twistedness of DNA

   – use *btwist* angle of dinucleotides with **Linear kernel**

State-of-the-art Performance

Receiver Operator Characteristic Curve and Precision Recall Curve

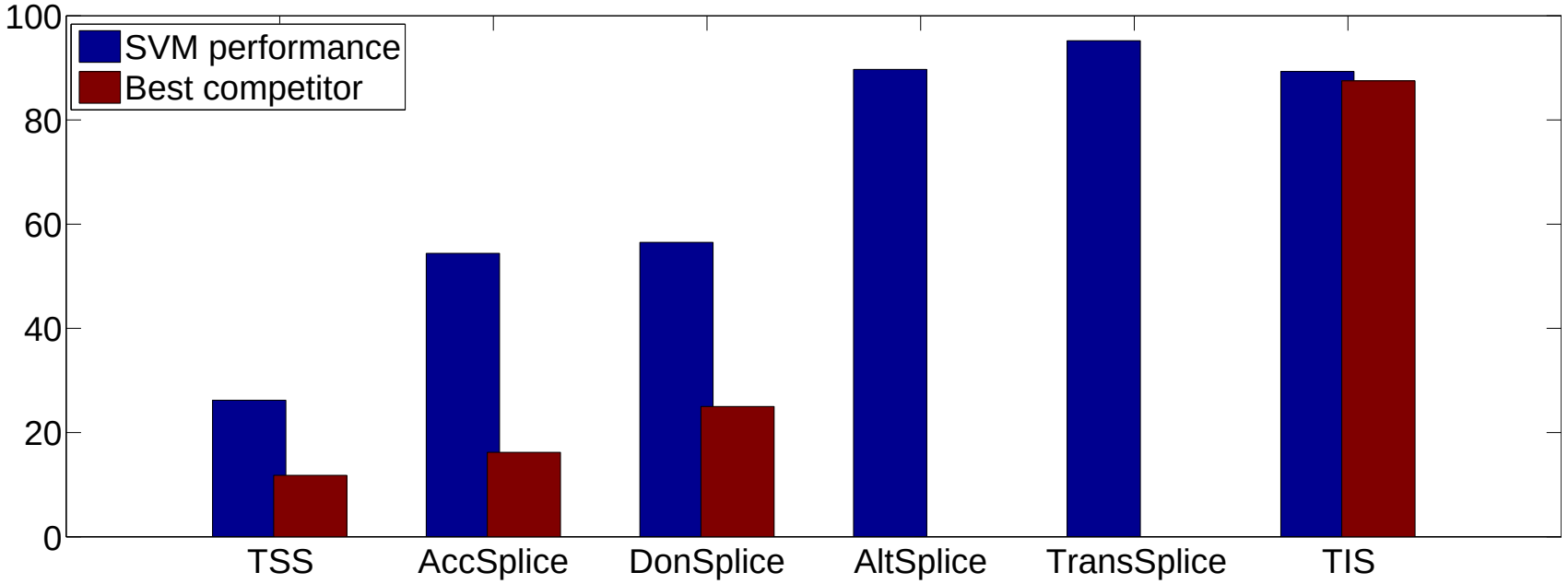




$\Rightarrow$ 35% **true positives at a false positive rate of** $1/1000$
(best other method find about a half (18%**))**

Friedrich Miescher Laboratory
of the Max Planck Society

Beauty in Generality

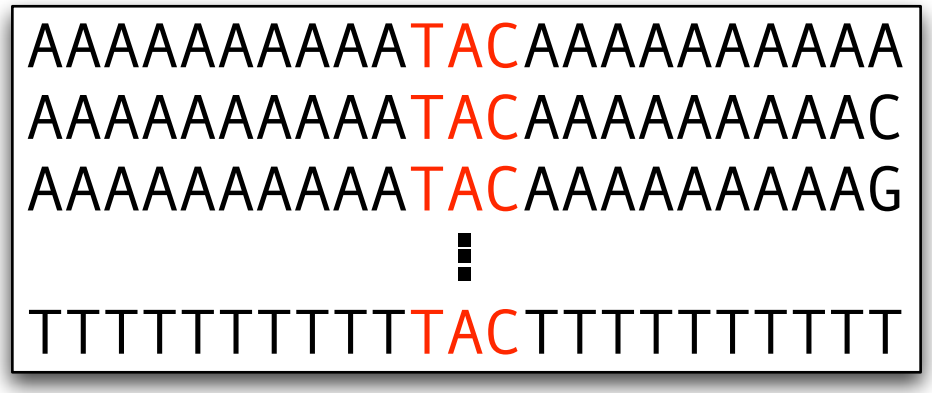


- Transcription Start (Sonnenburg et al., Eponine Down et al.)
- Acceptor Splice Site (Schweikert et al.)
- Donor Splice Site (Schweikert et al.)
- Alternative Splicing (Rätsch et al., -)
- Transsplicing (Schweikert et al., -)
- Translation Initiation (Sonnenburg et al., Saeys et al.)

Positional Oligomer Importance Matrices (POIMs)

Determine importance of $k-$mers at one glance:

- Given $k-$mer $\mathbf{z}$ at position j in the sequence, compute expected score $\mathbb{E}\left[\, s(\mathbf{x}) \mid \mathbf{x}[j] = \mathbf{z} \,\right]$ (**for small k**)



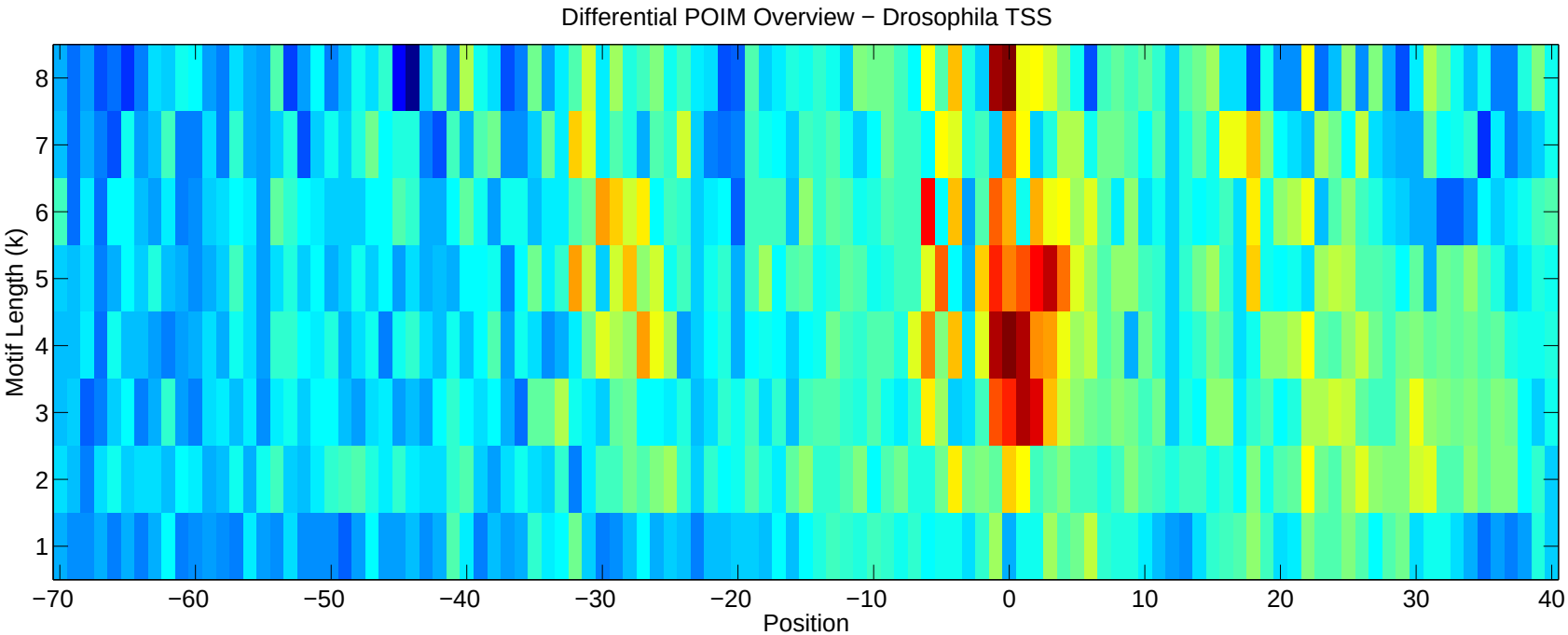
- Normalize with *expected score* over **all sequences**

POIMs

$$Q(\mathbf{z}, j) \;:=\; \mathbb{E}\left[\, s(\mathbf{x}) \mid \mathbf{x}[j] = \mathbf{z} \,\right] - \mathbb{E}\left[\, s(\mathbf{x}) \,\right]$$

Interpretable via Positional Oligomer Importance Matrices

Example: Drosophila Transcription Starts



TATAAAA	-29/++	**CAGT**CAGT	-01/++	CGTCGCG	+18/++
GTATAAA	-30/++	T**CAGT**TGT	-01/++	GCGCGCG	+23/++
ATATAAA	-28/++	CGT**CAGT**T	-03/++	CGCGCGC	+22/++
TATA-box		**Inr** **TCA**$\frac{G}{T}$**T**$\frac{T}{C}$		**CpG**	

Friedrich Miescher Laboratory
of the Max Planck Society

Conclusions

Support Vector Machines with string kernels

- General
- **Fast:** Applicable to genome-sized datasets
- Often are state-of-the art signal detectors
  - TSS
  - Acceptor and Acceptor Splice Site
  - ...
- Used in mGene gene finder `http://www.mgene.org`
- Positional Oligomer Importance Matrices help making SVMs interpretable

Galaxy web-interface `http://galaxy.fml.tuebingen.mpg.de`
Efficient implementation `http://www.shogun-toolbox.org`
More machine learning software `http://mloss.org`