2007 IEEE SYMPOSIUM SERIES ON COMPUTATIONAL INTELLIGENCE
April 1-5, 2007, Honolulu, Hawaii, USA
Hilton Hawaiian Village Beach Resort & Spa, Waikiki

2007 IEEE Symposium on Computational Intelligence on Computational Biology and Bioinformatics (CIBCB 2007)

Tutorial

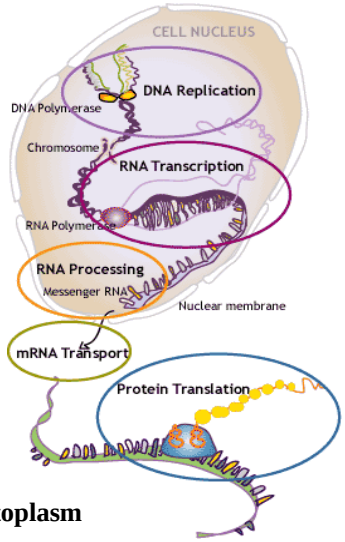
Signal and Motif Detection in Genomic Sequences

Jagath C. Rajapakse, Ph.D

BioInformatics Research Centre
Nanyang Technological University, Singapore

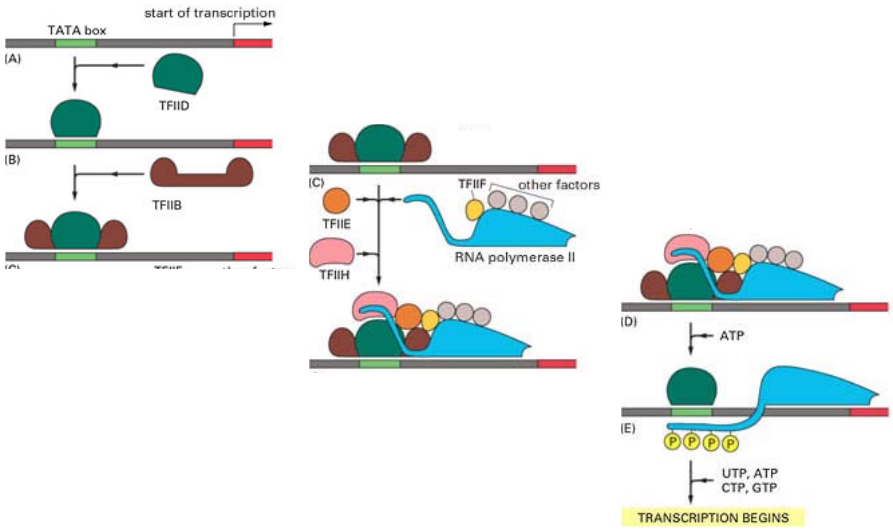
Central Dogma of Molecular Biology

DNA
↓ transcription
pre-mRNA
↓ splicing
mRNA
↓ translation
Protein

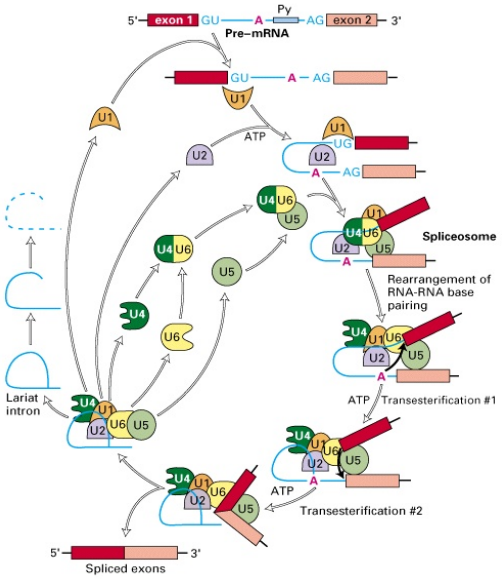


cell cytoplasm

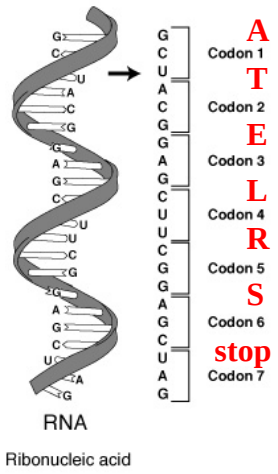
Transcription: DNA→nRNA (pre-mRNA)



Splicing: nRNA→mRNA



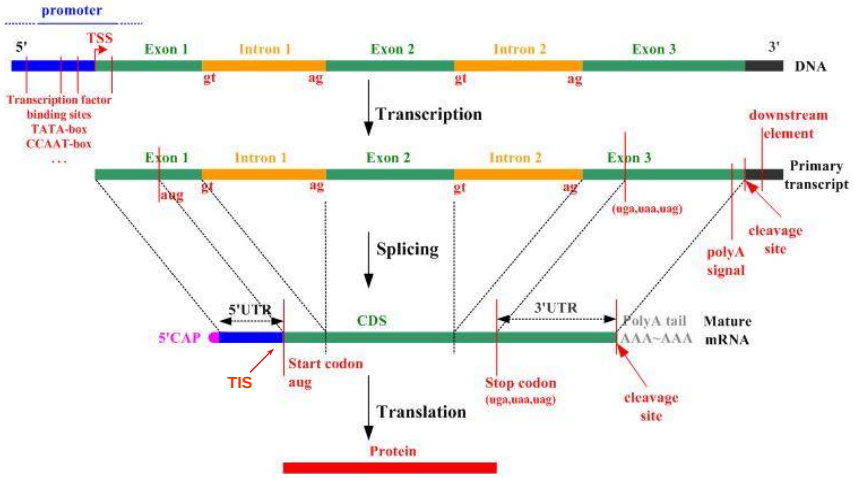
Translation: mRNA→protein



First	U	C	A	G	Last
U	Phe F	Ser S	Tyr Y	Cys C	U
	Phe	Ser	Tyr	Cys	C
	Leu L	Ser	**Stop** (Ochre)	**Stop** (Umber)	A
	Leu	Ser	**Stop** (Amber)	Trp W	G
C	Leu	Pro P	His H	Arg R	U
	Leu	Pro	His	Arg	C
	Leu	Pro	Gln Q	Arg	A
	Leu	Pro	Gln	Arg	G
A	Ile I	Thr T	Asn N	Ser	U
	Ile	Thr	Asn	Ser	C
	Ile	Thr	Lys K	Arg	A
	Met M	Thr	Lys	Arg	G
G	Val V	Ala A	Asp D	Gly G	U
	Val	Ala	Asp	Gly	C
	Val	Ala	Glu E	Gly	A
	Val	Ala	Glu	Gly	G

Stages in eukaryotic gene expression

From nobelprize.org



Signal and Motif Detection

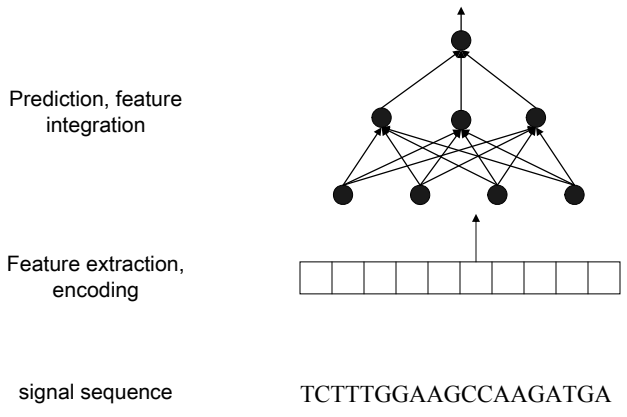
Signals

- Splice Sites (SS)
- Transcription Start Sites (TSS)
- Translation Initiation Sites (TIS)

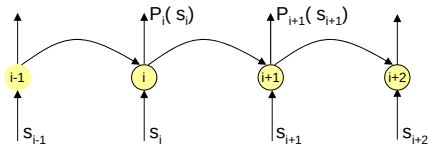
Promoters

Motifs

General Scheme of Signal Detection



Markov Chain Models



A region of DNA can be represented by Markov chain model.

In a *k*-order Markov chain, The nucleotide at site *i* depends *k* previous nucleotides.

The features extracted by a Markov chain of order *k* at site *i* is represented by a vector $P_i(s_i) = (P(s_i|s_{i-1},\ldots,s_{i-k}) : s_i \in \{A,C,G,T\})$.

Neural Networks

Consider a neural network with *n* input nodes and *m* hidden nodes.

The prediction *y* of a neural network receiving input $\mathbf{x} = (x_1, x_2, \ldots x_n)$

$$y = f\left(\sum_{k=1}^{m} w_k f_k\left(\sum_{j=1}^{n} w_{kj} x_j\right)\right)$$

w_k and w_{kj}, where $k = 1, 2, \ldots m$, and $j = 1, 2, \ldots n$ represent weights connected to the output neuron and hidden neurons, respectively.

Higher-Order Markov Models

Consider a sequence **s** = ($s_1, s_2, \ldots s_n$)

Higher-order conditional dependencies can be approximated by interpolation

$$P(s_i|s_{i-1},...,s_1) \approx \frac{\sum_{k=0}^{i-1} a_k g_k(s_{i-k},...,s_{i-1}) P(s_i|s_{i-1},...,s_{i-k})}{\sum_{k=0}^{i-1} a_k g_k(s_{i-k},...,s_{i-1})}$$

where a_k are real coefficients such that $\sum_{k=1}^{i-1} a_k = 1.0$, and $g_k(.)$ represents the relationships of different-order contextual nucleotide interactions that, for instance, can be chosen to be a sigmoid function dependent on the frequency of last *k* symbols (Ohler et al., 1999)

$$g_k(s_{i-k},...,s_{i-1}) = \frac{\#(s_{i-k},...,s_{i-1})}{\#(s_{i-k},...,s_{i-1}) + C}$$

By replacing conditional probabilities with probabilities conditioned by a less number of elements, and by using chain rule, the likelihood of the sequence is given by

$$\begin{aligned} P(s_1, s_2, \ldots s_i) &= P(s_1) \prod_{j=2}^{i} P(s_j|s_{j-1}, \ldots, s_1) \\ &\approx P(s_1) \prod_{j=2}^{i} \sum_{k=1}^{j-1} b_{kj} P(s_j|s_{j-1}, \ldots, s_{j-k}) \\ &\approx \sum_{j=1}^{i} \prod_{k=1, j-k \le 1; m_k, \ldots m_j = 0}^{k=j-1; m_k + \ldots + m_j = j} c_{m_{j-k}, \ldots, m_j} P^{d_{m_{j-k}, \ldots, m_j}}(s_j|s_{j-1}, \ldots, s_{j-k}) \end{aligned}$$

where coefficients *c* and *d* are non-negative integer coefficients.

That is, the nonlinear relationships amongst variables in the sequence is represented favorably by a polynomial of sufficient order.

Higher-Order Markov Models with Neural Networks

If the features are given to a neural network as inputs, the output

$$y = \sum_{m_1,\ldots m_l=0; m_1+\ldots m_l=l} c_{m_1,\ldots m_l} P_1(s_1)^{m_1} \ldots P_l(s_l)^{m_l}$$

approximates an higher-order polynomial of input features, determined by the number of hidden units.

Training:

Phase 1: estimate the Markov chains' parameters

Parameters of kth order Markov chain are estimated, according to

$$\widehat{P}_i(s_i) = \frac{\#(s_{i-k}^{i})}{\#(s_{i-k}^{i-1})}$$

Phase 2: train the component neural networks

- The training sequences are applied and the Markovian probabilities are used as inputs to the network.
- The neural networks are trained independently by using an online error backpropagation algorithm.

K-grams

k consecutive letters

k = 1, 2, 3, 4, 5, …

Window size vs. fixed position

Up-stream, downstream vs. any where in window

In-frame vs. any frame

For each value of *k*, there are $4^k * 3 * 2$ *k*-grams.

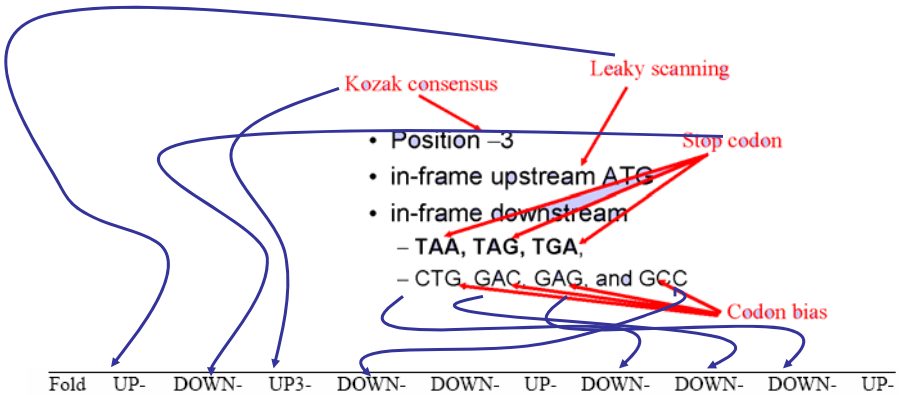
For *k* = 1, 2, 3, 4, 5, we have

4 + 24 + 96 + 384 + 1536 + 6144 = 8188 features!

This is too many for most CI algorithms.

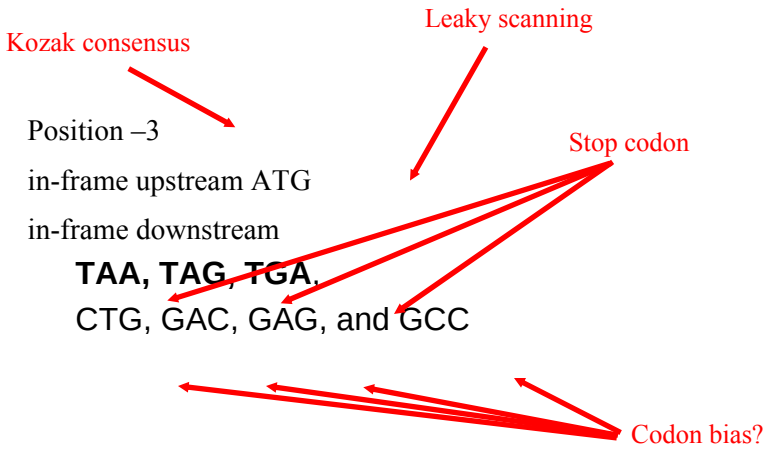
The counts of *k*-grams along the sequence should be obtained.

Amino Acid K-grams Discovered by Entropy

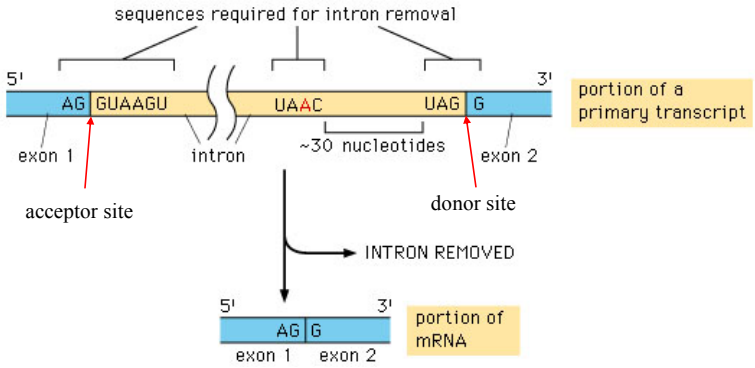


Fold	UP-ATG	DOWN-STOP	UP3-AorG	DOWN-A	DOWN-V	UP-A	DOWN-L	DOWN-D	DOWN-E	UP-G
1	1	2	4	3	6	5	8	9	7	10
2	1	2	3	4	5	6	7	8	9	10
3	1	2	3	4	5	6	8	9	7	10

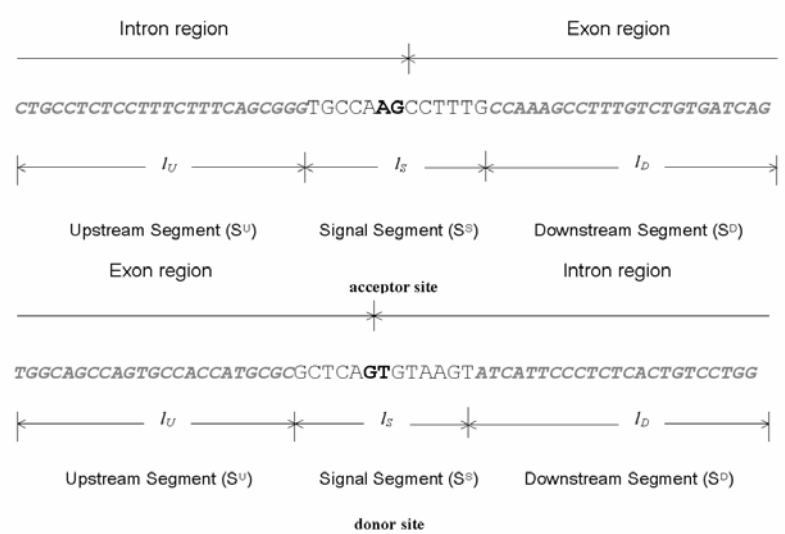
Sample *k*-grams Selected by CFS



Splice sites

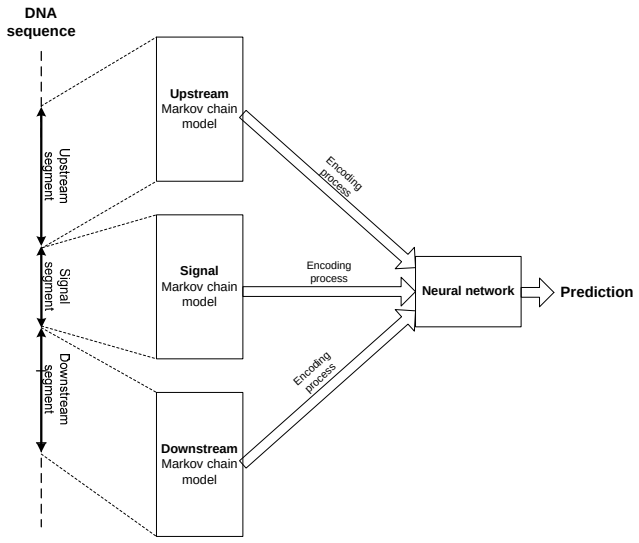


Representation of splice sites



Splice site detection with neural networks and Markov encoding

Rajapakse & Ho, 2005, *IEEE TCBB*



Markov models surrounding splice sites:

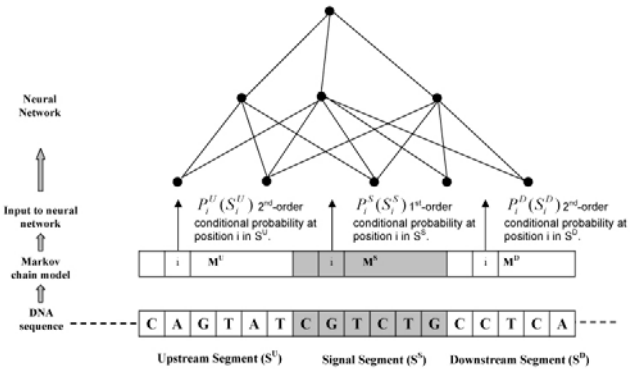
If the signal, upstream, downstream segment models are denoted by M^S, M^U, M^D, respectively, whose parameters are given by

$P_i^U(s_i) = P_i(s_i|s_{i-1}, s_{i-1}, M^U)$, second-order Markov property
$P_i^S(s_i) = P_i(s_i|s_{i-1}, M^S)$, first-order Markov property
$P_i^D(s_i) = P_i(s_i|s_{i-1}, s_{i-1}, M^D)$, second-order Markov property

where:

$M^S = \{ P_i^S(s) \,|s \in \Sigma_{\text{DNA}}, i=1,2,\ldots,l_S\}$
$M^U = \{P_i^U(s) \,|s \in \Sigma_{\text{DNA}}, i=1,2,\ldots,l_U\}$
$M^D = \{P_i^D(s) \,|s \in \Sigma_{\text{DNA}}, i=1,2,\ldots,l_D\}$

Features input to the neural network:



Incorporate the homology of potential splice sites into the neural network
Learn the compositional contrasts of coding and non-coding regions
Accounts for higher-order interactions among the nucleotides.

Accuracy measures

	predicted as positive	predicted as negative
positive	TP	FN
negative	FP	TN

$$\text{Sensitivity wrt positives} = \frac{\text{No. of correct positive predictions}}{\text{No. of positives}} = \frac{TP}{TP + FN}$$

$$\text{Specificity wrt positives} = \frac{\text{No. of correct negative predictions}}{\text{No. of negatives}} = \frac{TN}{TN + FP}$$

Correlation Coefficient

$$CC = \frac{(TP \times TN) - (FN \times FP)}{\sqrt{(TP + FN) \times (TN + FP) \times (TP + FP) \times (TN + FN)}}$$

Experiments

NN269 dataset (Reese et al., 1997)

269 human genes

Training set:

1116 true acceptor sites and 1116 true donor sites

4672 false acceptor sites and 4140 false donor sites

Testing set:

208 true acceptor sites and 208 true donor sites

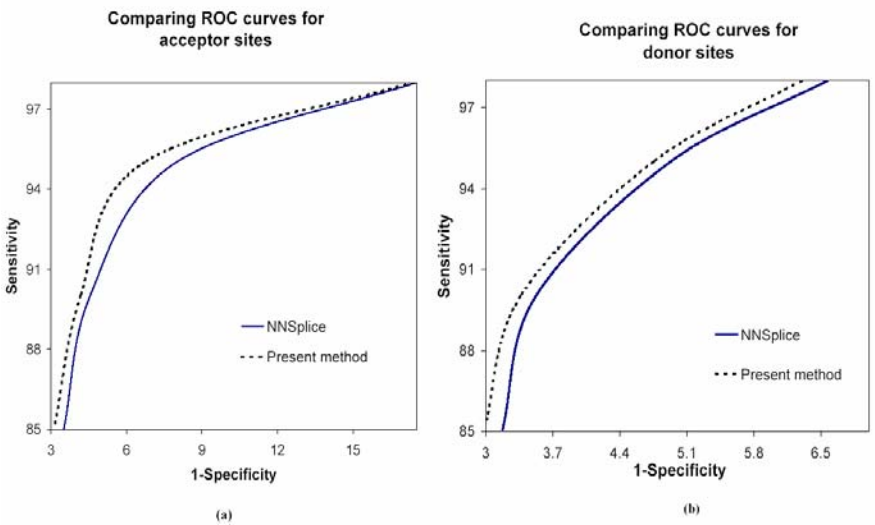
881 false acceptor sites and 782 false donor sites

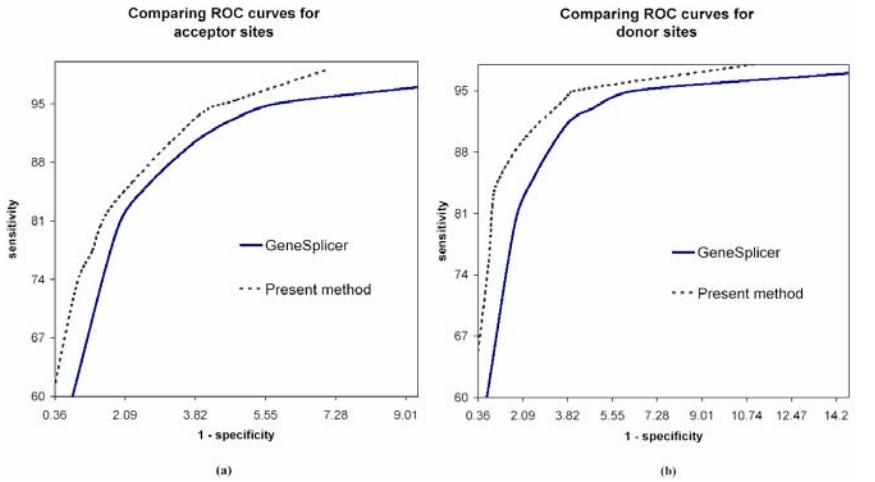
GS1115 dataset (Pertea et al., 2001)

1115 human genes

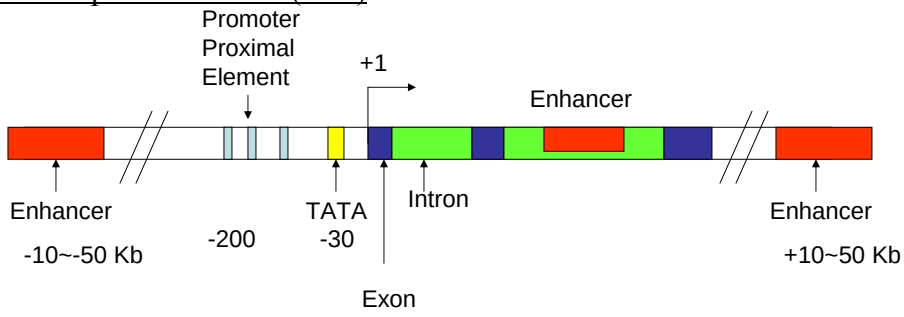
5733 true acceptor sites and 5733 true donor sites

650099 false acceptor sites and 478983 false donor sites

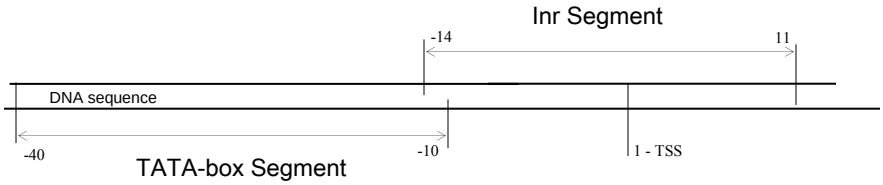




Transcription Start Site (TSS)



Representation and biological properties of TSS

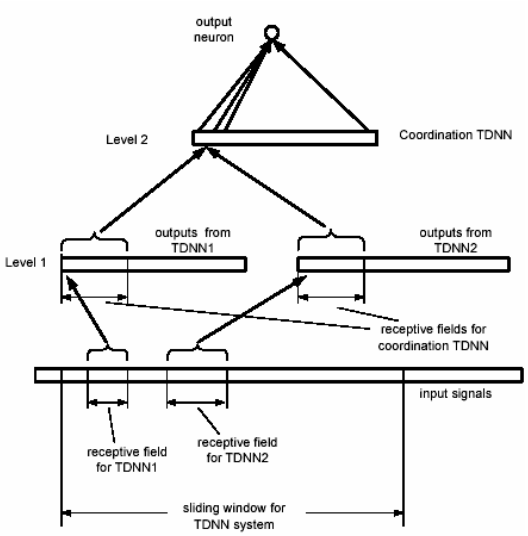


TATA-box, a binding site usually found at –25 bp upstream of TSSs, is the most conserved sequence motif.

Inr is a weaker signal than TATA-box

TSS signals are more complex than splice site signals.

NNPP (TSS Recognition)

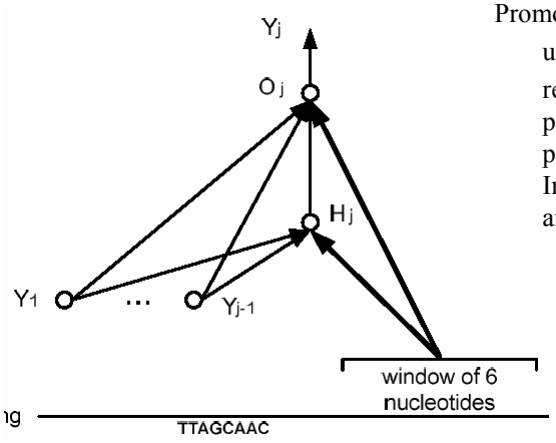


NNPP2.1

- use 3 time-delayed ANNs
- recognize TATA-box, Initiator, and their mutual distance

Makes about 1 prediction per 550 nt at 0.75 sensitivity

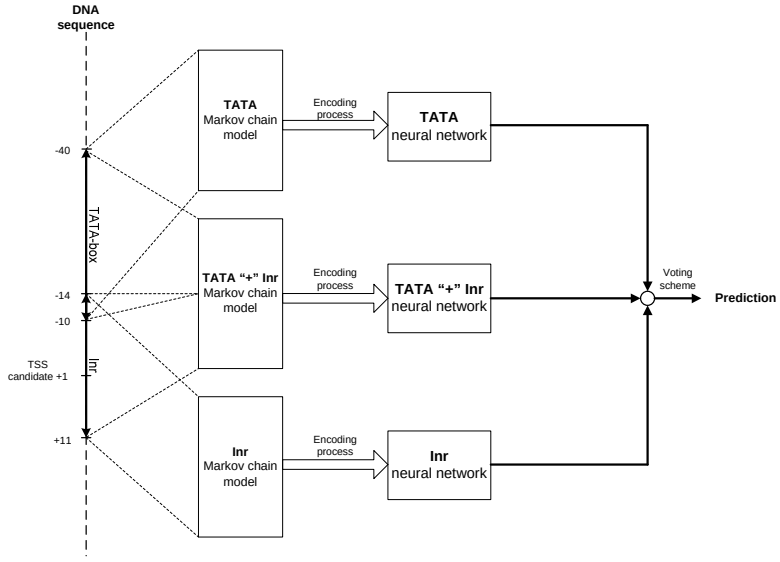
Promoter 2.0 (TSS Recognition)



Promoter 2.0

- use ANN
- recognize 4 signals commonly present in eukaryotic promoters: TATA-box, Initiator, GC-box, CCAAT-box, and their mutual distances

TSS detection with Neural Networks and Markov Encoding



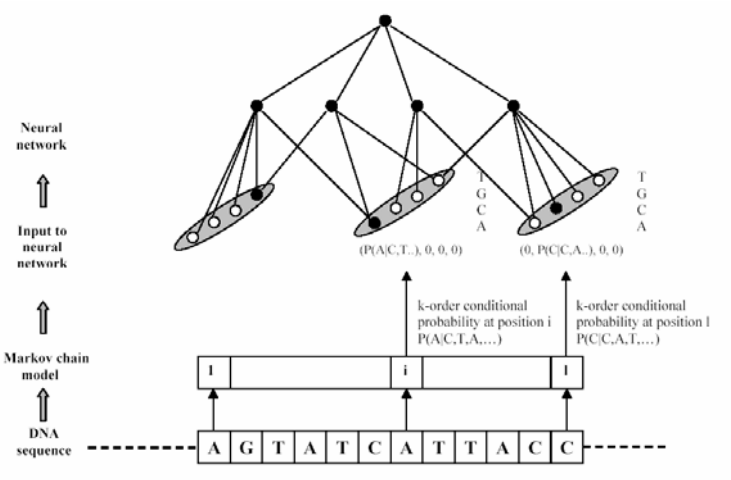
Enhanced Markov Encoding

In the orthogonal encoding method, a nucleotide s is encoded by a block of four digits bs, e.g. $b_A = (1,0,0,0)$, $b_C = (0,1,0,0)$, $b_G = (0,0,1,0)$, $b_T = (0,0,0,1)$.

The Markov encoding model combines the outputs from Markov chain model and the orthogonal encoding, e.g.:

1. Input sequence ($s_1, \ldots, s_l$)
2. Calculate the outputs of the Markov chain model $(P_1(s_1), P_2(s_2),\ldots, P_l(s_l))$
3. Calculate 4* l units $(b_{s1},b_{s2},\ldots,b_{sl})$ using the orthogonal encoding
4. Output a vector of 4 * l units $(c_{s1},c_{s2},\ldots,c_{sl})$ where $c_{si} = P_i(s_i)b_{si}$.

Inputs to Neural Network

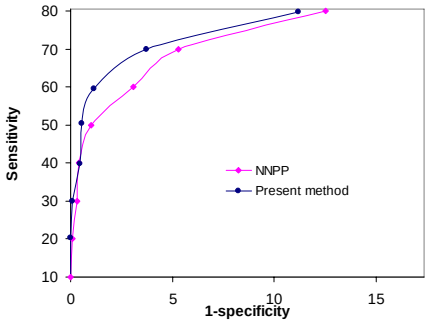


Experiments

>HUMCYC1A CDS 1 Intron 1
GTGAGCGCTGGGCCGGGCCCCGGCCTCCGCGCGGCCCCGCATCTCCGTGAAGGTCACGGC
GGGGAGGCTGCGGGCGCGGGCCTGGGCAGCGCGGAAGCGGTACCGGCCACCCAGCGTCC
CCGGTCCCAGCTGCCTGCCGACCTTGAGCTGGTGGGATCAGGGCTGGGCGCCCACCTCTCC
GAACGGCAGAGAGCCCGTCCCCAGCGTGGGGGTTGGCGGGACGGGCTAGCTGCCGTGGCG
GGGCTGGGGCTTTCCCGAATGGCGCGCCCAGGACGGCTCTTGCGGCTGGCTGTCCAAACT

Data set (Reese, 2001)

- 5800 sequences extracted from EDP release 50
- 300 bp long; 250 bp upstream and 50 bp downstream
- 565 true promoters, 890 CDS and 4345 intron sequences

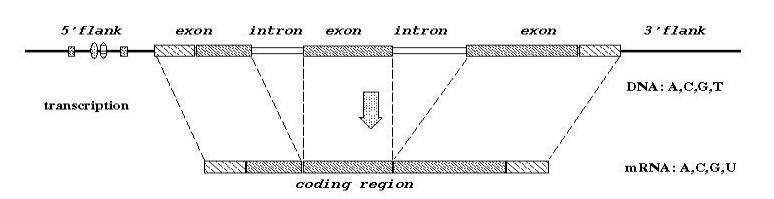


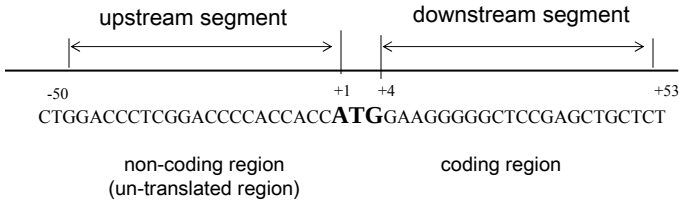
TP true positives	NNPP 2.1	MLP with orthogonal coding	Present method with Markov encoding
30	0.50	0.48	0.52
40	0.60	0.50	0.57
50	0.65	0.51	0.65
60	0.61	0.44	0.69
70	0.58	0.40	0.65
80	0.52	0.23	0.53
90	-	0.20	0.39

Comparison of correlation coefficients, average over five-fold cross validated sets

Letter '-'
ates that NNPP 2.1 was
le to report its correlation
icient at 90% true
ives

Translation Initiation Site (TIS)





Surrounding features of TIS

Highly conserved positions (e.g. +4, -3, Kozak et al.)

The compositional differences between coding downstream region and non-coding upstream region

Periodic properties (codon distributions) in the downstream region

The present of stop codon in the downstream region

The ribosome scanning rule

Hatzigeorgiou's DIANA-TIS

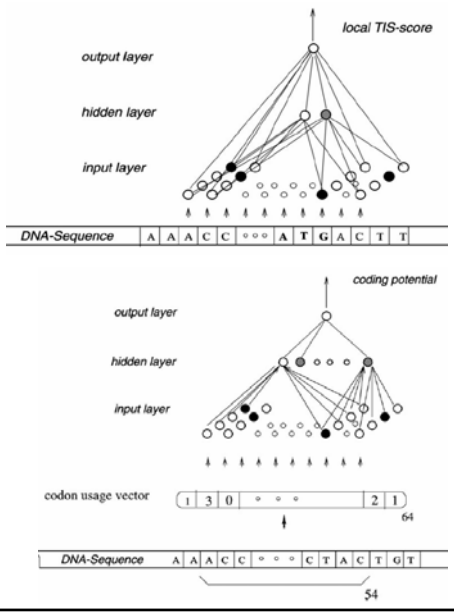
Get local TIS score of ATG and -7 to +5 bases flanking

Get coding potential of 60 in-frame bases up-stream and down-stream

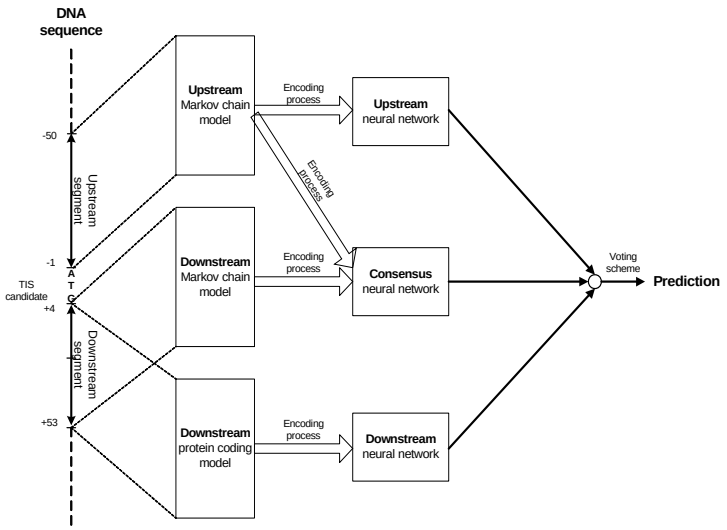
Get coding score by subtracting down-stream from up-stream

ATG may be TIS if product of two scores is > 0.2

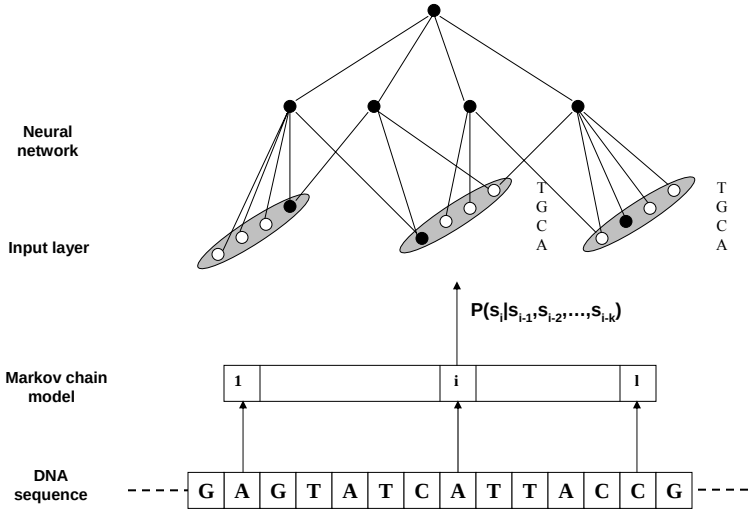
Choose the 1st one



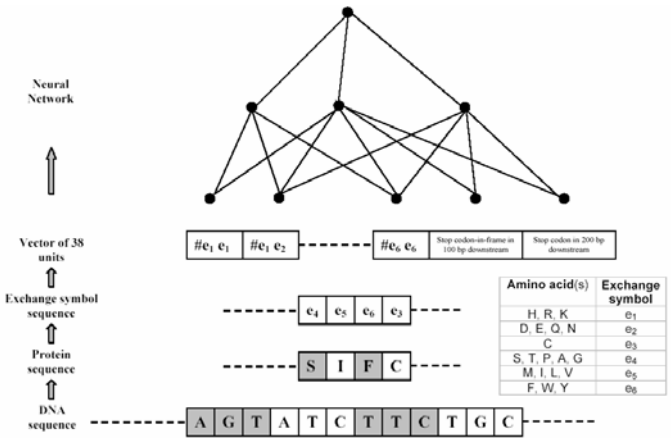
TIS detection with Neural Networks and Markov Encoding



Features input to Upstream and Consensus networks



Features input to Downstream Network



Coding differential

Conservative replacements of amino acids through evolution

Minimum length of a protein

Protein encoding model:

The protein encoding model converts the DNA sequence into a vector of 38 units, representing features of amino acids:

1. Replace each amino acid in corresponding protein sequence by one of six exchange symbols (conserved through evolution).
2. Count all overlapping occurrences of any two consecutive elements (di-exchange symbols) in the resulting sequence
3. Use the first 36 units to summarize the sequence, each unit gives the normalized frequency of the corresponding di-element.
4. Set the unit 37 to a value 1 if the in-frame stop codon is absent within 100 nucleotides downstream of the TIS or a value 0, otherwise
5. Set the unit 38 to a value 0 if the stop codon occurs in every reading frame within 200 nucleotides downstream of the TIS or a value 1, otherwise.

Experiments

```
299 HSU30473.1 CAT U30473 Eukaryotae; mitochondrial eukaryotes; Metazoa; Chordata;
Vertebrata; Eutheria; Primates; Catarrhini; Hominidae; Homo. Homo sapiens
TCCCCATGACAGCGACTGATGAAGAATTTCAATAGAAAGCTGCTACTTCAGAAAATAAGATCATTTGCTGCGAATGGAGA      80
ACATCTCAGGCAGCCCTGATGCTCCACCGGCTCTGGGCATCACCAGCGGCCCCAGGGAAAAAGAAAGAAATGGGAAACAG     160
CATGAAATCCACCCCTGCGCCTGCCGAGAGGCCCCTGCCCAACCCGGAGGGACTGGATAGCGACTTCCTTGCCGTGCTAA     240
GTGACTACCCGTCTCCTGACATCAGCCCCCCGATATTCCGCCGAGGGGAGAAACTGCGT
.....................................................................................      80
.....................................................................iEEEEEEEEEE     160
EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE     240
EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE
```

Data set (Pedersen and Nielsen, 1997)

- 3312 vertebrate DNA sequences
- 13503 ATG sites
- 3312 true TiSs and 10063 false TISs
- 2077 false TISs, that are upstream of true TISs

Three-fold cross validation

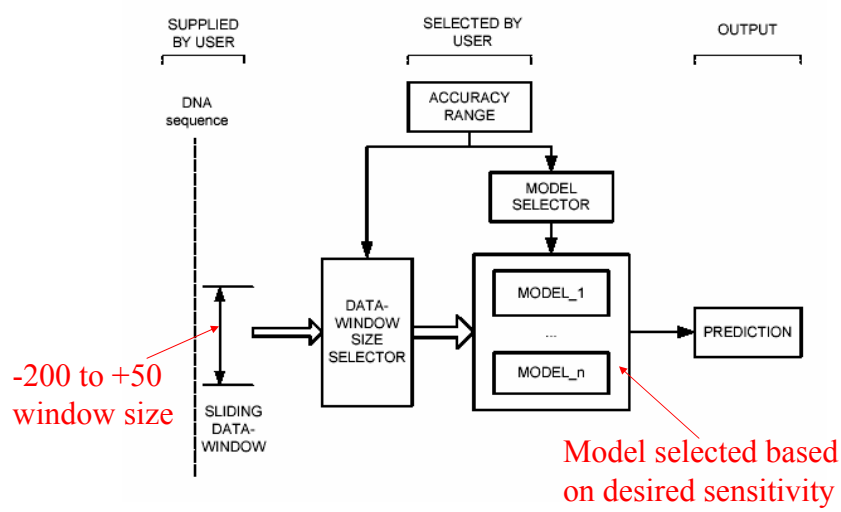
	Sensitivity	Specificity	Precision	Accuracy
Wong *et al.* [9]	88.5%	96.3%	88.6%	94.4%
Pedersen *et al.* [13]	78.0%	87.0%	-	85.0%
Zien *et al.* [21]	69.9%	94.1%	-	88.1%
Hatzigeorgiou [4]	-	-	-	94.0%
Present method	93.8%	96.9%	90.8%	96.1%

Wong's method was equipped with the ribosome scanning model.

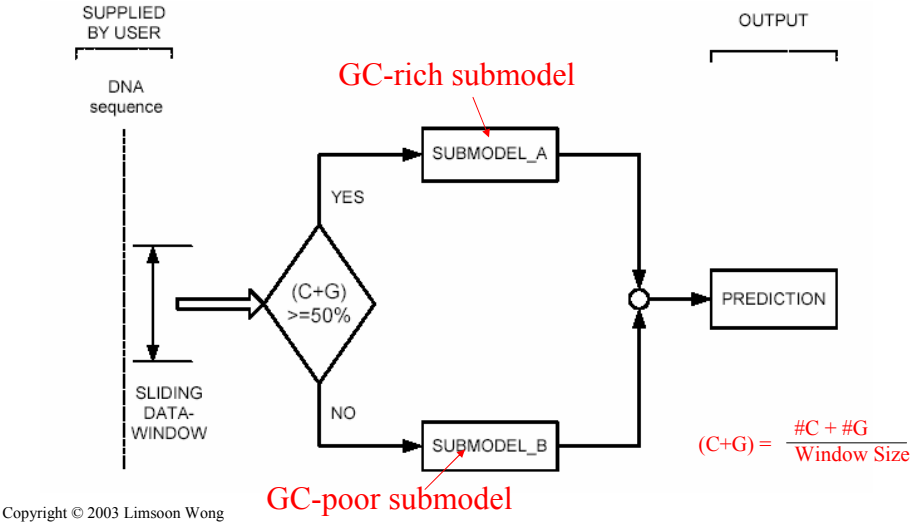
The present method used a majority voting scheme and the ribosome scanning model.

Hatzigeorgiou's method used a different data set (programs or the datasets are unavailable).

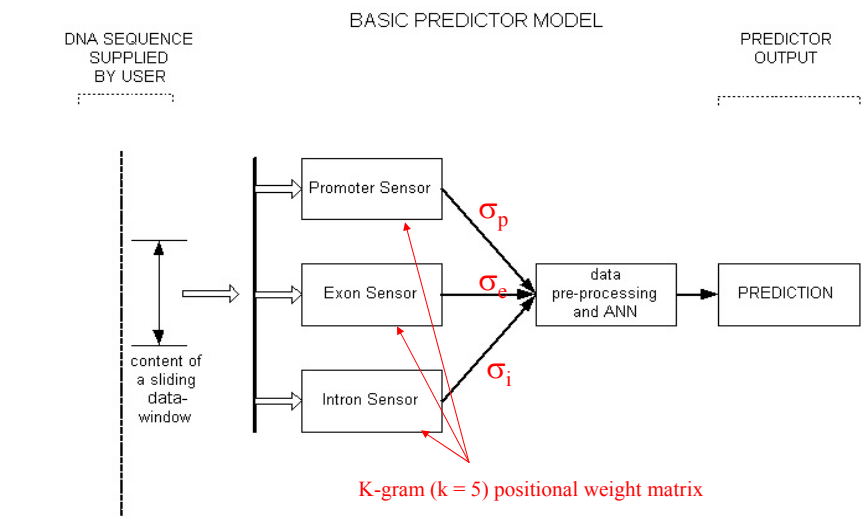
Dragon Promoter Finder



Each model has two submodels based on GC content



Data Analysis Within Submodel



Promoter, Exon, Intron Sensors

These sensors are positional weight matrices of k-grams, k = 5 (aka pentamers)

They are calculated as σ below using promoter, exon, intron data respectively

Window size

$$\sigma = \frac{\left(\sum_{i=1}^{L-4} p_j^i \otimes f_{j,i} \right)}{\left(\sum_{i=1}^{L-4} \max_j f_{j,i} \right)}, \qquad p_j^i \otimes f_{j,i} = \begin{cases} f_{j,i}, \text{ if } p_i = p_j^i \\ 0, \text{ if } p_i \neq p_j^i \end{cases},$$

Pentamer at i^{th} position in input

Frequency of jth pentamer at ith position in training window

j^{th} pentamer at i^{th} position in training window

Data Preprocessing & ANN

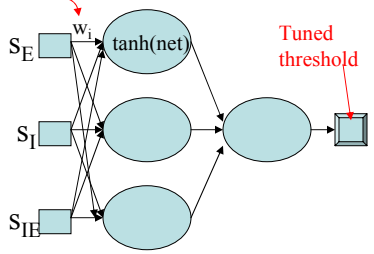
Tuning parameters

$$s_E = sat(\sigma_p - \sigma_e, a_e, b_e),$$
$$s_I = sat(\sigma_p - \sigma_i, a_i, b_i),$$
$$s_{EI} = sat(\sigma_e - \sigma_i, a_{ei}, b_{ei}),$$

where the function sat is defined by

$$sat(x,a,b) = \begin{cases} a, & if \quad x > a \\ x, & if \quad b \le x \le a. \\ b, & if \quad b > x \end{cases}$$

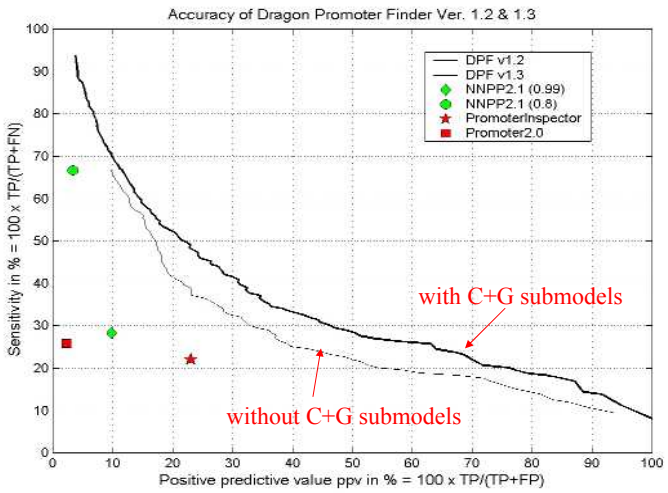
Simple feedforward ANN trained by the Bayesian regularisation method



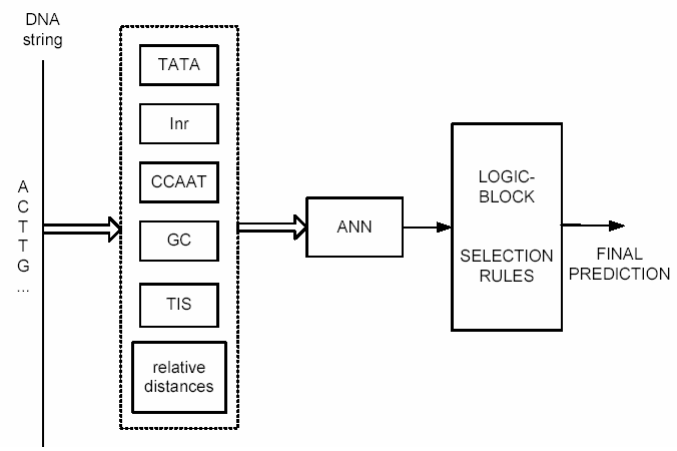
$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

$$net = \Sigma\, s_i * w_i$$

Accuracy Comparisons

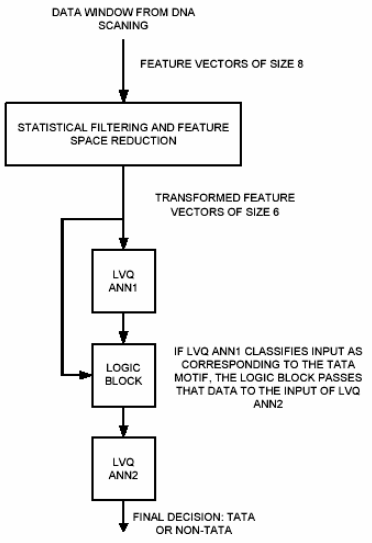


Grail's Promoter Prediction Module



Makes about 1 prediction per 230000 nt at 0.66 sensitivity

LVQ Networks for TATA Recognition



Achieves 0.33 sensitivity at 47 FP on Fickett & Hatzigeorgiou 1997

Sequence Motifs:

A motif is a conserved pattern found in two or more biological sequences (such as DNA, RNA, or protein sequences), that has a specific biological function or structure.

Examples:
TF binding sites in DNA, Ribosome binding sites in RNA, protein sequences with common functions or conserved pieces of structure

We look for motifs in
(1) *gene* families: a set of genes controlled by a common transcription factor or common environmental stimulus (e.g., constructed by microarray experiments)
(2) *protein* families: structurally conserved patterns

Motif Representations:

(1) *Consensus modeling:* involves comparing multiple, similar sequences to determine the general motif
E.g.: TGACGCA, TGACCCA, and AGACGCA; in position one, T out-number A; and position G outnumbers C, so the consensus is given by most probable nucleotides: i.e., TGACGCA

(2) *Degenerate modeling:* involves inventing nucleotides/amino acids that represent probability values of presence, instead.
E.g.: TGACGCA, TGACCCA, and AGACGCA; first nucleotide is either an A or T; thus, represented by hypothetical nucleotide, W (or T with prob. 2/3 and A with prob. 1/3) and in position 5 with X (G with prob. 2/3 and C with prob. 1/3); Consensus: WGACXCA

3. *Probability Weight Matrix* (PWM) is used to represent the nucleotide content in each position with probabilities.

Profile Analysis

The method involves building *profiles* (also referred to as *weight matrices*, templates, or position specific scoring matrices) for motifs or a given set of sequence families.

Profile analysis uses the fact that certain positions in a family are more conserved than other positions and allows substitutions less readily in these conserved positions.

The aim of the profile analysis is often to determine if a given sequence belongs to a particular sequence family or is a particular motif. The profile analysis that follows restricts to protein families, but is applicable to DNA sequences.

Steps of the profile analysis:

1. Align the sequences in the family
2. Create a profile, using the alignment
3. Test new sequences against the profile

Assume no gaps in the alignment and look at the alignment of m sequences of W positions, $(x_{i1}, x_{i2}, \ldots x_{iW})$ where $i = 1, \ldots m$:

$$\begin{matrix} x_{11} & x_{12} & x_{13} \ldots\ldots x_{1W} \\ x_{21} & x_{22} & x_{23} \ldots\ldots x_{2W} \\ \vdots & & \\ x_{m1} & x_{m2} & x_{m3} \ldots\ldots x_{mW} \end{matrix}$$

where $x_{ij} \in \Omega$ denotes the amino acid at j th position of i th sequence.

We build the profile $P_{x,j}$, $x \in \Omega$ and j=1,2,…W as

$$P_{xj} = \frac{f_{xj}}{b_x} \quad \text{, for all } x \in \Omega.$$

where f_{xj} is the percentage of column j containing amino acid x and b_x is the percentage of amino acid x in the *background* distribution. The background can be computed , for example, from a large sequence database, or from a genome, or from some particular protein family.

Intuitively, P_{xj} is the *propensity* for amino acid x in the j th position of the alignment.

Often, it is assumed that in the background, the elements are equally distributed; in such case the profile matrix $\{ P_{x,j} \}$becomes the positional weight matrix (PWM).

Example:

position

Amino acid	1	2	3	4 …	W
A	P_{A1}	P_{A2}	P_{A3} …		…P_{AW}
V	P_{V1}	P_{V2}	P_{V3} …		…P_{AW}
F	P_{F1}				
⋮					

In order to compute a score of presence of an amino acid, usually the log is used:

$score_{xj}$ = log (P_{xj}), for $x \in \Omega$ and j=1,2,…W .

Example: consider for the following alignment (W=4),

LEVK
LDIR
LEIK
LDVE

Assume that all amino acids are equally likely in the background. The weight matrix of the motif could be computed by

$$P_{L1} = \frac{4/4}{1/20} = 20; \quad P_{D2} = \frac{2/4}{1/20} = 10; \quad P_{E2} = \frac{2/4}{1/20} = 10;...$$

The weight matrix representing the family of sequences of the motif of length 4 containing in the set of sub-sequences is given by $\{ P_{x,j} \}_{|\Omega| \text{x} W}$

To use a profile to score for the presence of a motif in a given subsequence,

1. Slide a window of width W over the new sequence
2. The sum of the scores of each position in the window gives the overall score for the subsequence. Score giving the likelihood of a motif to begin at i.

$$\text{score}_i = \sum_{j=1}^{W} \log P_{x_{i+j},j}$$

Example: consider the new sequence, LEVEER,
For earlier profile:
Score at first site = *score*L1 + *score*E2 + *score*V3 + *score*E4
Score at second site = *score*E1 + *score*V2 + *score*E3 + *score*E4
Score at third site = *score*V1 + *score*E2 + *Score*E3 + *Score*R4

If the total score is higher than a threshold, the subsequence in the window is considered to be a member of the family or have the motif. The higher the score, more confident the presence of the motif at that location is.

Profile method could be justified from log-odds perspective. In particular, when scoring a subsequence, it assumes that each position is independent and estimated:

$$\log\left(\frac{P(\text{subsequence} \mid \text{family})}{P(\text{subsequence} \mid \text{not family})}\right)$$

The probabilities of numerator are estimated from frequencies of each amino acid being in that position of alignment. Probabilities in the denominator are estimated from background frequencies.

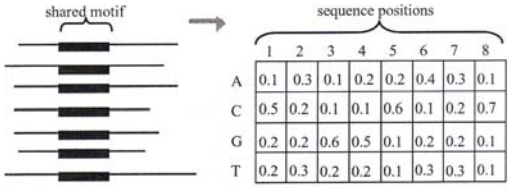
Note: An extension to above formula is required to handle zero frequency cases. If amino acid does not occur in a column, P_{xj} is zero and the score is undefined. A pseudo-count is used to deal with this:

$$P_{xj} = \frac{\#\text{ of amino acid } x \text{ in position } j + \varepsilon}{\text{fraction of amino acid } x \text{ in Genebank} \quad m + 20\varepsilon}$$

where ε is usually a small value (ε < 1.0).

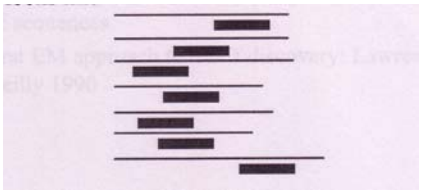
Expectation Maximization (EM) Approach

Given a set of aligned sequences, it is straightforward to construct a positional weight matrix (PWM) characterizing a motif of interest.

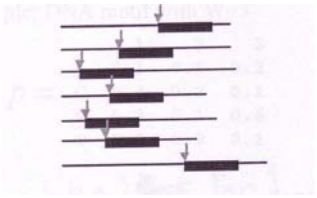


	1	2	3	4	5	6	7	8
A	0.1	0.3	0.1	0.2	0.2	0.4	0.3	0.1
C	0.5	0.2	0.1	0.1	0.6	0.1	0.2	0.7
G	0.2	0.2	0.6	0.5	0.1	0.2	0.2	0.1
T	0.2	0.3	0.2	0.2	0.1	0.3	0.3	0.1

How can we construct the profile if the sequences are not aligned? In a typical case, we don't know what the motif looks like. Then, we use Expectation Maximization (EM) algorithm:



EM is a family of algorithms for learning probabilistic models in problems that involve hidden states. In motif recognition, the hidden states are where the motif starts in training sequences:



Motif is represented by positional weight matrix (PWM) of probabilities $\mathbf{p} = \{ p_{x,j} \}_{|\Omega| \times W}$ where $p_{x,j}$ represents the probability of the character $x \in \Omega$ in column j. It is assumed to have a fixed width W.

Example: a DNA motif with $W = 3$ has a PWM of

		1	2	3
	A	0.1	0.5	0.2
$\mathbf{p}$ =	C	0.4	0.2	0.1
	G	0.3	0.1	0.6
	T	0.2	0.2	0.1

Note that $\sum_{x \in \Omega} p_{x,j} = 1, \quad \text{for all } j$

We also represent the *background* (i.e., outside the motif) by the probability of each character in the background by a column matrix:

$$\mathbf{p}_0 = \{ p_{x,0} \}_{|\Omega| \times 1}$$

where $p_{x,0}$ represents the probability of character x in the background.

Example:

		0
	A	0.26
$\mathbf{p}_0$ =	C	0.24
	G	0.23
	T	0.27

The hidden states of the model are the places where the motif start. Let's define the matrix $Z = \{ Z_{i,j} \}_{mxL\text{-}W+1}$ where the element $Z_{i,j}$ represents the probability that the motif starts in position j of the sequence i.

L is the sequence length and m is the number of sequences.

Example: Given 3 DNA sequences of length $L = 6$, where W=3:

		1	2	3	4
	seq1	0.1	0.1	0.2	0.6
Z =	seq2	0.4	0.2	0.1	0.3
	seq3	0.3	0.4	0.2	0.1

Note that $\sum_{j=1}^{L-W+1} Z_{i,j} = 1$

EM Approach to Motif Detection

Given: length parameter W, a training set of m sequences set initial values for **p**

do

re-estimate Z from **p** (E-step)

re-estimate **p** from Z (M-step)

until change in **p** < δ

Return: **p**, Z

E-step estimates the most probable locations of the motifs

M-step estimates the new motif configuration.

The convergence is achieved when the change (the entropy) of the motif falls below a threshold.

$$H(p) = -\sum_{x \in \Omega} \sum_{j=1}^{W} p_{x,j} \log p_{x,j}$$

Or the likelihood of the sequences, given the motif, exceeds a particular threshold.

Suppose, given m number of sequences of length L, we are looking for a motif of length W : $\mathbf{x}_i = (x_{i,1}, x_{i,2}, \ldots x_{i,L})$, i=1, 2, … m.
The probability of a training sequence having the motif starting at position j: $P(\mathbf{x}_i \mid Z_{i,j} = 1, \mathbf{p}, \mathbf{p}_0) = \prod_{k=1}^{j-1} p_{x_{i,k},0} \prod_{k=j}^{j+W-1} p_{x_{i,k},k-j+1} \prod_{k=j+W}^{L} p_{x_{i,k},0}$

where $Z_{ij} = 1$ indicates that the motif starts at position j in sequence i.
E-Step: estimating Z at the iteration t:

$$Z_{i,j}^t = P(Z_{i,j} = 1 \mid \mathbf{x}_i, \mathbf{p}^t, \mathbf{p}_0^t) = \frac{P(Z_{i,j} = 1, \mathbf{x}_i \mid \mathbf{p}^t, \mathbf{p}_0^t)}{\sum_{k=1}^{L-W+1} P(Z_{i,k} = 1, \mathbf{x}_i \mid \mathbf{p}^t, \mathbf{p}_0^t)}$$

$$= \frac{P(\mathbf{x}_i \mid Z_{i,j} = 1, \mathbf{p}^t, \mathbf{p}_0^t) P(Z_{i,j} = 1)}{\sum_{k=1}^{L-W+1} P(\mathbf{x}_i \mid Z_{i,k} = 1, \mathbf{p}^t, \mathbf{p}_0^t) P(Z_{i,k} = 1)}$$

$$= \frac{P(\mathbf{x}_i \mid Z_{i,j} = 1, \mathbf{p}^t, \mathbf{p}_0^t)}{\sum_{k=1}^{L-W+1} P(\mathbf{x}_i \mid Z_{i,k} = 1, \mathbf{p}^t, \mathbf{p}_0^t)}, \text{assuming an equally likely start of the motif}$$

Further, $\sum_{j=1}^{L-W+1} Z_{i,j} = 1.0$, for all $i = 1, 2, \ldots m$.

M-step: calculating p's from Z:

Recall $p_{x,k}$ represents the probability that character x in position k in the motif; values for position 0 represents the background.

For the motif:

$$p_{x,k}^{t+1} = \frac{n_{x,k} + \varepsilon_k}{\sum_{x' \in \Omega} (n_{x',k} + \varepsilon_k)} \quad \text{, for all } x \in \Omega.$$

where the frequency, $n_{x,k} = \sum_i \sum_{j=1, x_{i,j+k-1}=x}^{L-w+1} Z_{i,j}^t$

for background, the above equation for $p_{x,k}^{t+1}$ is applicable with

$$n_{x,0} = n_x - \sum_{j=1}^{W} n_{x,j}$$

where n_x indicates the total number of characters x in the dataset.
and ε_k indicates the pseudo-counts to deal with the zero frequencies.

The EM algorithm converges to a local maximum in the likelihood of the data given the model:

$$P(\mathbf{x} \mid \mathbf{p}, \mathbf{p}_0) = \prod_{i=1}^{m} P(\mathbf{x}_i \mid \mathbf{p}, \mathbf{p}_0)$$

Usually converges in a small number of iterations

Sensitive to initial starting point (i.e. initial values of **p**)

Example 1: Given the above probability vectors for background and the motif, find the positions of the motifs (Z matrix) for the following alignment and then the new profile matrix for the motif:

A C A G C

A G G C A

T C A G T

$P(x_1 \mid Z_{1,1} = 1, \mathbf{p}, \mathbf{p}_0) = p_{A,1} p_{C,2} p_{A,3} p_{G,0} p_{C,0} = 0.1\text{x}0.2\text{x}0.2\text{x}0.23\text{x}0.24 = 0.00022$

$P(x_1 \mid Z_{1,2} = 1, \mathbf{p}, \mathbf{p}_0) = p_{A,0} p_{C,1} p_{A,2} p_{G,3} p_{C,0} = 0.26\text{x}0.4\text{x}0.5\text{x}0.6\text{x}0.24 = 0.0075$

$P(x_1 \mid Z_{1,3} = 1, \mathbf{p}, \mathbf{p}_0) = p_{A,0} p_{C,0} p_{A,1} p_{G,2} p_{C,3} = 0.26\text{x}0.24\text{x}0.1\text{x}0.1\text{x}0.1 = 0.000062$

$P(x_2 \mid Z_{2,1} = 1, \mathbf{p}, \mathbf{p}_0) = p_{A,1} p_{G,2} p_{G,3} p_{C,0} p_{A,0} = 0.1\text{x}0.1\text{x}0.6\text{x}0.24\text{x}0.26 = 0.00037$

$P(x_2 \mid Z_{2,2} = 1, \mathbf{p}, \mathbf{p}_0) = p_{A,0} p_{G,1} p_{G,2} p_{C,3} p_{A,0} = 0.26\text{x}0.3\text{x}0.1\text{x}0.1\text{x}0.26 = 0.0002$

$P(x_2 \mid Z_{2,3} = 1, \mathbf{p}, \mathbf{p}_0) = p_{A,0} p_{G,0} p_{G,1} p_{C,2} p_{A,3} = 0.26\text{x}0.23\text{x}0.3\text{x}0.2\text{x}0.2 = 0.0007$

$P(x_3 \mid Z_{3,1} = 1, \mathbf{p}, \mathbf{p}_0) = p_{T,1} p_{C,2} p_{A,3} p_{G,0} p_{T,0} = 0.2\text{x}0.2\text{x}0.2\text{x}0.23\text{x}0.27 = 0.00049$

$P(x_3 \mid Z_{3,2} = 1, \mathbf{p}, \mathbf{p}_0) = p_{T,0} p_{C,1} p_{A,2} p_{G,3} p_{T,0} = 0.27\text{x}0.4\text{x}0.5\text{x}0.6\text{x}0.27 = 0.0087$

$P(x_3 \mid Z_{3,3} = 1, \mathbf{p}, \mathbf{p}_0) = p_{T,0} p_{C,0} p_{A,1} p_{G,2} p_{T,3} = 0.27\text{x}0.24\text{x}0.1\text{x}0.1\text{x}0.1 = 0.000065$

Since $\sum_j Z_{i,j} = 1$, for all i = 1,2,...*m*, normalizing the above values row - wise,

$$Z = \begin{bmatrix} 0.028 & 0.964 & 0.008 \\ 0.121 & 0.651 & 0.228 \\ 0.053 & 0.940 & 0.007 \end{bmatrix}$$

$A \quad C \quad A \quad G \quad C$

$Z_{1,1} = 0.028 \; Z_{1,2} = 0.964 \; Z_{1,3} = 0.008$

$A \quad G \quad G \quad C \quad A$

$Z_{2,1} = 0.121 \; Z_{2,2} = 0.651 \; Z_{2,3} = 0.228$

$T \quad C \quad A \quad G \quad T$

$Z_{3,1} = 0.053 \; Z_{3,2} = 0.940 \; Z_{3,3} = 0.007$

By taking $\varepsilon = 1$, $p_{x,k} = \dfrac{n_{x,k} + 1}{\sum_{x \in \Omega} n_{x,k} + 4}$

where for the motif, $n_{x,k} = \sum_{i=1}^{m} \sum_{j=1 | x_{i,j+k-1}=x}^{L-W+1} Z_{i,j}$;

$n_{A,1} = Z_{1,1} + Z_{1,3} + Z_{2,1} + Z_{3,3} = 0.164$; $n_{A,2} = Z_{1,2} + Z_{3,2} = 1.904$; $n_{A,3} = Z_{1,1} + Z_{2,3} + Z_{3,1} = 0.309$;

$n_{T,1} = Z_{3,1} = 0.053$; $n_{T,2} = 0$; $n_{T,3} = Z_{3,3} = 0.007$;

$n_{G,1} = Z_{2,2} + Z_{2,3} = 0.879$; $n_{G,2} = Z_{1,3} + Z_{2,1} + Z_{2,2} + Z_{3,3} = 0.787$; $n_{G,3} = Z_{1,2} + Z_{2,1} + Z_{3,2} = 2.025$;

$n_{C,1} = Z_{1,2} + Z_{3,2} = 0.964$; $n_{C,2} = Z_{1,1} + Z_{2,3} + Z_{3,1} = 0.309$; $n_{C,3} = Z_{1,3} + Z_{2,2} = 0.659$;

for background, $n_{x,0} = n_x - \sum_{j=1}^{W} n_{x,j}$

MEME (Multiple EM Elicitation) Approach (Bailey, 1994)

MEME enhances the basic EM approach in the following ways:

-- trying many starting points

-- not assuming that there is exactly one motif occurrence in every sequence

-- allowing multiple motifs to be learned

-- incorporating Dirichlet prior distributions

Starting points in MEME:

For every distinct subsequence of length *W* in the training set:

-- derive an initial **p** matrix from this subsequence

-- run EM for 1 iteration

Choose the motif model (i.e., **p** matrix) with highest likelihood

Run EM for convergence.

Using subsequences as starting points for EM:

Set values corresponding to letters in the subsequence, say q

Set other values to $(1-q)/(|\Omega|-1)$

Example: for the sequence TAT with $q = 0.5$:

		1	2	3
	A	0.17	0.5	0.17
	C	0.17	0.17	0.17
p =	G	0.17	0.17	0.17
	T	0.5	0.17	0.5

ZOOPS Model

The approach so far we discussed assumes that each sequence has exactly one motif occurrence per sequence: that is the OOPS model.

The ZOOPS model assumes zero or one occurrences per sequence.

E-step in the ZOOPS model:

We need to consider another alternative: the i th sequence doesn't contain the motif.

Another variable (and its relative) is added:

λ: the prior probability that any position in a sequence is the start of a motif

$\gamma = (L-W+1)\lambda$: prior probability of a sequence containing a motif

computation of Z:

$$Z_{i,j}^{t} = \frac{P(\mathbf{x}_i \mid Z_{i,j} = 1, \mathbf{p}^t, \mathbf{p}_0^t)\lambda^t}{P(\mathbf{x}_i \mid Q_i = 0, \mathbf{p}^t, \mathbf{p}_0^t)(1-\gamma^t) + \sum_{k=1}^{L-W+1} P(\mathbf{x}_i \mid Z_{i,k} = 1, \mathbf{p}^t, \mathbf{p}_0^t)\lambda^t}$$

here Q_i is a random variable that takes 0 to indicate that the sequence doesn't contain a motif occurrence:

$$Q_i = \sum_{j=1}^{L-W+1} Z_{i,j}$$

M-step in the ZOOPS model:
Update p same as before
Update λ and γ as follows

$$\lambda^{t+1} = \frac{\gamma^{t+1}}{L-W+1} = \frac{1}{m(L-W+1)} \sum_{i=1}^{m} \sum_{j=1}^{L-W+1} Z_{i,j}^{t}$$

Average of $Z_{i,j}$ across all sequences, and positions.

TCM model

The TCM (two-component mixture model) assumes zero or more occurrences per sequence

The TCM model treats each length W subsequence independently. If $\mathbf{x}_{i,j}$ denotes the subsequence starting at site j of sequence i, the likelihood of such a subsequence is given by

$$P(\mathbf{x}_{i,j} \mid Z_{i,j} = 1, \mathbf{p}, \mathbf{p}_0) = \prod_{k=j}^{j+W-1} p_{x_{ik}, k-j+1}, \quad \text{assuming a motif starts there;}$$

$$P(\mathbf{x}_{i,j} \mid Z_{i,j} = 0, \mathbf{p}, \mathbf{p}_0) = \prod_{k=j}^{j+W-1} p_{x_{ik}, 0}, \quad \text{assuming a motif doesn't start there;}$$

E-step in the TCM model:

$$Z_{i,j}^{t} = \frac{P(X_{i,j} \mid Z_{i,j} = 1, \mathbf{p}^{t}, \mathbf{p}_{0}^{t})\lambda^{t}}{P(X_{i,j} \mid Z_{i,j} = 0, \mathbf{p}^{t}, \mathbf{p}_{0}^{t})(1-\lambda^{t}) + P(\mathbf{x}_{i,j} \mid Z_{i,j} = 1, \mathbf{p}^{t}, \mathbf{p}_{0}^{t})\lambda^{t}}$$

M-step is same as in the ZOOPS model.

TCM model describes sequences with multiple *occurrences* of the *same* motif.

Finding Multiple Motifs

Basic idea: discount the likelihood that a new motif starts in a given position if the motif would overlap with a previously learned one (a greedy method).

When re-estimating $Z_{i,j}$, multiply by $P(V_{i,j} = 1)$ where

$$V_{i,j} = \begin{cases} 1, & \text{no previous motifs in } [x_{i,j}, \ldots x_{i,j+w-1}] \\ 0, & \text{otherwise.} \end{cases}$$

V_{ij} is estimated using Z_{ij} values from previous *passes* of motif finding.

REFERENCES

J. C. Rajapakse and L. S. Ho, "Markov encoding for detecting signals in genomic sequences," *IEEE Transactions on Computational Biology and Bioinformatics*, Vol. 2, No. 2, pp. 131-142, April-June 2005.

L. Wong et al., "Using feature generation and feature selection for accurate prediction of translation initiation sites", *GIW* 13:192--200, 2002

A. Zien et al., "Engineering support vector machine kernels that recognize translation initiation sites", *Bioinformatics* 16:799--807, 2000

A. G. Hatzigeorgiou, "Translation initiation start prediction in human cDNAs with high accuracy", *Bioinformatics* 18:343--350, 2002

V. B. Bajic et al., "Dragon Promoter Finder: Recognition of vertebrate RNA polymerase II promoters", *Bioinformatics* 18:198--199, 2002.

M. G. Reese, "Application of a time-delay neural network to promoter annotation in the *D. melanogaster* genome", *Comp. & Chem.* 26:51--56, 2001

A. G. Pedersen et al., "The biology of eukaryotic promoter prediction---a review", *Comp. & Chem.* 23:191--207, 1999

S. Knudsen, "Promoter 2.0 for the recognition of Pol II promoter sequences", *Bioinformatics* 15:356--361, 1999.

H. Wang, "Statistical pattern recognition based on LVQ ANN: Application to TATA-box motif", *M.Tech Thesis*, Technikon Natal, South Africa

A. G. Pedersen, H. Nielsen, "Neural network prediction of translation initiation sites in eukaryotes", *ISMB* 5:226--233, 1997

J. W. Fickett, A. G. Hatzigeorgiou, "Eukaryotic promoter recognition", *Gen. Res. 7:861--878,* 1997

T. L. Bailey and C. Elkan, "The value of prior knowledge in discovering motifs with MEME," *ISMB*, pp. 21-29, 1995