Gene Prediction

This exposition is based on the following sources, which are all recommended reading (in this order):

1. Pavel A. Pevzner. Computational Molecular Biology, an algorithmic approach. MIT, 2000, chapter 9.
2. Chris Burge and Samuel Karlin. Prediction of complete gene structures in human genomic DNA. Journal of Molecular Biology, 268:78-94 (1997).
3. Ian Korf, Paul Flicek, Danial Duan and Michael R. Brent, Integrating Genomic Homology into Gene Structure Prediction, Bioinformatics, Vol. 1 Suppl 1., pages S1-S9 (2001).
4. Vineet Bafna and Daniel Huson. The conserved exon method for gene finding. ISMB 2000, 3-12 (2000).
5. M. S. Gelfand, A. Mironov and P. A. Pevzner, Gene recognition via spliced alignment, PNAS, 93:9061–9066 (1996).

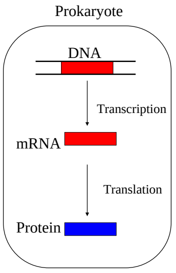
Introduction

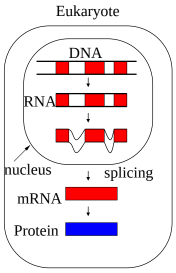
In the 1960s, it was discovered that a gene and its protein product are colinear structures with a direct correlation between the triplets of nucleotides in the gene and the amino acids in the protein.

It soon became clear that genes can be difficult to determine, due to the existence of overlapping genes, and genes within genes etc.

Moreover, the paradox arose that the genome size of many eukaryotes does not correspond to “genetic complexity”, for example, the salamander genome is 10 times the size of that of human.

In 1977, the surprising discovery of “split” genes was made: genes that consist of multiple pieces of coding DNA called *exons*, separated by stretches of non-coding DNA called *introns*.





The existence of split genes and junk-DNA raises a computational gene prediction problem that is still unsolved:

> Given a string of DNA. The gene prediction problem is to reliably predict all genes contained in the sequence.

Three approaches to gene finding

One can distinguish between three types of approaches:

- Statistical or *ab initio* methods. These methods attempt to predict genes based on statistical properties of the given DNA sequence. Programs are e.g. GENSCAN, GeneID, GENIE and FGENEH.
- Homology methods. The given DNA sequence is compared with known protein structures, e.g. using “spliced alignments”. Programs are e.g. Procrustes and GeneWise.
- Comparative methods. The given DNA string is compared with a similar DNA string from a different species at the appropriate evolutionary distance and genes are predicted in both sequences based on the assumption that exons will be well conserved, whereas introns will not. Programs are e.g. CEM (conserved exon method) and TWINSCAN.

Simplest approach to gene prediction

The simplest way to detect potential coding regions is to look at *Open Reading Frames (ORFs)*. An ORF is a sequence of codons in DNA that starts with a Start codon (ATG), ends with a Stop codon (TAA, TAG or TGA) and has no other (in-frame) stop codons inside.

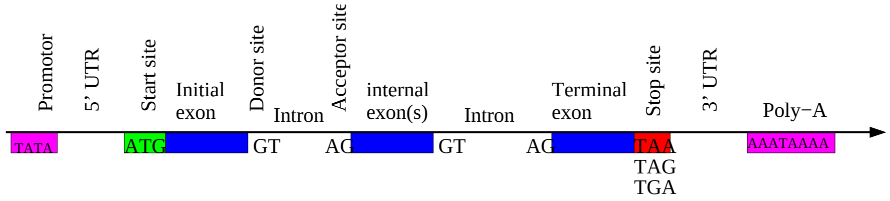
The average distance between stop codons in “random” DNA is $\frac{64}{3} \approx 21$, much smaller than the number of codons in an average protein (≈ 300).

Essentially, long ORFs indicate genes, whereas short ORF may or may not indicate genes or short exons.

Additionally, features such as *codon usage* or hexamer counts can be taken into account. The *codon usage* of a string of DNA is given by a 64-component vector that counts how many times each codon is present in the string. These values can differ significantly between coding and non-coding DNA.

Eukaryotic gene structure

For our purposes, a eukaryotic gene has the following structure:

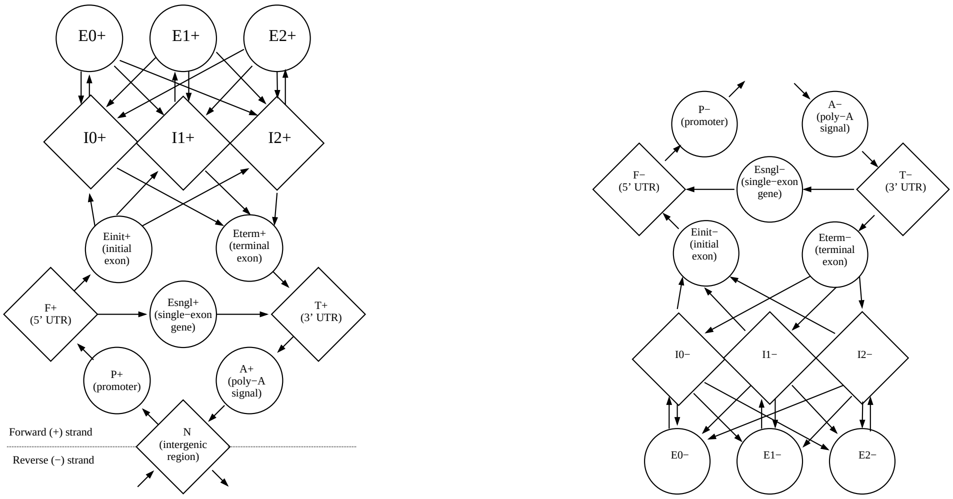


Ab initio gene prediction methods use statistical properties of the different components of such a gene model to predict genes in unannotated DNA. For example, for the bases around the start site we may have the following observed frequencies (given by this *position weight matrix*):

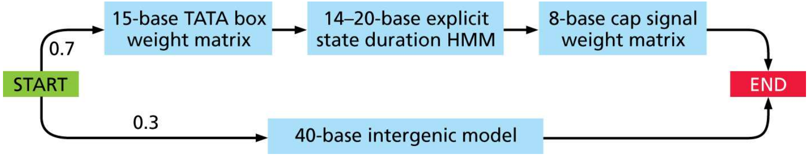
Pos.	-8	-7	-6	-5	-4	-3	-2	-1	+1	+2	+3	+4	+5	+6	+7
A	.16	.29	.20	.25	.22	.66	.27	.15	1	0	0	.28	.24	.11	.26
C	.48	.31	.21	.33	.56	.05	.50	.58	0	0	0	.16	.29	.24	.40
G	.18	.16	.46	.21	.17	.27	.12	.22	0	0	1	.48	.20	.45	.21
T	.19	.24	.14	.21	.06	.02	.11	.05	0	1	0	.09	.26	.21	.21

GENSCAN’s model

We are going to discuss the popular program GENSCAN in detail, which is based on a *semi-Markov* model:



First note that each node in the above model is a model itself. For example the promoter singal node is depicted as follows:


15-base TATA box weight matrix
14–20-base explicit state duration HMM
8-base cap signal weight matrix
0.7
START
END
0.3
40-base intergenic model

GENSCAN's model can be formulated as an *explicit state duration* HMM. This is an HMM in which, additionally, a *duration period* is explicitly modeled for each state, using a probability distribution.

The model is thought of generating a *parse* ϕ, consisting of:

- a sequence of states $q = (q_1, q_2, \ldots, q_n)$, and
- an associated sequence of durations $d = (d_1, d_2, \ldots, d_n)$,

which, using probabilistic models for each of the state types, generates a DNA sequence S of length $L = \sum_{i=1}^{n} d_i$.

The generation of a parse of a given sequence length L proceeds as follows:

1. An initial state q_1 is chosen according to an initial distribution π on the states, i.e. $\pi_i = P(q_1 = Q^{(i)})$, where $Q^{(j)}$ $(j = 1, \ldots, 27)$ is an indexing of the states of the model.
2. A state duration or *length* d_1 is generated conditional on the value of $q_1 = Q^{(i)}$ from the duration distribution $f_{Q^{(i)}}$.
3. A sequence segment s_1 of length d_1 is generated, conditional on d_1 and q_1, according to an appropriate sequence generating model for state type q_1.
4. The subsequent state q_2 is generated, conditional on the value of q_1, from the (first-order Markov) state transition matrix T, i.e. $T_{i,j} = P(q_{k+1} = Q^{(j)} \mid q_k = Q^{(i)})$.

This process is repeated until the sum $\sum_{i=1}^{n} d_i$ of the state durations first equals or exceeds L, at which point the last state duration is appropriately truncated, the final stretch of sequence is generated and the process stops.

The resulting sequence is simply the concatenation of the sequence segments, $S = s_1 s_2 \dots s_n$.

Note that the generated sequence is not restricted to correspond to a single gene, but could represent multiple genes, in both strands, or none.

In addition to its topology involving the 27 states and 46 transitions depicted above, the model has four main components:

- a vector of initial probabilities π,
- a matrix of state transition probabilities T,
- a set of length distributions f, and
- a set of sequence generating models P.

(Recall that an HMM has initial-, transition- and emission probabilities).

Maximum likelihood prediction

Given such a model M. For a fixed sequence length L, consider

$$\Omega = \Phi_L \times \mathcal{S},$$

where Φ_L is the set of all possible parses of M of length L and $\mathcal{S}_L$ is the set of all possible sequences of length L.

The model M assigns a probability density to each point (parse/sequence pair) in Ω. Thus, for a given sequence $S \in \mathcal{S}_L$, a conditional probability of a particular parse $\phi \in \Phi_L$ is given by:

$$P(\phi \mid S) = \frac{P(\phi, S)}{P(S)} = \frac{P(\phi, S)}{\sum_{\phi' \in \Phi_L} P(\phi', S)},$$

using $P(M, D) = P(M \mid D)P(D)$.

The essential idea is to specify a precise probabilistic model of what a gene looks like in advance and then to select the parse ϕ through the model M that has highest likelihood, given the sequence S.

Computational issues

Given a sequence S of length L, the joint probability $P(\phi, S)$ of generating the parse ϕ and the sequence S is given by:

$$P(\phi, S) = \pi_{q_1} f_{q_1}(d_1) P(s_1 \mid q_1, d_1) \times \prod_{k=2}^{n} T_{q_{k-1}, q_k} f_{q_k}(d_k) P(s_k \mid q_k, d_k),$$

where the states of ϕ are $q_1, q_2, \ldots, q_n$ with associated state lengths $d_1, d_2, \ldots, d_n$, which break the sequence into segments $s_1, s_2, \ldots, s_n$.

Here, $P(s_k \mid q_k, d_k)$ is the probability of generating the segment s_k under the appropriate sequence generating model for a type-q_k state of length d_k.

A modification of the Viterbi algorithm may be used to calculate ϕ_{opt}, the parse with maximal joint probability (under M), that gives the predicted gene or set of genes in the sequence.

We can compute $P(S)$ using the "forward algorithm" discussed under HMMs. With the help of the "backward algorithm", certain additional quantities of interest can also be computed.

For example, consider the event $E^{(k)}_{[x,y]}$ that a particular sequence segment $[x,y]$ is an internal exon of phase $k \in \{0,1,2\}$. Under M, this event has probability

$$P(E^{(k)}_{[x,y]} \mid S) = \frac{\sum_{\phi: E^{(k)}_{[x,y]} \in \phi} P(\phi, S)}{P(S)},$$

where the sum is taken over all parses that contain the given exon $E^{(k)}_{[x,y]}$. This sum can be computed using the forward-backward algorithm.

Details of the model

So far, we have discussed the topology and the other main components of the GENSCAN model in general terms. The following details need to be discussed:

- the initial and transition probabilities,
- the state length distributions,
- transcriptional and translational signals,
- splice signals, and
- reverse-strand states.

Initial and transition probabilities

For gene prediction in randomly chosen blocks of contiguous human DNA, the initial probability of each state should be chosen proportionally to its estimated frequency in bulk human genomic DNA.

This is a non-trivial problem, because gene density and certain aspects of gene structure vary significantly in regions of differing C+G content (so-called "isochores") of the human genome, with a much higher gene density in C+G-rich regions.

Hence, in practice, initial and transitional probabilities are estimated for four different categories: (I) $<$ 43% C+G, (II) $43-51$% C+G, (III) $51-57$% C+G, and (IV) $>$ 57% C+G.

The following initial probabilities were obtained from a training set of 380 genes by comparing the number of bases corresponding to each of the different states:

Group	I	II	III	IV
C+*G*-range	$<$ 43%	43 $-$ 51%	51 $-$ 57%	$>$ 57%
Initial probabilities:				
Intergenic (N)	0.892	0.867	0.540	0.418
Intron (I_i^+, I_i^-)	0.095	0.103	0.338	0.388
5' UTR (F^+, F^-)	0.008	0.018	0.077	0.122
3' UTR (T^+, T^-)	0.005	0.011	0.045	0.072

For simplicity, the initial probabilities for the exon, promoter and poly-A states were set to 0.

Transition probabilities are obtained in a similar way.

State length distributions

In general, the states of the model correspond to sequence segments of highly variable length.

For certain states, most notably for internal exon states E_k, length is probably important for proper biological function, i.e. proper splicing and inclusion in the final processed mRNA.

For example, it has been shown *in vivo* that internal deletions of exons to sizes below about 50 bp may often lead to exon skipping, and there is evidence that steric interference between factors recognizing splice sites may make splicing of small exons more difficult. There is also evidence that spliceosomal assembly is inhibited if internal exons are expanded beyond 300 bp.

In summary, these arguments support the observation that internal exons are usually $\approx 120 - 150$ bp long, with only a few of length less that 50 bp or more than 300 bp.

Constraints for initial and terminal exons are slightly different.

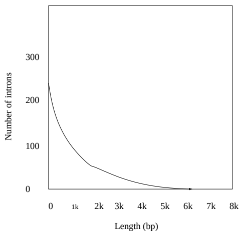
The duration in initial, internal and terminal exon states is modeled by a different empirical distribution for each of the types of states.

In contrast to exons, the length of introns does not seem critical, although a minimum length of $70 - 80$ may be preferred.

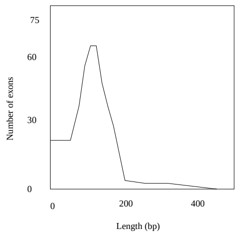
The length distribution for introns appears to be approximately geometric (exponential). However, the average length of introns differs substantially between the different *C*+*G* groups: In group I, the average length is 2069 bp, whereas for group *IV*, the average length is only 518 bp.

Hence, the duration in intron states is modeled by a geometric distribution with parameter q estimated for each *C*+*G* group separately.

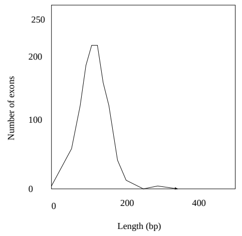
Empirical length distributions for introns and exons:



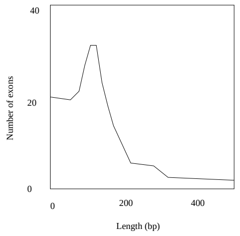
Introns



Initial exons



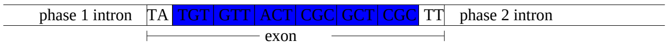
Internal exons



Terminal exons

Note that the exon lengths generated must be consistent with the phases of adjacent introns. To account for this, first the number of complete codons is generated from the appropriate length distribution, then the appropriate number (0, 1 or 2) of bp is added to each end to account for the phases of the preceding and subsequent states.

For example, if the number of complete codons generated for an internal exon is $C = 6$, and the phase of the previous and next intron is 1 and 2, respectively, then the total length of the exon is $l = 3C + 2 + 2 = 22$:



For the $5'$ UTR and $3'$ UTR states, geometric distributions are used with mean values of 769 and 457 bp, respectively.

Simple signal models

There are a number of different models of biological signal sequences, such as donor and acceptor sites, promoters, etc.

One of the earliest and must influential approaches is *the weight matrix method (WMM)*, in which the frequency $p_a^{(i)}$ of each nucleotide a at position i of a signal of length n is derived from a collection of aligned signal sequences.

The product $P(A) = \prod_{i=1}^{n} P_{a_i}^{(i)}$ is used to estimate the probability of generating a particular sequence $A = a_1 a_2 \ldots a_n$.

The *weight array matrix (WAM)* is a generalization that takes dependencies between adjacent positions into account. In this model, the probability of generating a particular sequence is $P(A) = p_{a_1}^{(1)} \prod_{i=2}^{n} p_{a_{i-1},a_i}^{i-1,i}$, where $p_{v,w}^{i-1,i}$ is the conditional probability of generating a particular nucleotide v at position i, given nucleotide w at position $i-1$.

Here is a WMM for recognition of a start site:

Pos.	-8	-7	-6	-5	-4	-3	-2	-1	+1	+2	+3	+4	+5	+6	+7
A	.16	.29	.20	.25	.22	.66	.27	.15	1	0	0	.28	.24	.11	.26

C	.48	.31	.21	.33	.56	.05	.50	.58	0	0	0	.16	.29	.24	.40
G	.18	.16	.46	.21	.17	.27	.12	.22	0	0	1	.48	.20	.45	.21
T	.19	.24	.14	.21	.06	.02	.11	.05	0	1	0	.09	.26	.21	.21

Under this model, the sequence `...CCGCCACC ATG GCGC...` has the highest probability of containing a start site, namely: $P = 0.48 \cdot 0.31 \cdot 46 \cdot 0.33 \cdot 0.56 \cdot 0.66 \cdot 0.5 \cdot 0.58 \cdot 1 \cdot 1 \cdot 1 \cdot 0.48 \cdot 0.29 \cdot 0.45 \cdot 0.4 = 0.006$.

The sequence `...AGTTTTTT ATG TAAT ...` has the lowest non-zero probability of containing a start site at the indicated position, namely: $P = 0.16 \cdot 0.16 \cdot 0.14 \cdot 0.21 \cdot 0.06 \cdot 0.02 \cdot 0.11 \cdot 0.05 \cdot 1 \cdot 1 \cdot 1 \cdot 0.09 \cdot 0.24 \cdot 0.11 \cdot 0.21 = 20.4 \cdot 10^{-11}$.

Transcriptional and translational signals

Poly-A signals are modeled as a 6 bp WMM model with consensus sequence AATAAA.

A 12 bp WMM, beginning 6 bp prior to the start codon, is used for the translation initiation signal.

In both cases, one can estimate the probabilities using the GenBank annotated “polyA_signal” and “CDS” features of sequences.

Approximately 30% of eukaryotic promoters lack a TATA signal. Hence, a TATA-containing promoter is generated with 0.7 probability, and a TATA-less one with probability 0.3.

TATA-containing promoters are modeled as a 15 bp TATA WMM and an 8 bp *cap site* WMM. The length between the two WMMs is generated uniformly from the range 14 – 20 bp.

TATA-less ones are modeled as intergenic regions of 40 bp.

Splice signals

The donor and acceptor splice signals are probably the most important signals, as the majority of exons are internal ones. Previous approaches use WMMs or WAMs to model them, thus assuming independence of sites, or that dependencies only occur between adjacent sites.

The consensus region of the donor splice sites covers the last 3 bp of the exon (positions -3 to -1) and the first 6 bp of the succeeding intron (positions 1 to 6):

			...exon	intron...					
Position	-3	-2	-1	+1	+2	+3	+4	+5	+6
Consensus	c/a	A	G	G	T	a/g	A	G	t
WMM:									
A	.33	.60	.08	0	0	.49	.71	.06	.15
C	.37	.13	.04	0	0	.03	.07	.05	.19
G	.18	.14	.81	1	0	.45	.12	.84	.20
T	.12	.13	.07	0	1	.03	.09	.05	.46

Donor site model

However, donor sites show significant dependencies between non-adjacent positions, which probably reflect details of donor splice site recognition by U1 snRNA and other factors.

Given a sequence *S*. Let C_i denote the *consensus indicator variable* that is 1, if the given nucleotide at position *i* matches the consensus at position *i*, and 0 otherwise. Let X_j denote the nucleotide at position *j*.

For example, consider:

			. . . exon		intron. . .					
Position		-3	-2	-1	+1	+2	+3	+4	+5	+6	
Consensus		c/a	A	G	G	T	a/g	A	G	t	
S	. . . T	A	A	C	G	T	A	A	G	C	C . . .

Here, $C_{-1} = 0$ and $C_{+6} = 0$, and = 1, for all other positions. Similarly, $X_{-3} = A$, $X_{-2} = A$, $X_{-1} = C$ etc.

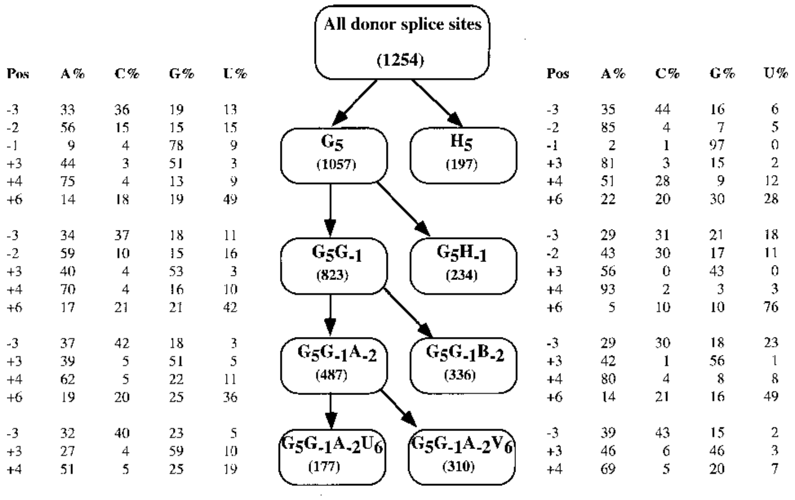
We use χ^2 statistics for the variable C_i *versus* X_j, for all pairs i, j with $i \neq j$ in the set of donor sites from the genes of the given learning set, based on the C_i *versus* X_j contingency table:

	X_j			
C_i	A	C	G	T
0	$f_0(A)$	$f_0(C)$	$f_0(G)$	$f_0(T)$
1	$f_1(A)$	$f_1(C)$	$f_1(G)$	$f_1(T)$,

where $f_i(x)$ is the frequency at which the training set has the consensus base at position *i* and the base *x* at position *j*.

A significant χ^2 score indicates that there is a dependency between site *i* and *j*.

The idea is then to identify an ordering of the sites by decreasing *discriminatory power* and then to derive separate WMMs for each of the different cases, thus obtaining a so-called *maximal dependence decomposition*:



Pos	A%	C%	G%	U%
-3	33	36	19	13
-2	56	15	15	15
-1	9	4	78	9
+3	44	3	51	3
+4	75	4	13	9
+6	14	18	19	49
-3	34	37	18	11
-2	59	10	15	16
+3	40	4	53	3
+4	70	4	16	10
+6	17	21	21	42
-3	37	42	18	3
+3	39	5	51	5
+4	62	5	22	11
+6	19	20	25	36
-3	32	40	23	5
+3	27	4	59	10
+4	51	5	25	19

Pos	A%	C%	G%	U%
-3	35	44	16	6
-2	85	4	7	5
-1	2	1	97	0
+3	81	3	15	2
+4	51	28	9	12
+6	22	20	30	28
-3	29	31	21	18
-2	43	30	17	11
+3	56	0	43	0
+4	93	2	3	3
+6	5	10	10	76
-3	29	30	18	23
+3	42	1	56	1
+4	80	4	8	8
+6	14	21	16	49
-3	39	43	15	2
+3	46	6	46	3
+4	69	5	20	7

Here, $H = A|C|U$, $B = C|G|U$ and $V = A|C|G$. For example, G_5 , or H_5, is the set of donor sites with, or without, a G at position +5, respectively.

Acceptor site model

Intron/exon junctions are modeled by a (first-order) WAM for bases -20 to +3, capturing the pyrimidine (C,T) rich region and the acceptor splice site itself.

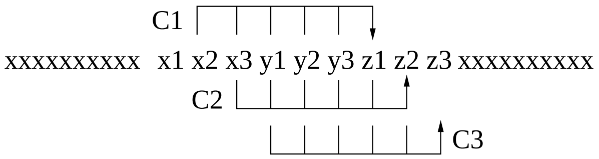
It is difficult to model the branch point in the preceding intron, and only 30% of the test data had an YYRAY sequence in the appropriate region $[-40, -21]$.

A modified variant of a *second-order* WAM is employed in which nucleotides are generated conditional on the previous two ones, in an attempt to model the weak but detectable tendency toward YYY triplets as well as certain branch point-related triplets such as TGA, TAA, GAC, and AAC in this region, without requiring the occurrence of any specific branch point consensus.

(A windowing and averaging process is used to obtain estimates from the limited training data.)

Exon models

Coding portions of exons are modeled using an inhomongeneous 3-periodic fifth order Markov model. Here, separate Markov transition matrices, c_1, c_2 and c_3, are determined for hexamers ending at each of the three codon positions, respectively:



This is based on the observation that frame-shifted hexamer counts are generally the most accurate compositional discriminator of coding *versus* non-coding regions.

However, A+T rich genes are often not well predicted using hexamer counts based on bulk DNA and so GENSCAN uses two different sets of transition matrices, one trained for sequences with $<$ 43% C+G content and one for all others.

Performance studies

The performance of a gene prediction program is evaluated by applying it to DNA sequences for which all contained genes are known and annotated with high confidence.

To calculate accuracy statistics, each nucleotide of a test sequence is classified as:

- a *predicted positive (PP)* if it is predicted to be contained in a coding region,
- a *predicted negative (PN)* if it is predicted to be contained in non-coding region,
- an *actual positive (AP)* if it is annotated to be contained in coding region, and
- an *actual negative (AN)* if it is annotated to be contained in non-coding region.

The performance is measured both on the level of nucleotides and on whole predicted exons, using a similar classification.

Based on this classification, we compute the number of:

- *true positives*, $TP = PP \cap AP$,
- *false positives*, $FP = PP \cap AN$,
- *true negatives*, $TN = PN \cap AN$, and
- *false negatives*, $FN = PN \cap AP$.

The *sensitivity Sn* and *specificity Sp* of a method are then defined as

$$Sn = \frac{TP}{AP} \quad \text{and} \quad Sp = \frac{TP}{PP},$$

respectively, measuring both the ability to predict true genes and to avoid predicting false ones.

Performance of GENSCAN

GENSCAN was run on a test set of 570 vertebrate sequences and the forward strand exons in the optimal GENSCAN parse of the sequence were compared to the annotated exons. The following table shows the results and compares them with results obtained using other programs:

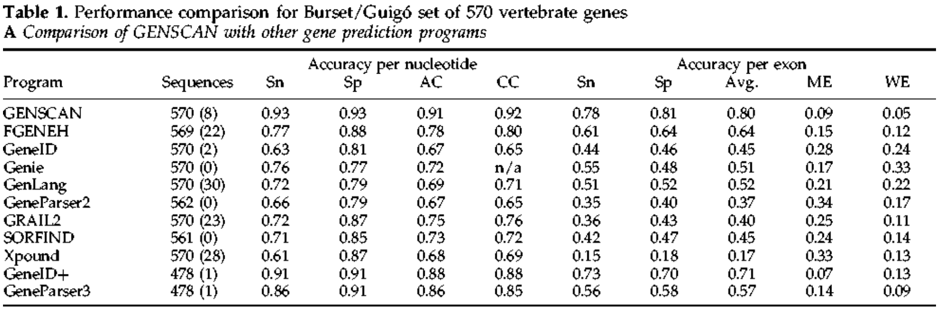


Table 1. Performance comparison for Burset/Guigó set of 570 vertebrate genes
A *Comparison of GENSCAN with other gene prediction programs*

		Accuracy per nucleotide				Accuracy per exon				
Program	Sequences	Sn	Sp	AC	CC	Sn	Sp	Avg.	ME	WE
GENSCAN	570 (8)	0.93	0.93	0.91	0.92	0.78	0.81	0.80	0.09	0.05
FGENEH	569 (22)	0.77	0.88	0.78	0.80	0.61	0.64	0.64	0.15	0.12
GeneID	570 (2)	0.63	0.81	0.67	0.65	0.44	0.46	0.45	0.28	0.24
Genie	570 (0)	0.76	0.77	0.72	n/a	0.55	0.48	0.51	0.17	0.33
GenLang	570 (30)	0.72	0.79	0.69	0.71	0.51	0.52	0.52	0.21	0.22
GeneParser2	562 (0)	0.66	0.79	0.67	0.65	0.35	0.40	0.37	0.34	0.17
GRAIL2	570 (23)	0.72	0.87	0.75	0.76	0.36	0.43	0.40	0.25	0.11
SORFIND	561 (0)	0.71	0.85	0.73	0.72	0.42	0.47	0.45	0.24	0.14
Xpound	570 (28)	0.61	0.87	0.68	0.69	0.15	0.18	0.17	0.33	0.13
GeneID+	478 (1)	0.91	0.91	0.88	0.88	0.73	0.70	0.71	0.07	0.13
GeneParser3	478 (1)	0.86	0.91	0.86	0.85	0.56	0.58	0.57	0.14	0.09

GENSCAN performs very well here and is currently the most popular gene finding method.

Comparative gene finding

GENSCAN's model makes use of statistical features of the genome under consideration, obtained from an annotated training set.

More recently, a number of methods have been suggested that attempt to also make use of comparative data. They are based on the observation that

> the level of sequence conservation between two species depends on the function of the DNA, e.g. coding sequence is more conserved than intergenic sequence.

One such program is Rosetta, which first computes a global alignment of two homologous sequences and then attempts to predict genes in both sequences simultaneously. A second is the conserved exon method, that uses local conservation.

The TWINSCAN program is an extension of GENSCAN, that additionally models a conserved sequence.