IDENTIFICATION OF GENES IN HUMAN GENOMIC DNA

Christopher Burge

Stanford University
Stanford, CA 94305

March 1997

Abstract

A general probabilistic model of the gene structural and compositional properties of human genomic DNA is introduced and applied to the problem of identifying genes in unannotated human genomic sequences. The model uses a "Hidden semi-Markov" or semi-Markov source architecture which incorporates probabilistic descriptions of fundamental transcriptional, translational and splicing signals, as well as length distributions and compositional features of exons, introns and intergenic regions. Distinct sets of model parameters are derived which account for many of the substantial differences in gene density and structure observed in distinct C+G compositional regions ("isochores") of the human genome. A novel model building procedure, termed Maximal Dependence Decomposition, is introduced which captures potentially important dependencies between non-adjacent as well as adjacent positions in a biological signal. Application of this model to the donor splice signal not only gives better discrimination of potential donor sites than previous probabilistic models, but also reveals subtle properties of this signal which suggest aspects of its biochemical function. Acceptor splice signals are modeled using a "windowed" version of the previously developed "weight array model", which is also shown to give significant improvements in discriminative power. Development of a computer program, GENSCAN, which identifies complete exon/intron structures of genes in genomic DNA is described. Novel features of the program include the capacity to predict multiple genes in a sequence, to handle partial as well as complete genes, and to identify consistent sets of genes occurring on either or both DNA strands. The program is also capable of indicating with high accuracy the reliability of each predicted exon. The accuracy of GENSCAN is shown to be substantially better than existing methods when tested on standardized sets of

human and vertebrate genes, with 75 to 80% of exons identified exactly. Consistently high levels of accuracy are observed for sequences of differing C+G content, for primates, rodents and non-mammalian vertebrates, and accuracy is only slightly lower for *Drosophila* and maize sequences. Applications of the program to finding genes in newly sequenced genomic regions and to prediction of alternatively spliced regions of genes are discussed, with examples of each.

Acknowledgments

I gratefully acknowledge John Tillinghast, Luciano Brocchieri, Jan Mràzek, Edwin Blaisdell, Volker Brendel, Zhan-Yang Zhu, Sophia Chen and Nancy Witthaus for their support and encouragement.

Contents

List of Tables

List of Figures

Chapter 1

INTRODUCTION

In recent years, development of the technology for efficient, automated DNA sequencing has led to the accumulation of large databases of DNA and protein sequences, and a new field of study known variously as "computational molecular biology", "mathematical biology" or "bioinformatics" has begun to take shape as researchers work to interpret and draw conclusions from this wealth of new information. Though difficult to define precisely, the field might be described as the area of research at the intersection of molecular biology, molecular evolution and structural biology which seeks to understand the relationships between sequence, structure, evolution and biological function by statistical/computational analysis of molecular sequences. Some of the goals of research in this area include: (i) prediction of protein structure (secondary and/or tertiary) from the primary amino acid sequence; (ii) detection of regulatory signals (promoters, enhancers, origins of replication, etc.) in genomic DNA sequences; and (iii) inferring evolutionary history from comparison of homologous gene or protein sequences (or genomes).

This thesis addresses another significant open problem in this field, namely identification of the precise exon-intron structures of genes in higher eukaryotic (especially human) genomic DNA sequences. The problem has a certain intrinsic interest in that it challenges us to define precisely the sequence dependence of the basic biochemical processes of transcription, translation and RNA splicing, and studies of the sequence properties of known genes may yield clues about the mechanisms of these processes.

On the other hand, with the recent shift in the emphasis of the Human Genome Project from physical mapping to intensive sequencing, the problem has taken on significant practical importance. Indeed, efficient and reliable means of gene detection will be required if the stated goal of identification of all human genes (Watson, 1992) is to be achieved in a timely fashion. The approach taken here is to develop a probabilistic model of gene structure based on studies of properties of known human genes. This model is then applied to the problem of gene identification in a computer program called GENSCAN. The work described is essentially interdisciplinary in nature in that, while the basic subject matter is biological and results of biological interest are obtained, techniques from other fields are used fairly heavily, including certain discrete stochastic models from statistics and dynamic programming algorithms used primarily in electrical engineering applications.

1.1 Background

A large body of literature on the subject of gene prediction has accumulated in the past fifteen years or so. Early studies by Shepherd (1981), Fickett (1982), and Staden & McLachlan (1982) showed that statistical measures related to biases in amino acid and codon usage could be used to approximately identify protein coding regions in genomic sequences. Since then, numerous other compositional differences between coding and non-coding DNA sequences have been noted, including differences in general k-tuple (oligonucleotide) frequencies (e.g., Claverie & Bougueleret, 1986), measures of autocorrelation (Michel, 1986), Fourier spectra (Silverman & Linsker, 1986), purine/pyrimidine periodicity (Arques & Michel, 1990), and local compositional complexity/entropy (Konopka & Owens, 1990). Based on these differences, the first generation of gene prediction programs, designed to identify approximate locations of coding regions in genomic DNA were developed. The most widely known such programs are probably TestCode, based on Fickett's (1982) work, and GRAIL (Uberbacher & Mural, 1991), which uses a neural network approach to integrate multiple types of content statistics in order to classify sequence windows as coding or non-coding. These methods are generally able to identify coding regions of sufficient

length, i.e. at least one or two hundred nucleotides, with fairly high reliability, but do not accurately predict precise exon locations.

In order to more accurately pinpoint exon boundaries, two subsequent generations of algorithms have been developed. Second generation methods, such as SORFIND (Hutchinson & Hayden, 1992), GRAIL II (Xu *et al.*, 1994a), and Xpound (Thomas & Skolnick, 1994), use a combination of splice signal and coding region identification techniques to predict "spliceable open reading frames" (potential exons), but do not attempt to assemble predicted exons into complete genes. Third generation methods attempt the more difficult task of predicting complete gene structures, i.e. sets of exons which can be assembled into translatable mRNA sequences. The earliest examples of such integrated gene finding algorithms were probably the gm program (Fields & Soderlund, 1990) for prediction of genes in *Caenorhabditis elegans* and the method of Gelfand (1990) for mammalian sequences. Subsequently, there has been a mini-boom of interest in development of such methods, and a wide variety of programs have appeared, including (but not limited to): GeneID (Guigó *et al.*, 1992), which uses a hierarchical rule based system to rank potential exons; GeneParser (Snyder & Stormo, 1993, 1995), which uses a combination of neural network and dynamic programming approaches; GenLang (Dong & Searls, 1994), which treats the problem by linguistic methods; FGENEH (Solovyev *et al.*, 1994), which uses discriminant analysis and other statistical techniques; and GAP III (Xu *et al.*, 1994b), which uses dynamic programming to assemble gene models from clusters of potential exons predicted by GRAIL II.

The sheer number of such algorithms raises the obvious question of whether the gene finding problem has perhaps already been solved by one or more of these programs. This question was definitively answered in the negative by a recent systematic comparison of available integrated gene finding methods undertaken by Burset & Guigó (1996). Despite the considerable effort which has been lavished on this problem, the authors concluded that the predictive accuracy of all such methods remains rather low, with most programs identifying less than 50% of exons on average when tested on a standard set of 570 vertebrate multi-exon genes. Another significant shortcoming of existing integrated gene finding methods is that, in general, the

assumption is made that the input sequence contains exactly one complete gene. As a consequence, when presented with a sequence containing multiple genes or a partial gene, some of the predicted exons may be correct, but the assembled gene structure typically does not make sense. This and other limitations of existing methods are discussed in a recent review by Fickett (1996).

1.2 Goals

My primary goal was to develop a fourth generation method of gene identification, which would be capable of predicting the number of genes in a sequence as well as the locations of coding exons, and would be sufficiently general so as to include partial as well as complete genes and genes occurring on either or both DNA strands. At the outset, one can imagine two opposing schools of thought on the issue of gene modeling. First, there is the "pragmatic" or "heuristic" viewpoint: that one should combine all known discriminatory properties of introns, exons, etc. into some sort of composite function for prediction, weighting each property by an appropriate factor derived by trial and error or perhaps by some statistical or machine learning procedure. This approach has been by far the most popular to date, as exemplified by programs like GRAIL (II) and GeneParser, which combine multiple types of content statistics in complicated multi-layer neural networks which bear little resemblance to anything that might be happening in the cell. On the other hand, there is what could be called the "biochemical" viewpoint: that one should construct a model which mimics *in silico* (on the computer) the underlying processes of transcription, RNA splicing and translation which define genes *in vivo*. Aside from the usefulness of the predictions, one might also hope to gain some insight into these processes from such an approach. Unfortunately, a biochemical approach does not yet appear feasible, owing to the extreme complexity of eukaryotic transcription and RNA splicing mechanisms and our currently incomplete understanding of these processes. To my knowledge, no serious attempts have yet been made in this direction: such an undertaking may have to be postponed for at least a few more years.

I decided to take an intermediate path between these two extremes, adopting

what might be called the "probabilistic" viewpoint. Specifically, the model should be a probabilistic description of a gene, in terms of both structural and compositional properties, which should incorporate statistical descriptions of the signals known to be recognized by the general transcriptional, translational and splicing machinery. Thus, the model should attempt to capture the sequence constraints imposed by these biochemical processes but should not attempt to model the processes themselves. Furthermore, the model should be flexible enough so that, as new information is learned about the signals involved in transcription and splicing, improved signal models can be developed and integrated into the existing framework. In addition to the obvious goal of improving predictive accuracy, several additional model properties were considered desireable.

1) All model parameters should be explicit (i.e. no hidden neural network weights), have simple intuitive interpretations, and be estimable from available sets of known human genes.

2) The model should be computationally tractable in the sense that the most likely gene structure or set of structures should be calculable in a reasonable amount of time, say not more than a few minutes for sequences up to 100 kilobases.

3) The model should be capable of assigning a measure of reliability to each predicted exon (or gene) so that, for instance, PCR primers could be designed based on the portions of the prediction which are most certain.

4) The method should be robust with respect to the C+G content of the sequence.[1]

5) Ideally, the method should be capable of predicting whether or not a gene is alternatively spliced and, if so, should predict the exon/intron structure of each alternatively spliced product.

6) The method should be capable of finding new genes as well as genes which are homologous to known proteins.

[1]Most available methods perform less well on A+T rich sequences.

The reasoning behind the last goal was that, since several homology-based gene finding methods already exist, e.g., BLASTX (Gish & States, 1993), GeneID+ (Guigó & Knudsen, unpublished), GeneParser3 (Snyder & Stormo, 1995) and Procrustes (Gelfand *et al.*, 1996), and such methods often work quite well provided that a sufficiently close homolog exists, the really important practical problem is to identify truly novel genes not substantially similar to existing proteins in the databases. Finally, it was decided to work primarily with human sequences, since far more sequence data is available for human than for other higher eukaryotes and because the gene finding problem is notoriously difficult in human genomic DNA.

Many of the goals listed above have been achieved in whole or in part by the current implementation of the GENSCAN program, which is based on a semi-Markov source model of gene structure. The model incorporates several types of features, including splice signal models, exon length distributions, promoter and poly-adenylation signals, etc. of presumed importance for gene function, and accounts for many of the substantial differences in gene density and structure (e.g., intron length) that exist between distinct C+G compositional regions of the human genome. The predictive accuracy of GENSCAN, in particular, is significantly higher than existing methods when tested on standard sets of human and vertebrate gene sequences, and the program is able to exactly reconstruct complex multi-exon gene structures in a substantial proportion of cases. The program is also able to give useful estimates of the reliability of its own predictions, enabling the user to choose predicted exons with any desired degree of certainty. In addition, accuracy is consistently high for sequences of differing C+G content. Finally, in some cases at least, suboptimal exons indicated by the program correspond to alternatively spliced variants of a gene. However, certain features remain difficult to predict including very small exons and exact promoter locations: further improvements in this area will have to be left to the next generation of algorithms.

1.3 Organization

This thesis develops a probabilistic model of gene structure and sequence: the chapters were organized so as to describe the model from its most general structure to its most specific details.[2] Thus, Chapter 2 covers the overall architecture of the model and (the closely related subject of) the algorithms needed to make its use practical. Chapter 3 discusses how the general structural and compositional features of human genes are represented in the model and Chapter 4 describes the particular models of biological signals which were developed, focusing mostly on the acceptor/branch point and donor splice signals. Finally, Chapter 5 covers the specific implementation of the GENSCAN program and how it was tested and gives specific examples of its use. The last chapter briefly reviews some of the unique or unusual features of GENSCAN relative to existing programs and outlines some potential applications of the model architecture to areas beyond gene finding.

A point which should be made at the outset is that, for concreteness, one particular model architecture is described, as if GENSCAN were conceived and implemented in exactly its current form, whereas in reality the model evolved over time and many variations were explored before the final form was decided upon. Thus, in some cases, particular features of the model are described but not fully justified relative to reasonable alternatives because to do so in every case would soon become tedious. In other words, some of the approaches which were tried but failed to improve prediction are not described. Finally, concreteness was desired in order that this thesis would provide a record of the performance of a particular precise model specification at a particular point in time so that further developments in the field of gene identification could be compared to this benchmark. A paper describing the essential features of GENSCAN which overlaps with many areas of this thesis was recently accepted for publication (Burge & Karlin, 1997): further references to this paper are not given since otherwise they would be too numerous.

[2]Though there are several advantages to this organization, there is also the unfortunate disadvantage that some of the most technically difficult sections occur in the second chapter, specifically Sections 2.5 – 2.10. Therefore, some readers may find it more convenient to read these sections after Chapters 3 and 4.

Chapter 2

MODEL ARCHITECTURE AND ALGORITHMS

In this chapter, a probabilistic model of gene structure is introduced and the algorithms necessary for practical use of the model in prediction are developed. The model relies on several standard types of discrete stochastic models[1] which are briefly reviewed in Section 2.1. In Sections 2.2 to 2.5, a basic framework for gene modeling is described, and some of the strengths and limitations of the model structure are addressed. Section 2.6 discusses the serious combinatorial problems involved in identifying complex multi-gene structures in long genomic DNA sequences, and addresses some of the related algorithmic issues. In Sections 2.7 to 2.9, three fundamental algorithms are described which allow efficient determination of the most likely gene structure(s) in a sequence and other quantities of interest. Finally, Section 2.10 gives an explicit example of some of the calculations involved in the optimization (Viterbi) algorithm, which helps to provide insight into how each of the model components contributes to prediction.

[1] A good general reference is Karlin & Taylor (1975)

2.1 Discrete stochastic processes

Consider a discrete time stochastic process (sequence of random variables), $X_1, X_2, ...$ which takes on values from a finite state space $A = \{A_1, A_2, ..., A_N\}$. If the probability of transition from state A_i at time n to state A_j at time $n+1$ depends only on A_i and not on the previous history of the process, then the process is said to have the *Markov* property or to be a Markov Model (MM) or Markov chain. The basic theory of Markov processes is described in Howard (1971a) and elsewhere (e.g., Freedman, 1983). If the transition probabilities depend only on i and j and not on the time, n, the process is said to be (temporally) *stationary* or (time) *homogeneous*. Such a stationary Markov process can be described by an $N \times N$ transition matrix, T, with entries $T_{ij} = P\{X_{n+1} = A_j | X_n = A_i\}$. In general, for a Markov process the probability T_{ii} of returning to the same state in the next time interval may be non-zero. If the transition probabilities depend on n, the process is said to be (temporally) *inhomogeneous*. A particular case of interest here is when the transition probabilities depend only on n modulo m (i.e. the remainder when n is divided by the positive integer m): such a process is referred to here as an m-periodic Markov chain. If instead of depending only on the previous state, the process depends on the previous k states, the process is referred to as a kth-order Markov chain.

A *Semi-Markov Model* (SMM) is a stochastic process whose successive state occupancies are governed by a Markov transition matrix (with the restriction that $T_{ii} = 0$ for all i), but where the duration of time spent in each state is a (positive) integer valued random variable described by a separate probability distribution τ_i which depends on the state type A_i (or, in some cases, on A_{i+1} as well). The theory of semi-Markov processes is described in Howard (1971b). For a Markov process, by contrast, each state is occupied for only a single time unit. The SMM is strictly more general than the MM in the sense that any MM can be represented as an SMM with *geometrically* distributed state lengths (actually, 1-shifted geometric — see below). The geometric distribution is the discrete analog of the well-known exponential distribution and is described by a single parameter, q. A random variable λ is said to have geometric distribution with parameter q if $P\{\lambda = k\} = (1-q)^k q$ for $k = 0, 1, ...,$

corresponding to the probability of k consecutive independent events of probability $1-q$ followed by an event of probability q. It is convenient to define certain trivial variants of the geometric distribution as well. We will say that λ has a "c-shifted" geometric distribution with parameter q if $P\{\lambda = k + c\} = (1-q)^k q$ for $k = 0, 1, ...$, corresponding to a random variable with a minimum value of c, with geometrically decaying probability beyond c.

We now consider a more complex type of model, in which a second stochastic process, $Y_1, Y_2, ...$ which takes on values from a distinct finite state space, $B = \{B_1, B_2, ..., B_M\}$ is generated from the underlying (hidden) stochastic process, X_i, according to probabilistic functions corresponding to each of the state types $A_1, ..., A_N$. If the underlying process is Markov, such a model is called a *Hidden Markov Model* (HMM) or a *Markov Source* (MS). The theory of HMMs, originally developed by Baum and colleagues (e.g., Baum & Petrie, 1966) is reviewed in Rabiner (1989). If the underlying process is semi-Markov, such a model has been referred to as an explicit state duration HMM (Rabiner, 1989) or generalized HMM (Kulp *et al.*, 1996): here we will use the term *Hidden Semi-Markov Model* (HSMM) or *Semi-Markov Source* (SMS) to more clearly distinguish it from the simpler HMM/MS class of models.

2.2 Choice of model structure

2.2.1 Overview

Given a genomic DNA sequence of length L, the goal is to classify each of the L sequence positions as to its functional state: exon, intron, intergenic, 5′ untranslated region (UTR), etc. Of course, we would also like to know for each state whether it occurs on the forward or reverse (complementary) DNA strand and, for exons, the reading frame, so that the encoded amino acid sequence can be derived. To put this in a probabilistic framework, the approach taken here is to imagine that an underlying (random) process generates a series of functional states, e.g., 5′ UTR, exon, intron, exon, 3′ UTR, ... which in turn generate the DNA sequence according to probabilistic models of each of the states. For prediction, then, we must solve the inverse problem:

given a sequence, infer the series of states which most likely gave rise to it.

2.2.2 A previous model

Before describing the model structure in detail, a previous probabilistic approach to gene prediction developed by Haussler and colleagues (Krogh *et al.*, 1994) is briefly reviewed. These authors describe a method for identifying genes in *Escherichia coli* genomic DNA using an HMM with four basic states: Coding (C), Intergenic (I), Start (S) and Terminate (T). In this very simple model, the Start state generates one of the two initiation codons used by prokaryotes (ATG or GTG); the Terminate state generates one of the three stop codons (TAA, TAG or TGA); the Coding state generates one of the 61 non-termination codons; and the Intergenic state generates one of the four nucleotide bases. The states themselves are generated according to an underlying Markov chain which enforces the obvious biological constraints on the series of states generated: Start is always followed by Coding, Coding is followed by Coding or Terminate, Terminate is always followed by Intergenic, and Intergenic is followed by Intergenic or Start. Notice that under such a model, the length of intergenic regions (strings of consecutive Intergenic states) will have a (1-shifted) geometric distribution with parameter $q_I = 1 - T_{II}$, i.e. $P\{\lambda = k + 1\} = q_I(1 - q_I)^k$, where λ is the length in base pairs and T_{II} is the Markov transition probability for an I state to be followed by another I state. Similarly, the number of codons in a coding region (string of consecutive Coding states) will also be distributed geometrically, with parameter $q_C = 1 - T_{CC}$. In fact, an important limitation of HMMs is that no matter what the structure of the model or number states, the lengths of consecutive runs of any state type are always geometrically distributed.

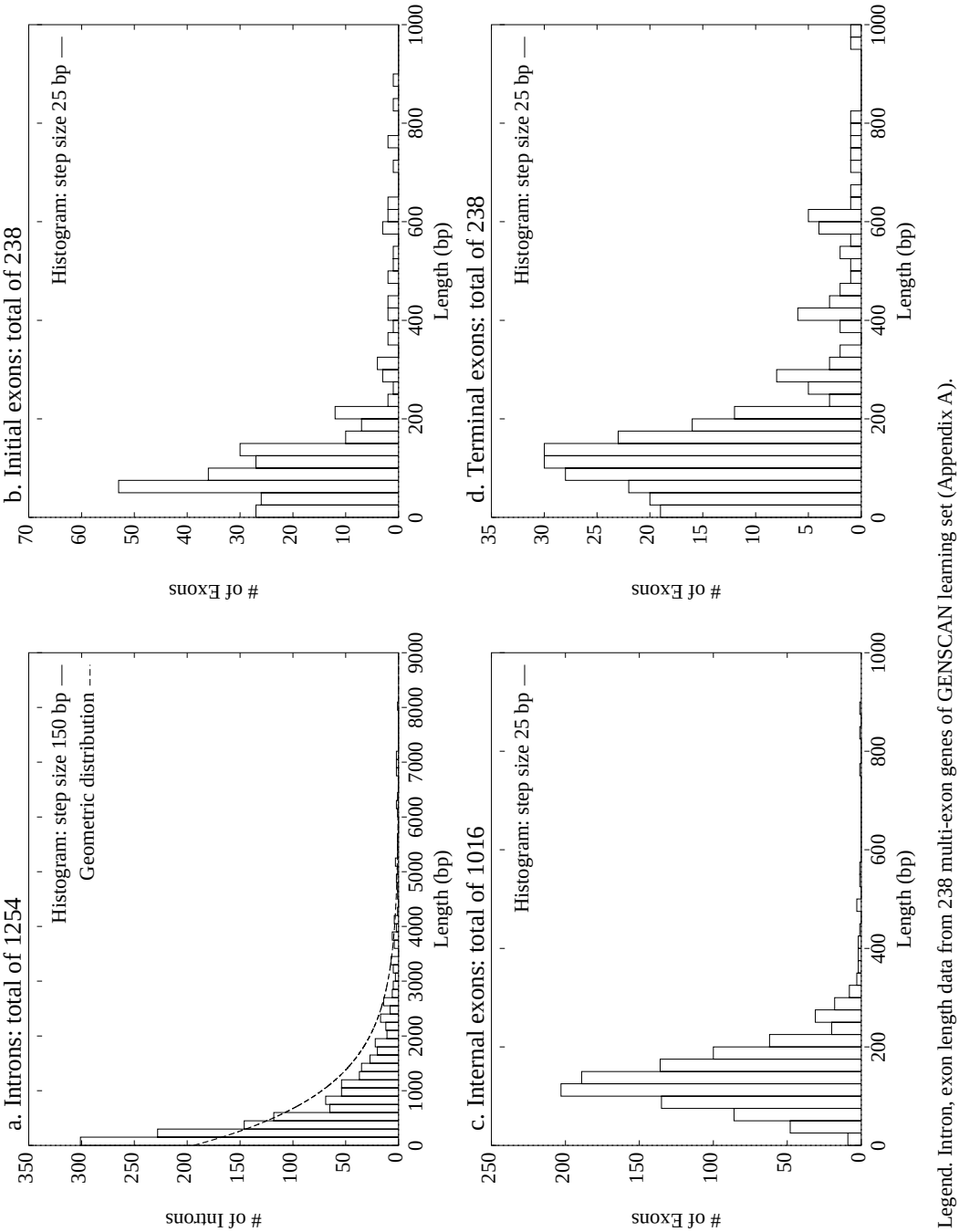
2.2.3 Length distributions of exons and introns

While the geometric distribution of lengths is in fact fairly general, arising whenever a discrete stochastic process has the "memoryless" property (e.g., Karlin & Taylor, 1975), there is no obvious reason to expect the length distributions of gene components to necessarily have this form. Fig. 1 displays the distributions of the lengths of

introns and initial, internal and terminal exons derived from the nonredundant set of 238 human multi-exon genes described in Section 3.1 and Appendix A. For introns, the observed distribution does in fact appear approximately geometric, as would be expected in the absence of significant functional constraints on intron length, and there is some experimental evidence relating to this issue. For example, for rabbit β-globin, intron length was observed to be unimportant for splicing provided that a certain minimum threshold of about 70 to 80 nucleotides was exceeded (Wieringa *et al.*, 1984). The observed distribution of intron lengths (Fig. 1a) tends to support this idea: no introns less than 65 bp were observed, but above this size the distribution appears to be approximately geometric.

For exons, on the other hand, length does appear to be an important property for 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 constitutively recognized internal exons to sizes below about 50 bp may lead to exon skipping, i.e. failure to include the exon in the final processed mRNA (Dominski & Kole, 1991), and there is some evidence that steric interference between factors recognizing splice sites may make splicing of small exons more difficult (e.g., Black, 1991). Of course, a number of small exons exist and are efficiently spliced, so any such limitation cannot be absolute. At the other end, there is some evidence that spliceosomal assembly is inhibited if internal exons are internally expanded beyond about 300 nucleotides, e.g., Robberson *et al.* (1990), but conflicting evidence also exists (Chen & Chasin, 1994), and recent results (Sterner *et al.*, 1996) have suggested that the situation may be more complicated, involving the lengths of adjacent introns as well. Overall, though, most results have tended to support the idea that "medium-sized" internal exons (between about 50 and 300 bp in length) may be more easily spliced than excessively long or short exons. This idea is given substantial support by the observed distribution of internal exon lengths (Fig. 1c), which shows a pronounced peak at around 120-150 nucleotides, with few internal exons more than 300 bp or less than 50 bp in length (see also Hawkins, 1988 for a previous tabulation of exon and intron lengths). Initial (Fig. 1b) and terminal (Fig. 1d) exons also have substantially peaked distributions (possibly multi-modal) but don't exhibit such a steep dropoff in density after 300 bp,

Fig. 1. Length distributions of introns and exons in human genes



Legend. Intron, exon length data from 238 multi-exon genes of GENSCAN learning set (Appendix A).

suggesting that somewhat different constraints may exist for splicing of exons at or near the ends of the pre-mRNA. In order to be able to accurately account for these potentially important properties of exon length, I chose to use an underlying semi-Markov architecture rather than the simpler Markov structure used previously. Such a framework is capable of describing many of the basic structural features of genes, e.g., the typical number of exons per gene, typical exon/intron length distributions, etc. and yet still remains simple enough so as to be computationally tractable.

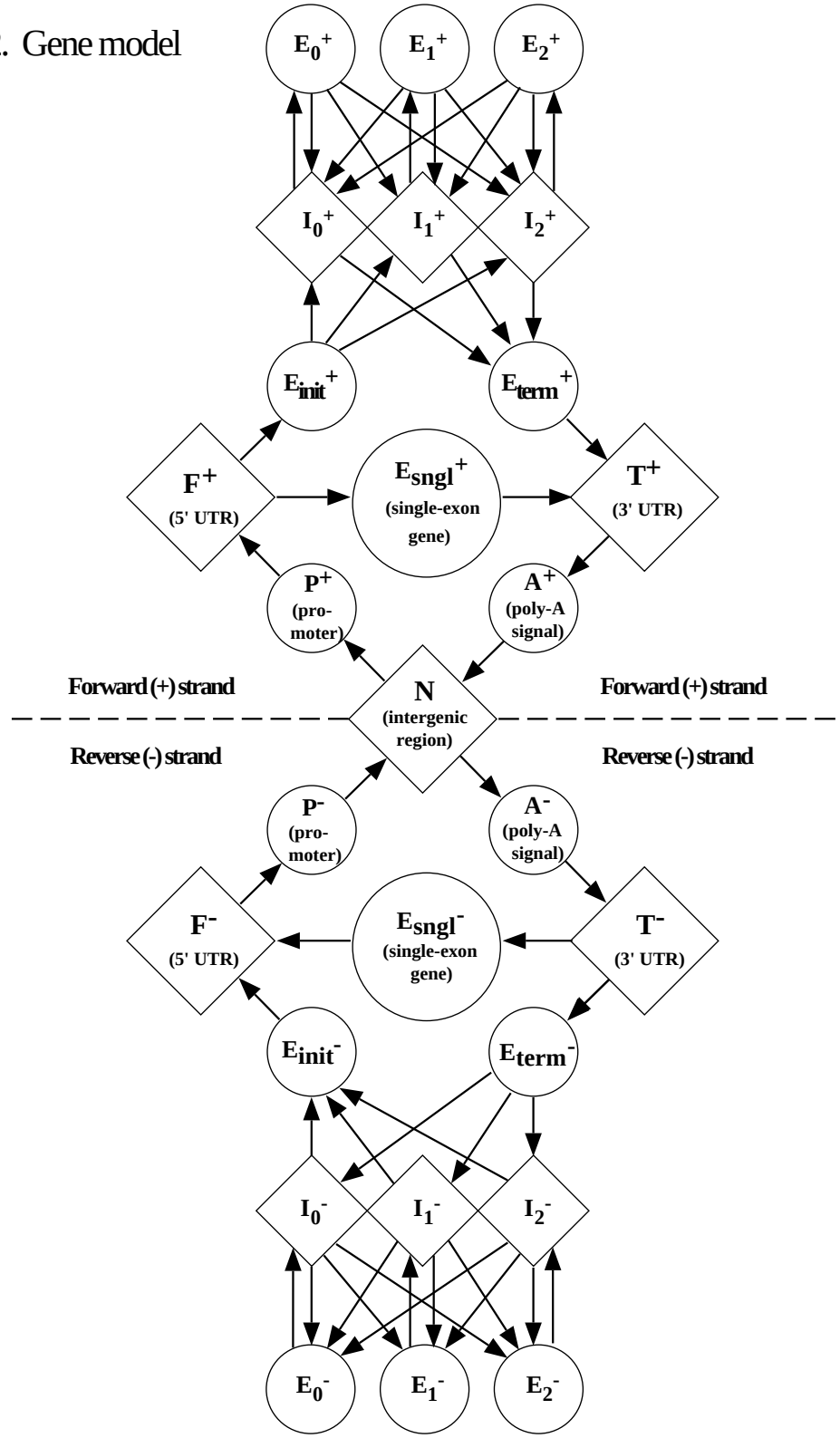
2.3 Model architecture

2.3.1 States of the model

The model structure employed is illustrated in Fig. 2. Each circle or diamond in the figure represents a particular functional element type (state) of a gene or genomic region, namely: N, intergenic region; P, promoter; F, $5'$ untranslated region (extending from the start of transcription to just before the translation initiation signal); E_{sngl}, single-exon gene (encompassing the translation initiation, coding region and translation termination signals); E_{init}, initial exon (comprising the translation initiation, coding region and donor splice signals); E_k $(0 \leq k \leq 2)$, phase k internal exon (acceptor splice signal, coding region, and donor splice signal); E_{term}, terminal exon (acceptor splice signal, coding regions, translation termination signal); T, $3'$ untranslated region (extending from just after the translation termination signal to the poly-adenylation signal); A, poly-adenylation signal; and I_k $(0 \leq k \leq 2)$, phase k intron (extending from just after the end of the donor splice signal to just before the branch point/acceptor splice signal). Note that, in order to keep the number of states manageable, translation initiation/termination signals and donor/acceptor splice signals have been included as subcomponents of the associated exon state. The total number of state types in this model is $N = 27$.

In higher eukaryotes, where coding sequences are typically disrupted by one or more introns, it is necessary to have some device to maintain a consistent reading

Fig. 2. Gene model



frame across the gene, since otherwise predicted exons will not always be assemblable into a translatable mRNA. Interestingly, the maintenance of reading frame along the mRNA is not necessarily a completely artificial consideration arising only in computer models of gene structure, but may in fact resemble constraints existing in the cell. Specifically, while the splicing machinery itself presumably does not make use of reading frame information, there is some experimental evidence, reviewed in Maquat (1995), that eukaryotic cells may accomplish this task by selectively degrading improperly spliced mRNAs or mRNAs derived from pseudogenes which contain premature termination codons. In the model described here, the reading frame is kept track of by dividing introns and internal exons according to "phase": thus, an intron which falls between codons is considered phase 0; after the first base of a codon, phase 1; and after the second base of a codon, phase 2. Internal exons are similarly divided according to the phase of the *previous* intron, which determines the codon position of the first base pair of the exon, hence the reading frame.

All states within the transcription unit are also divided according to DNA strand: the upper half of the figure corresponds to the states (designated with a superscript "+") of a gene on the forward strand, while the lower half (designated with superscript "–") corresponds to a gene on the opposite (complementary) strand. For example, proceeding in the $5'$ to $3'$ direction on the (arbitrarily chosen) forward strand, the components of an E_k^+ (forward-strand internal exon) state will be encountered in the order: acceptor site, coding region, donor site, while the components of an E_k^- (reverse-strand internal exon) state will be encountered in the order: inverted complement of donor site, inverted complement of coding region, inverted complement of acceptor site. Only the intergenic state N is not divided according to strand. The model structure is thus capable of describing the essential gene organization of most vertebrate genomic sequences likely to be encountered — some limitations are discussed in Section 2.4.

2.3.2 Transitions between states

Successive states may occur in any biologically consistent order, as represented by the arrows in Fig. 2: in the model they are generated according to the semi-Markov

process described below. Specifically a "parse", ϕ, is generated consisting of an ordered set of states, $\vec{q} = \{q_1, q_2, ..., q_n\}$, with an associated set of lengths (durations), $\vec{d} = \{d_1, d_2, ..., d_n\}$ which, using probabilistic models of each of the state types, generates a DNA sequence S of length $L = \sum_{i=1}^{n} d_i$. The parse is thus a complete description of the precise locations of all coding exons and other functional features in the sequence (e.g., untranslated regions, promoter and poly-adenylation signals). The generation of a parse corresponding to a (pre-defined) sequence length L is as follows:

1) An initial state q_1 is chosen according to an initial distribution on the states, $\vec{\pi}$, i.e. $\pi_i = P\{q_1 = Q_i\}$, where $Q_0, Q_1..., Q_{26}$ is an indexing of the 27 state types.

2) A length (state duration), d_1, corresponding to the state q_1 is generated conditional on the value of q_1 from the length distribution f_{q_1}.

3) A sequence segment s_1 of length d_1 is generated, conditional on d_1 and q_1, according to the sequence generating model P_{q_1}, corresponding to state type q_1.[2]

4) The subsequent state q_2 is generated, conditional on the value of q_1, from the (Markov) state transition matrix T, i.e. $T_{ij} = P\{q_{k+1} = Q_j | q_k = Q_i\}$.

5) This process is repeated until the sum, $\sum_{i=1}^{n} d_i$, of the state durations first equals or exceeds the length L, at which point the last state duration d_n is appropriately truncated, the final stretch of sequence is generated, and the process stops. The sequence generated is simply the concatenation of the sequence segments, $S = s_1 s_2 ... s_n$.

One slight modification to this sequence of steps has to be made to ensure that exon lengths generated are compatible with the phases of adjacent intron states. Specifically, exon lengths are generated in two steps: first, the number of complete codons is generated from the appropriate length distribution; then the appropriate

[2]The dependence on the length d_1 is not explicitly indicated in order to simplify the notation.

number (0, 1 or 2) of bp is added to each end to account for the phases of the preceding and subsequent states, i.e. step 4) precedes steps 2) and 3) for exon states. For example, if the number of complete codons generated for an initial exon is c and the phase of the subsequent intron is k, then the total length of the exon is: $\lambda = 3c+k$.

The model thus has four main components: a vector of initial probabilities $\vec{\pi}$, a matrix of state transition probabilities T_{ij}, a set of length distributions f_i, and a set of sequence generating models P_i. These model components are described in detail in Chapters 3 and 4. The remainder of this chapter is devoted to a discussion of certain more general issues, specifically how the model is used for prediction and how the resulting combinatorial problems can be solved, subjects which are largely independent of the precise choice of the model components.

2.4 Limitations

Of course, real genes are not generated by any sort of Markov or semi-Markov process. The real mechanisms by which genes are created are undoubtedly far more complex, involving events such as gene duplication and various types of mutations, e.g., point mutations, insertions, deletions, inversions, rearrangements, possibly "exon shuffling", etc., and the whole process occurs under the guidance of natural selection. The semi-Markov assumption, in particular, limits the types of dependencies between gene components which can be described. Specifically, the model can treat interactions between adjacent state types (introns/exons) in a gene, but cannot deal with "long range" dependencies between widely separated functional elements of a gene. Though such dependencies may exist, the model structure arguably provides a reasonable first approximation to gene structure which might be extended or elaborated as more complex types of dependencies between gene components come to light.

The model structure is also limited in terms of the types of functional sequences described. Specifically: (i) only protein-coding genes are treated (and not tRNA or rRNA genes, for example); (ii) only introns occurring within the translation unit are considered (and not those occurring in $5'$ or $3'$ untranslated regions); (iii) overlapping transcription units are not considered; (iv) certain types of regulatory elements

(e.g., enhancers) are not represented; and (v) signals related to alternative splicing are not included (but see Section 5.5 where an application of the model to prediction of alternative splicing is described). Nor, of course, does the model attempt to mimic the processes by which genes are transcribed, spliced and translated, although signals relating to these processes are incorporated. Nevertheless, as will be seen in Chapter 5, even such an approximate description of gene structure provides a very useful framework for gene prediction, and the resulting program, GENSCAN, is capable of reconstructing highly complex multi-exon gene structures in many cases.

2.5 Prediction

Assuming for the moment that the four basic components of the model described in Section 2.3.2 have been specified, the model can be used for prediction in the following way. For a fixed sequence length L, consider the space $\Omega = \Phi_L \times \mathcal{S}_L$, where Φ_L is the set of (all possible) parses of length L and $\mathcal{S}_L$ is the set of (all possible) DNA sequences of length L. The model M can then be thought of as a probability measure on this space, i.e. a function which assigns a probability mass (density) to each parse/sequence pair. Specifically, the joint probability $P\{\phi_i, S\}$, of generating a specific parse $\phi_i \in \Phi_L$ and a specific sequence $S \in \mathcal{S}_L$ is given by:

[1] $$P\{\phi_i, S\} = \pi_{q_1} f_{q_1}(d_1) P_{q_1}(s_1) \prod_{k=2}^{n} T_{q_{k-1},q_k} f_{q_k}(d_k) P_{q_k}(s_k)$$

where the states of ϕ_i are $q_1, q_2, ..., q_n$ with associated state lengths $d_1, d_2, ..., d_n$, which break the sequence into segments $s_1, s_2, ..., s_n$. The conditional probability of the parse ϕ_i *given* the sequence S can then be calculated (Bayes' Rule) as:

[2] $$P\{\phi_i|S\} = \frac{P\{\phi_i, S\}}{P\{S\}} = \frac{P\{\phi_i, S\}}{\sum_{\phi_j \in \Phi_L} P\{\phi_j, S\}}$$

The basic assumption is that if the model accurately reflects the biological constraints on gene sequence/structure, then the parse or parses with highest likelihood (conditional probability) should correspond closely to the correct gene structure(s). One

way of looking at the prediction problem for such a model is to think of the sequence S as being the observable manifestation of the underlying (hidden) series of states, $q_1, q_2, ...$ which we would like to infer.

2.6 Algorithmic issues

2.6.1 Combinatorial explosion of gene structures

To use such a model for gene prediction, we must have some means of determining which of the many possible gene structures (involving any valid combination of states/lengths) have highest likelihood for a given sequence. Just how big is this search space? To address this question, a simple computer experiment was performed in which the number of potential exons meeting the minimal constraints of the state models were counted in a set of human genomic sequences. Specifically, the constraints were that potential exons should begin with ATG or a minimal acceptor site (AG), end with a stop codon or minimal donor site (GT), and have no in-frame stop codons (for potential internal exons, the three possible reading frames are treated separately). The results were that, for sequences of a few kb or longer, the number of potential exons grows roughly linearly with sequence length, and is typically comparable to the number of base pairs in the sequence (data not shown). The implication is that the number of possible multi-exon gene structures, involving compatible combinations of potential exons, will grow approximately *exponentially* with sequence length, leading to a combinatorial explosion for even moderate length sequences.

A recent paper by Wu (1996) addressed this question by explicitly calculating how many possible multi-exon gene structures were minimally compatible in terms of initiation codon, open reading frame / stop codon and the same minimal splice site constraints used above, with each of a set of vertebrate genomic sequences. The results were that, as expected, the number of possible gene structures grows approximately exponentially with sequence length, and that even for sequences as short as 10 kilobases, the number of possible structures typically exceeds 10^{100} (more than the

number of atoms in the solar system). Thus, it is utterly impossible to explore all possible gene structures in a sequence, and if the model structure is to be of practical use for gene prediction, efficient searching algorithms are required.

2.6.2 Quantities to be computed

It is of primary interest to solve the following three problems:

(i) *Partition function.* Calculate the quantity $P\{S\} = \sum_{\phi_j \in \Phi_L} P\{\phi_j, S\}$. This quantity, corresponding to the partition function Z in statistical physics, allows us to interconvert joint and conditional probabilities using eq. [2] and has other important uses.

(ii) *Exon probabilities.* For a potential exon ϵ, calculate the quantity $P\{\epsilon|S\}$, i.e. the probability that the exon is correct (part of a gene), given the sequence. For the model used here, this involves summing over all parses (potential gene structures) which contain the exon in the correct reading frame.

(iii) *Optimization.* Find the parse (gene structure description) which has highest conditional probability given the sequence.

It will be seen that these three problems are closely related and that similar types of algorithms may be used to solve each of them. Some brief calculations described below for the optimization problem serve to motivate two key approximations which are made in order to reduce the computational complexity of these problems.

2.6.3 Algorithmic complexity of the optimization problem

For a Markov source (HMM) model, the optimization problem is efficiently solved by the Viterbi algorithm (Viterbi, 1967, Forney, 1973), requiring on the order of $N^2 L$ computations, where N is the number of state types in the model and L is the length of the sequence. In practice, assuming that a current computer workstation can perform about 10^8 computations per second, the amount of time required to process a 100 kb sequence with a 27-state HMM would be on the order of $27^2 \times 10^5 / 10^8 = 0.73$

seconds – certainly fast enough for practical use. For a semi-Markov source (HSMM) model, on the other hand, a more complex algorithm is required (e.g., Rabiner, 1989, Section IV D), involving a recursion which must at each position search back over all previous positions, and the number of computations increases to approximately $N^2L^3/2$, i.e. it grows cubically with sequence length rather than linearly. Thus, the time to process a 100 kb sequence with a general 27-state HSMM model would be about: $27^2 \times (10^5)^3/(2 \times 10^8) = 3.6 \times 10^9$ seconds, or about 115 years! Obviously, the model and/or algorithm must be simplified in some way to make this approach practical.

In applications of semi-Markov source models to speech recognition (e.g., Levinson, 1986), this computational problem has been dealt with by assuming that state durations (which correspond to spoken words or syllables in these models) are at most a fixed number, D, units long. With this constraint, the number of computations is reduced to approximately $N^2D^2L/2$, which is practical for small enough values of D. However, setting limits on state durations is not a reasonable simplification in the context of gene modeling since some of the states, particularly those corresponding to intron and intergenic regions, can be almost arbitrarily long in human genomic sequences. For example, intron 17 in the human retinoblastoma gene is more than 71 kilobases in length, and even longer introns are known.

2.6.4 Two simplifying assumptions

I have chosen a different approach (which, to my knowledge, has not been used previously) to reducing the computational complexity of the problem, which is to assume that the length distributions and sequence generating models for a certain subset of the state types have a particular form. Specifically, states of the class $\mathcal{D} = \{N, F^+, F^-, T^+, T^-, I_0^+, I_1^+, I_2^+, I_0^-, I_1^-, I_2^-\}$ (represented as diamonds in Fig. 2), which can apparently be almost arbitrarily long in human genes, are assumed to have: 1) geometric length distributions; and 2) sequence generating models which are "factorable", i.e. such that $P_i(S_{a,c}) = P_i(S_{a,b})P_i(S_{b+1,c})$, where a, b, c are sequence positions with $1 \leq a \leq b < c \leq L$, and $S_{x,y}$ represents the sequence segment from position x to y inclusive. Under these assumptions, for any type-$\mathcal{D}$ state, the joint

probability of generating the sequence segment $S_{a,b+1}$ differs from that of generating $S_{a,b}$ by a constant factor[3], $p_i P_i(S_{b+1})$, independent of a. This property allows the recursions needed to solve the three problems described in Section 2.6.2 to be written in a particularly simple form, as will be seen below. The remaining types of states, designated $\mathcal{C}$ (represented as circles in Fig. 2), are still treated using general length distributions and sequence generating models, as described in Chapters 3 and 4. In the worst case, for a sequence which contains arbitrarily long open reading frames, the number of computations grows quadratically with sequence length; for virtually all real sequences, however, these approximations result in run times which grow only linearly with sequence length.

2.7 The partition function

2.7.1 Preliminaries

Before describing the algorithms, a convention regarding states which extend off the edges of the sequence must be adopted. The problem is how to treat, for example, a potential E^+_{term} state ending with a stop codon at position $97, 98, 99$ of the sequence and beginning at some unspecified position prior to the beginning of the sequence. Such a state is difficult to evaluate under the model since we do not know the exact length of the potential exon, whether or not there is an appropriate acceptor splice site sequence, etc. The expedient used here is to assume that the exon begins exactly at the boundary of the sequence (position 1) so that position 0 would correspond to the preceding type-$\mathcal{D}$ state, in this case an I^+_0 (intron phase 0) state.[4] A similar assumption is made for all potential type-$\mathcal{C}$ states which extend to the sequence boundaries so that, in particular, positions 0 (immediately before the sequence) and $L + 1$ (immediately after the sequence) will always correspond to type-$\mathcal{D}$ states. Examination of Fig. 2 shows that all allowed state transitions are from type-$\mathcal{D}$ states

[3] S_x represents the nucleotide at position x of the sequence.

[4] Of course, in interpreting the program results, one does not assume that predicted exons which extend to the edge of the sequence necessarily end at exactly this point — this assumption is made merely to simplify the algorithm description.

to type-$\mathcal{C}$ states or vice versa, but never between two type-$\mathcal{D}$ states or two type-$\mathcal{C}$ states. Thus, if we begin at position 0 and end at $L+1$, any valid parse of the sequence will consist of an alternating series of states of types: $\mathcal{D},\mathcal{C},\mathcal{D},\mathcal{C},\mathcal{D},...,\mathcal{C},\mathcal{D}$, comprising M type-$\mathcal{C}$ states and $M+1$ type-$\mathcal{D}$ states. A practical consequence of this assumption is that only eleven variables, corresponding to the type-$\mathcal{D}$ states, rather than 27, corresponding to all state types, are required in the recursions described below.

All of the recursions described require construction of a list (array) L_j at each position j in the sequence, representing the set $\{\epsilon_0, \epsilon_1, ..., \epsilon_{m_j-1}\}$ of all potential type-$\mathcal{C}$ states ending exactly at position j which have non-zero probability. With each such state ϵ_k are associated the following properties (also stored in arrays): a_k, the beginning (first nucleotide position) of the state; b_k $(= j)$, the end (last nucleotide position) of the state; λ_k $(= b_k + 1 - a_k)$, the length of the state; y_k, the state type; x_k, the previous state type; and z_k, the subsequent state type. For example, if there is a GT dinucleotide (minimal donor site) at positions $j+1, j+2$ of the sequence, and an ATG trinucleotide beginning at a position $i < j-2$, then the list L_j will contain a potential initial exon ϵ_k, beginning at position[5] $a_k = i$, ending at $b_k = j$, of length $\lambda_k = j+1-i$. The state type y_k in this example is E^+_{init}, the previous state type x_k is F^+ (5′ UTR), and the subsequent state type z_k is I^+_h, where h is the phase of the subsequent intron, which can be calculated from the exon length as: $h = \lambda_k \bmod 3$. Notice in particular (Fig. 2) that, given the length of the state, each type-$\mathcal{C}$ state type has a unique previous and subsequent state type, so the variables x_k and z_k are always uniquely defined. In the recursion descriptions, the type-$\mathcal{D}$ state types are indexed $Q_0, ..., Q_{10}$, with geometric length parameters $g_0, ..., g_{10}$, respectively, and complements $p_i = 1 - g_i$ for $0 \leq i \leq 10$.

[5]Actually, the exon states as previously defined extend from a few bases before i to a few bases after j since the translation initiation/acceptor and termination signal/donor splice signals are included as part of the exon state. For convenience, only the actual exon boundaries (i, j) will be referred to from here on, with the understanding that the state endpoints are slightly different.

2.7.2 The forward algorithm

The partition function, $Z = P\{S\} = \sum_{\phi_j \in \Phi_L} P\{\phi_j, S\}$, can be calculated using what is known as a "forward" algorithm (e.g., Rabiner, 1989). In this approach, variables $\alpha_i(j)$ are defined which store the *sum* of the joint probabilities of all parses of the *subsequence* $S_{1,j}$ which end in state type Q_i at position j. The key observation allowing calculation of this enormous sum in a reasonable number of steps is that the $\alpha_i(j)$ variables can be updated recursively, as given below.

Initialization:

[3a] $$\alpha_i(1) = \pi_i p_i P_i(S_1), \qquad 0 \leq i \leq 10.$$

Induction:

[3b] $$\alpha_i(j+1) = \alpha_i(j) p_i P_i(S_{j+1}) \quad +$$

$$\sum_{\epsilon_k \in L_j, z_k = i} \alpha_{x_k}(a_k - 1)(1 - p_{x_k}) T_{x_k,y_k} f_{y_k}(\lambda_k) P_{y_k}(S_{a_k,j}) T_{y_k,z_k} p_i P_i(S_{j+1}),$$

$$0 \leq i \leq 10, \quad 1 \leq j \leq L-1.$$

Termination:

[3c] $$\alpha_i(L+1) = \alpha_i(L) \quad +$$

$$\sum_{\epsilon_k \in L_L, z_k = i} \alpha_{x_k}(a_k - 1)(1 - p_{x_k}) T_{x_k,y_k} f_{y_k}(\lambda_k) P_{y_k}(S_{a_k,L}) T_{y_k,z_k}, \qquad 0 \leq i \leq 10.$$

Note that the recursion is carried out up to position $L+1$, to allow for exon/signal states which extend to the edge of the sequence as discussed previously. Finally, since all parses must end in one of the eleven type-$\mathcal{D}$ states, the partitition function is given by: $P\{S\} = \sum_{i=0}^{10} \alpha_i(L+1)$. The key simplification resulting from the two assumptions described in the previous section occurs in the induction step, eq. [3b]. Specifically, note that the probability of *all* parses ending in state i at position $j+1$ which were

in state i at the previous position can be calculated simply as $\alpha_i(j)p_iP_i(S_{j+1})$. A problem which arises in the practical implementation of this algorithm is that the $\alpha_i(j)$ variables tend to zero as j increases, and may fall below the limits of the precision of the computer. This issue may be handled in a number of ways, e.g., by periodically rescaling the $\alpha_i(j)$ (multiplying by a large constant), which can later be divided out when the quantity $P\{S\}$ is to be used.

2.8 Exon probabilities

We now consider the event ϵ, that a particular exon state (from one of the lists L_j) is correct, i.e. is part of a gene in the correct reading frame. Under the model, this event has probability

[4] $$P\{\epsilon|S\} = \frac{P\{\epsilon, S\}}{P\{S\}} = \frac{\sum_{\phi_j:\epsilon\in\phi_j} P\{\phi_j, S\}}{P\{S\}}$$

where the sum is taken over all parses in Φ_L which contain the exact exon ϵ as a component. As in the case of the partition function problem, this sum can involve an enormous number of potential parses making solution by exhaustive enumeration impractical. However, it is possible to perform this calculation more efficiently, using the the previously described forward algorithm combined with an analogous "backward" algorithm (e.g., Rabiner, 1989).

2.8.1 The backward algorithm

In this approach, variables $\beta_i(j)$ are defined which store the *sum* of the joint probabilities of all parses of the *subsequence* $S_{j,L}$ which are in state Q_i at position j. Again, these variables can be updated recursively, in this instance beginning from the end of the sequence and proceeding backwards.

Initialization:

[5a] $$\beta_i(L+1) = 1, \qquad 0 \le i \le 10.$$

Induction:

[5b]
$$\beta_i(j) = p_i P_i(S_j)\beta_i(j+1) \quad +$$
$$\sum_{\epsilon_k \in \hat{L}_{j+1}, x_k = i} p_i P_i(S_j) T_{x_k,y_k} P_{y_k}(S_{j+1,b_k}) f_{y_k}(\lambda_k) T_{y_k,z_k}(1-p_{z_k})\beta_{z_k}(b_k+1),$$
$$0 \le i \le 10, \quad 2 \le j \le L.$$

Here, $\hat{L}_j$ is analogous to the previously described list L_j, but instead represents the list of potential type-$\mathcal{C}$ states which *begin* at position j. No termination step is required, since for our purposes only the intermediate values of $\beta_i(j)$ are needed. Again, measures must be taken to ensure that the $\beta_i(j)$ variables remain within the precision limits of the computer.

2.8.2 The forward-backward formula

Consider a potential exon ϵ beginning at a and ending at b (of length $\lambda = b+1-a$). If the exon is of type y, and the preceding and subsequent states are of types x and z, respectively, then the desired conditional probability $P\{\epsilon|S\}$ can be calculated using the intermediate values of the forward and backward algorithms and the previously calculated value of the partition function as:

[6]
$$P\{\epsilon|S\} = \frac{\alpha_x(a-1)(1-p_x)T_{x,y} f_y(\lambda) P_y(S_{a,b}) T_{y,z}(1-p_z)\beta_z(b+1)}{P\{S\}}$$

Here, $\alpha_x(a-1)$ captures the probabilities of all "left-parses" of $S_{1,a-1}$ which end in the appropriate state (e.g., intron or 5′ UTR) immediately before the exon, and $\beta_z(b+1)$ captures the probabilities of all "right-parses" of $S_{b+1,L}$ which begin in the appropriate state (e.g., intron or 3′ UTR) immediately after the exon: multiplying these quantities corresponds to summing the probabilities of all possible compatible pairings of a left-parse with a right-parse. This general approach to calculation has been refered to as the "forward-backward" procedure (e.g., Rabiner, 1989 — see also Stormo & Haussler,

1994) for obvious reasons. It is worthy of note that the probability $P\{\epsilon|S\}$ introduced here is an intrinsically *non-local* property of the sequence, since it derives not only from features of the sequence segment from a to b, but also from potentially distant portions of the sequence through its dependence on the joint probabilities of parses of the sequence from 1 to $a-1$ and from $b+1$ to L. Thus, for example, the probability of a potential initial exon at $[a, b]$ would in general be increased by the presence of a strong consensus promoter signal at an appropriate distance upstream of a, since this would tend to increase the joint probabilities of parses ending in the F ($5'$ UTR) state at $a-1$. This probability provides a very useful measure of the reliability of predicted exons, as will be seen in Chapter 5. It is also noteworthy that the non-locality of this quantity makes it fundamentally different from typical exon "scores" derived in other gene prediction programs — GRAIL, GeneParser, etc., which typically depend only on local properties of the exon such as splice signals, codon or hexamer composition, and so forth.

2.9 Optimization and "suboptimization"

2.9.1 The Viterbi algorithm

As mentioned previously, an efficient method known as the Viterbi algorithm exists for finding the optimal parse in the case of a Markov source model. In this approach, variables $\gamma_i(j)$ are defined which store the joint probability of the *optimal* (highest probability) parse of the *subsequence* $S_{1,j}$ which ends in state Q_i at position j. These variables can be calculated recursively as follows.

Initialization:

[7a] $$\gamma_i(1) = \pi_i P_i(S_1) p_i, \qquad 0 \leq i \leq 10.$$

Induction:

[7b] $$\gamma_i(j+1) = \max \quad \{ \quad \gamma_i(j) p_i P_i(S_{j+1}),$$

$$\max_{\epsilon_k \in L_j, z_k = i} \quad \{ \quad \gamma_{x_k}(a_k - 1)(1 - p_{x_k}) T_{x_k,y_k} f_{y_k}(\lambda_k) P_{y_k}(S_{a_k,j}) T_{y_k,z_k} p_i P_i(S_{j+1}) \quad \} \quad \},$$

$$0 \leq i \leq 10, \quad 1 \leq j \leq L-1.$$

Termination:

[7c] $$\gamma_i(L+1) = \max \quad \{ \quad \gamma_i(L),$$

$$\max_{\epsilon_k \in L_L, z_k = i} \quad \{ \quad \gamma_{x_k}(a_k - 1) T_{x_k,y_k} f_{y_k}(\lambda_k) P_{y_k}(S_{a_k,L}) T_{y_k,z_k} \quad \} \quad \}, \qquad 0 \leq i \leq 10.$$

Since any parse must end in one of the eleven type-$\mathcal{D}$ state types, the probability of the optimal parse is given by: $P\{\phi_{opt}, S\} = \max_i\{\gamma_i(L+1)\}$. Note that the algorithm is almost identical in form to the forward algorithm, except that maxima are taken at each step rather than sums. Again, the recursion is carried out up to position $L+1$, to allow for exon/signal states which extend to the edge of the sequence as discussed previously. Whenever a maximum is taken, the locations and nature of the transitions between states which gave rise to the maximum value are recorded in a separate array. The exact series of states in ϕ_{opt} can then be reconstructed by a standard backtracking procedure (e.g., Rabiner, 1989), essentially searching back through this array to recover the sequence of transitions which led to the optimal probability, which gives the most likely parse (gene structure description) in the sequence.

2.9.2 Computational complexity of algorithms

Examining the induction steps in the three algorithms described (forward, backward, and Viterbi), it can be seen that all have essentially the same complexity, which is on the order of the number of potential type-$\mathcal{C}$ states in the sequence which have non-zero probability. In practice, this number remains manageable since potential exons lacking minimal splice site constraints (acceptor: AG, donor: GT), initiation (ATG) or termination (stop codon) constraints, or having in-frame stop codons have probability zero under the model (see Chapters 3 and 4) and are thus excluded from the lists L_j (and $\hat{L}_j$). As mentioned previously, the number of such potential exons

grows roughly linearly with sequence length for sequences longer than a few kilobases. The only exception is for very unusual sequences which have enormous open reading frames — run time for such sequences will be significantly longer. Besides exons, the other type-$\mathcal{C}$ states, promoter and poly-adenylation signals, have short, highly restricted length distributions so that the total number of such states which need to be evaluated per position is only a small constant. Thus, total run time tends to grow only linearly with sequence length.

2.9.3 Suboptimal parses and exons

Of course, the parse with maximal joint probability may not necessarily correspond to the "correct" biological parse, i.e. the actual set of exons/genes present in the sequence. In addition, there may be more than one parse which can be considered "correct", for example, in the case of a gene which is alternatively transcribed, translated or spliced. "Suboptimal parses", i.e. parses which have joint probability which is slightly less than the optimal parse, may be of interest in both of these contexts. First, in studying the performance of the algorithm it may be useful to determine how close the correct parse was to the optimal parse in its joint probability. Secondly, it may be of interest to see whether alternative splicing patterns of a gene correspond to suboptimal parses, and perhaps to see whether alternative splicing can be predicted in cases where several optimal and/or sub-optimal parses have almost equally high probabilities. There are a number of ways to calculate suboptimal parses in general (e.g., Zuker, 1990). Most simply, one can store at each position of the Viterbi recursion the joint probabilities of the top k parses ending in state i at position j rather than just the single top solution: in this manner the top k parses overall can be found. A disadvantage of this approach is that k times as much memory and k times as many operations are required, which can be quite computationally expensive. An alternative approach, which has been implemented in the GENSCAN program, is to determine "suboptimal exons" rather than suboptimal parses. Specifically, the conditional probability of each potential exon is calculated (using the forward-backward formula), and all such exons whose probabilities exceed a prescribed minimum threshold are recorded. This task requires very little extra memory or time to accomplish

and can be helpful in identifying true exons missed by the optimal parse or exons with weak splice signals which are included in only some alternative splices of a gene (examples are given in Chapter 5).

2.10 Exon ratios and scores

The formal descriptions of the model and associated algorithms given in the preceding sections do not necessarily give much insight into how the method actually works. The purpose of this section is to help clarify exactly how each of the model components contribute to exon/gene prediction by detailing the Viterbi calculations for the forward-strand intron state types (I_0^+, I_1^+, I_2^+) for a short sample sequence:

```
                    10                  20                  30
1 2 3 4 5 6 7 8 9 | 1 2 3 4 5 6 7 8 9 | 1 2 3 4 5 6 7 8 9 |
T T T T A C A G G A C C A T G C T A C A C C G G T G G A T T
```

To simplify the discussion, only exons completely contained in the sequence are considered (and not those extending off the edges), and the states are indexed such that $Q_i = I_i^+$ for $0 \leq i \leq 2$. Since the sequence contains no stop codons and only one GT dinucleotide (at positions 24, 25), all of the lists L_j will be devoid of potential exons except for L_{23}. Therefore, in the induction step (eq. [7b]) for $\gamma_i(j)$, the second maximum is always over an empty set for $j \leq 23$, so the $\gamma_i(j)$ variables have the simple form:

$$[8] \qquad \gamma_i(j) = \pi_i \prod_{k=1}^{j} p_i P_i(S_k), \qquad 0 \leq i \leq 2, \quad 1 \leq j \leq 23,$$

representing the simple parses $\{q_1 = I_i^+, d_1 = j\}$, in which the sequence from 1 to j is in state I_i^+ (i.e., forward-strand phase i intron). The list L_{23}, however, will contain four potential exons (Table 1). Since the model distinguishes internal exon phase, the segment $[9, 23]$ represents three distinct potential exons, corresponding to the three phase (reading frame) types (see last column of Table 1). On the other hand,

Table 1. Exon list at position 23 of sample sequence

Exon	a_k	b_k	λ_k	x_k	y_k	z_k	Reading frame
ϵ_0	9	23	15	I_0^+	E_0^+	I_0^+	GAC CAT GCT ACA CCG
ϵ_1	9	23	15	I_1^+	E_1^+	I_1^+	GA CCA TGC TAC ACC G
ϵ_2	9	23	15	I_2^+	E_2^+	I_2^+	G ACC ATG CTA CAC CG
ϵ_3	13	23	11	F^+	E_{init}^+	I_2^+	ATG CTA CAC CG

Legend. Potential exons in the sample 30 bp sequence given in the text are described.

the segment $[13, 23]$ corresponds to only one potential initial exon, since the reading frame is determined by the location of the ATG.

The induction step (eq. [7b]) for $\gamma_0(24)$ will therefore involve a comparison between the quantities:

$A_0 = \gamma_0(23)p_0P_0(S_{24})$, corresponding to extension by one nucleotide of the previous optimal parse ending in state I_0^+ at position 23; and

$B_0 = \gamma_0(8)(1-p_0)T_{0,E_0^+}f_{E_0^+}(15)P_{E_0^+}(S_{9,23})T_{E_0^+,0}p_0P_0(S_{24})$, corresponding to the optimal parse of $S_{1,8}$ ending in state I_0^+ followed by an E_0^+ exon of length 15, followed by one nucleotide in state I_0^+.

The comparison for state I_1^+ is similar. For the I_2^+ recursion at $j = 24$, three different quantities are compared:

A_2, corresponding to extension of the $\gamma_2(23)$ parse by one nucleotide;

B_2, corresponding to addition of ϵ_2 (plus 1 bp of I_2^+) to the $\gamma_2(8)$ parse; and

C_2, corresponding to addition of ϵ_3 (plus 1 bp of I_2^+) to the $\gamma_{F^+}(8)$ parse.

Consider the "exon ratio", $R_0 = \frac{B_0}{A_0}$, the value of which determines whether ϵ_0 is included in the optimal parse ending in state I_0^+ at position 24. This ratio may be simplified using eq. [8] as follows:

$$[9] \qquad R_0 = \frac{\pi_0(\prod_{k=1}^{8} p_0P_0(S_k))(1-p_0)T_{0,E_0^+}f_{E_0^+}(15)P_{E_0^+}(S_{9,23})T_{E_0^+,0}p_0P_0(S_{24})}{\pi_0 \prod_{k=1}^{24} p_0P_0(S_k)}$$

$$= \underbrace{\frac{T_{0,E_0^+}T_{E_0^+,0}}{1}}_{(i)} \cdot \underbrace{\frac{(1-p_0)f_{E_0^+}(15)}{p_0^{15}}}_{(ii)} \cdot \underbrace{\frac{P_{E_0^+}(S_{9,23})}{P_0(S_{9,23})}}_{(iii)}$$

From this factorization it can be seen that inclusion or exclusion of the exon depends on: (i) the relative likelihood of making the state transitions $I_0^+ \to E_0^+$ and $E_0^+ \to I_0^+$ vs no transitions; (ii) the relative likelihood of ending an intron state $(1-p_0)$ and generating a 15 bp internal exon ($f_{E_0^+}(15)$) vs continuing an existing intron state by 15 bp (p_0^{15}); and (iii) the relative likelihood of generating the sequence segment $S_{9,23}$ under the model for an internal exon (E_0^+) vs the model for phase zero intron (I_0^+). Generally, the most important contribution to such exon ratios comes from the sequence generating models (particularly the splice signal models, which are components of the exon state models), so that the critical quantity for prediction is the ratio $R_0' = \frac{P_{E_0^+}(S_{9,23})}{P_0(S_{9,23})}$. Intuitively, this makes sense: the algorithm decides whether to classify the segment $[9,23]$ as exon or intron depending on the relative likelihood that the sequence was generated under these two alternative models. From another point of view, prediction depends not on the absolute probability of generating a sequence segment under the exon model, for instance, but rather on the ratio of this probability to that of generating the segment under the alternative (intron) model.

It is convenient to define the "score", σ, of an exon as the logarithm (base 2) of the exon ratio, e.g., the score of exon ϵ_0 is $\sigma_0 = \log_2(R_0)$. In general, positively scoring exons (i.e. those with exon ratios > 1) will tend to be included in the optimal parse while negatively scoring exons will not, but this is not always the case. For instance, if two potential exons ending at the same position and in the same state both have positive scores, then at most one will end up being included in the optimal parse ending in that state. The score, which is a measure of exon quality depending only on local sequence properties, can be contrasted with the conditional probability $P\{\epsilon|S\}$ described in Section 2.8, which depends on the entire sequence. For example, if two identical copies A, A' of a sequence segment which is potentially an exon occur at different places in a larger sequence, both will have the same exon score, but they will not in general have the same conditional probability. Biologically, the score may represent in part how "spliceable" an exon is, which may be a local property of

the splice signals, but the conditional probability may be a better representation of whether or not the exon will actually be included as part of a gene. For instance, if the segment A occurs upstream of the promoter and therefore is not even transcribed, then no matter how strong the splice signals happen to be, A will not become part of the mRNA, while the identical segment A' occurring in the middle of the transcription unit may very well be spliced as an exon.

Chapter 3

GENE STRUCTURAL AND COMPOSITIONAL PROPERTIES

This chapter describes the studies of the structural and compositional features of human genes which were undertaken in order to derive the parameters of the gene model described in the previous chapter. Section 3.1 describes the construction of the sets of human gene sequences which were studied. The next section reviews the isochore organization of the human genome and describes some of the quite dramatic relationships between C+G content and certain aspects of gene structure. These studies motivate the use of separate sets of model parameters to describe genes in distinct C+G% compositional groups of sequences. Section 3.3 describes the way in which state initial probabilities were derived using estimates of gene density and structural properties appropriate to sequences belonging to each of four C+G% compositional groups; the next section discusses the corresponding derivation for state transition probabilities. Section 3.5 discusses the length distributions used for introns, exons and other states of the model, and describes the procedure used to smooth the rather sparse empirical exon length distributions. Finally, the last section considers the compositional properties of bulk human coding and non-coding DNA and how these properties may be used in prediction.

3.1 Sequence sets

In order to derive a realistic description of the structural and compositional features of human genes, large nonredundant sets of sequences are required. A set of 386 human genes sequenced at the genomic level was initially constructed in the summer of 1994 and used for preliminary studies of gene compositional properties. Subsequently, a similar (but larger) set of 491 human genes was independently constructed by David Kulp (University of California, Santa Cruz) and Martin Reese (Lawrence Berkeley National Laboratories) for use as a common dataset for the training and testing of gene prediction methods[1]. In order to be able to more easily compare my results with those of other groups working on gene prediction, I chose to use the Kulp/Reese dataset (version of 22 August, 1995) for further studies of gene properties related to the development of the GENSCAN program. This set was constructed by searching the GenBank nucleic acid sequence database (Release 89, 1995) for sequences containing single complete human genes (i.e. containing at least the initial ATG through the stop codon) which were sequenced at the genomic level (as opposed to cDNA sequences). Certain additional constraints were imposed, e.g., there should be only one "CDS" (coding region) feature in the annotation (in order to avoid alternatively spliced genes), and the CDS annotation should be minimally self-consistent, e.g., there should be no inframe stop codons in coding regions and splice signals should match the minimal consensus (AG for acceptor sites, GT for donor sites). The resulting set was culled of redundant or highly similar entries by comparison at the protein level with the program BLASTP (Altschul *et al.*, 1990) — more complete details are given in Appendix A.

Close examination of this dataset revealed the presence of certain sequences which were deemed inappropriate for inclusion, including genes of mitochondrial or viral origin, genes for which the exon locations were described as "uncertain" or "putative", etc. — see Appendix A.[2] Elimination of these sequences resulted in a "clean" set of 428 sequences. For subsequent testing of the program, this set was further reduced

[1] [ftp://ftp.cse.ucsc.edu/pub/dna/genes]

[2] After discussions with M. G. Reese, it was decided to omit most of these sequences from subsequent versions of the Kulp/Reese dataset.

by removing all genes more than 25% identical at the amino acid level to those of the GeneParser test sets (Snyder & Stormo, 1995) using the PROSET program (Brendel, 1992) with default parameters. The set of 380 gene sequences resulting from this procedure, listed in Appendix A, are referred to as the GENSCAN learning set, $\mathcal{L}$.

The sequences of $\mathcal{L}$ have an average length of about 6,800 bp and in total represent 2,581 kilobases of genomic DNA, of which 16% is coding. The average amount of coding sequence per gene is 1100 bp, corresponding to an average protein length of 367 amino acids. Some of the sequences begin at or near the initial ATG and end soon after the stop codon, while others contain the complete 5′ and 3′ UTRs and a certain amount of the flanking (intergenic) regions. The set comprises a total of 142 single-exon (intronless) genes (this subset is designated $\mathcal{L}_{single}$) and 238 multi-exon genes (designated $\mathcal{L}_{multi}$): the multi-exon genes contain a total of 1,492 exons and 1,254 introns (hence 1,254 donor and acceptor splice signals). All of the sequences contain a single CDS feature which indicates the locations of the coding exons: since the coding region is usually of primary interest to the laboratory submitting the sequence and is typically determined by comparing the genomic and cDNA sequences, this feature is likely to be highly accurate. In addition, some sequences have a "prim_transcript" feature indicating the boundaries of the primary transcript and/or annotation indicating the location of the promoter or poly-adenylation signals: annotation of these features is in general much less reliable, however, and must be treated with caution.

All model parameters were derived from this data set as described later in this chapter except the promoter model, which was based on published sources, and the coding region model, for which this set was supplemented with a set of complete human cDNA sequences constructed as follows. All complete human cDNA sequences (containing at least the initial ATG through the stop codon) corresponding to proteins of at least 100 amino acids in length[3] were extracted from GenBank Release 83 (1994). This set was cleaned at the amino acid level using PROSET as above both with respect to itself and with respect to the GeneParser test sets, resulting in a set of 1,619 cDNA sequences, designated $\mathcal{L}_{cDNA}$ (sequence list available on request). This set was then combined with the coding sequences from $\mathcal{L}$ to form a set $\mathcal{L}_{coding}$ of 1,999 complete

[3]The length minimum was imposed in order to avoid inclusion of cDNA fragments.

coding sequences totaling in excess of 1,065,000 codons.

3.2 Gene structure and C+G content

A question which must be addressed at the outset is whether the human genome is essentially homogeneous, in which case a single set of model parameters might be suitable to describe all genes, or heterogeneous, in which case several sets of model parameters might be more appropriate. Several lines of evidence suggest that heterogeneity is the rule rather than the exception. First, significant differences in gene structure (e.g., intron length) were observed between distinct C+G% compositional subsets of the learning set $\mathcal{L}$ (described later in this section). Second, the predictive accuracy of most existing gene prediction programs varies depending on the C+G content of the sequence (e.g., Lopez *et al.*, 1994, Snyder & Stormo, 1995), with typically lower levels of accuracy observed for A+T rich sequences. Third, a large body of work on the compositional structure of the human genome has demonstrated that the genome is substantially heterogeneous with respect to C+G content and that gene density, gene length and other important properties appear to be strongly correlated with C+G content. This work is reviewed briefly below.

3.2.1 The isochore organization of the human genome

Bernardi and other researchers (e.g., Bernardi *et al.*, 1985 and references therein) have used $CsCl$ and Cs_2SO_4 density gradient centrifugation of randomly sheared genomic DNA and other experimental techniques to study genome compositional properties. These studies have shown that the human genome (and the genomes of other warm-blooded vertebrates) is a mosaic of "isochores", large regions perhaps several hundreds of kilobases or more in length whose base composition is locally homogeneous but varies significantly between disjoint regions. Although the exact number of distinct isochores and the range of C+G% composition corresponding to each are not completely settled, the genome is typically divided into five categories or "compartments", labeled L1, L2 (L for light, A+T rich), H1, H2 and H3 (H for

heavy, C+G rich) in increasing order of C+G% content.

Using several sources of information, including the $CsCl$ centrifugation data of Cuny *et al.* (1981) combined with compositional studies of sequenced human genes of known isochore localization, Mouchiroud *et al.* (1991) estimated the approximate proportion of genomic DNA and proportion of genes found in each of three isochore groupings:

L1+L2 (less than about 43% genomic C+G), 62% of genome, 34% of genes;

H1+H2 (about 43-51% C+G), 31% of genome, 38% of genes; and

H3 (> 51% C+G), 3-5% of genome and 28% of human genes.

As a consequence, gene density in C+G rich regions (H3) is estimated to be at least five times higher than in moderate C+G regions (H1+H2) and at least ten times higher than in A+T rich regions (L1+L2)! Recent studies comparing transcriptional mapping data with isochore classification of distinct regions of human chromosome 21 (reviewed in Gardiner, 1996) have confirmed the existence of extreme differences in gene density between A+T rich and C+G rich portions of the human genome. Studies of available GenBank sequences by Duret *et al.* (1995) revealed other striking differences between regions of differing C+G content, e.g., that the amount of intronic DNA is on average three times higher for genes in A+T rich regions (L1+L2) than for genes in C+G rich regions (H3).

3.2.2 Effect of C+G% content on gene structural properties

Given this previous work, it was naturally of interest to see whether corresponding differences existed between genes of differing C+G content in the learning set. The set $\mathcal{L}$ was initially partitioned into three subsets corresponding to those used by Duret *et al.* and Mouchiroud *et al.*, but since the H3 subset was far more populated than the others (200 out of 380 sequences), it was divided approximately in half to give a total of four subsets: I (< 43% C+G); II (43 – 51); III (51 – 57); and IV (> 57). The sequences of $\mathcal{L}$ were assigned to these groups based on the C+G% composition of the GenBank sequence, assuming that this is a reasonably close approximation to

the content of the genomic region from which the sequence derived. Some structural properties of the single- and multi-exon genes in these four subsets are compared in Table 2.

The most dramatic difference seen between the groups I to IV is that, consistent with the results of Duret *et al.* (1995), intron length increases dramatically as a function of A+T content, with introns in the most A+T rich group (I) almost four times longer on average than those in the most C+G rich group (IV). Exon lengths, on the other hand, appear to decrease slightly with increasing A+T content, but the differences are far less pronounced. The number of introns per multi-exon gene also appears to be roughly the same in the four groups. Other features of note are the surprisingly high proportion (142/380 = 37%) of single-exon genes and the skewed distribution of genes across the C+G compositional classes. Notably, the proportion of genes in group I (65/380 = 17%) is only half that expected from the data of Mouchiroud *et al.* (34%) for isochores L1+L2, the proportion in group II (115/380 = 30%) is lower than the 38% estimated for isochores H1+H2, and the proportion in groups III+IV (200/380 = 53%) is almost twice the expected value (28%) for isochore H3. Additionally, the average CDS lengths for single- and multi-exon genes (1,224 and 1,029 bp, respectively) are surprisingly short compared to the average CDS length of 1,719 bp observed in the set $\mathcal{L}_{cDNA}$.

A likely explanation which accounts at least qualitatively for all of these somewhat puzzling results is that, since short genes are easier to sequence completely, there is a strong systematic bias in the set $\mathcal{L}$ toward short genes. Such a bias would likely be weaker in the set $\mathcal{L}_{cDNA}$ since the length of the cDNA of a gene is typically much shorter than its full genomic extent and therefore doesn't pose so much of an impediment to complete sequencing. A bias toward short genes also explains the skewed distribution of genes across the four C+G groups. Specifically, the fact that genes in A+T rich regions (group I) are approximately twice as long as genes in moderate to high C+G regions (e.g., comparing the estimated transcript lengths in Table 2) and therefore require approximately twice as much time and effort to sequence, probably explains why there are only half as many such genes as expected in the set $\mathcal{L}$. Conversely, the fact that single-exon genes are typically less than half as

Table 2. Structural properties of genes partitioned according to C+G% content

	Group				
Property	I	II	III	IV	All
C+G% range	< 43	43-51	51-57	> 57	0-100
Corresponding isochore	L1+L2	H1+H2	H3	H3	All
Number of single-exon genes	21	44	45	32	142
Number of multi-exon genes	44	71	54	69	238
Total number of genes	65	115	99	101	380
Mean CDS length, $\mathcal{L}_{single}$ (bp)	1,130	1,251	1,304	1,137	1,224
Mean CDS length, $\mathcal{L}_{multi}$ (bp)	902	908	1,118	1,165	1,029
Mean exon length, $\mathcal{L}_{multi}$ (bp)	148	154	172	177	164
Mean intron length, $\mathcal{L}_{multi}$ (bp)	2,069	1,086	801	518	1018
Introns per gene, $\mathcal{L}_{multi}$	5.1	4.9	5.5	5.6	5.3
Mean transcript length, $\mathcal{L}_{single}$ (bp)	2,356	2,477	2,530	2,363	2,450
Mean transcript length, $\mathcal{L}_{multi}$ (bp)	12,680	7,455	6,750	5,292	7,621

Legend. Genes of the learning set were partitioned into four groups as described in the text. Average properties were derived from the sequences of each group as a whole, or restricted to the single-exon genes ($\mathcal{L}_{single}$) or multi-exon genes ($\mathcal{L}_{multi}$) of the group, as indicated. Since the proportion of sequences containing the necessary "prim_transcript" annotation was too low to permit reliable estimation of average $5'$ and $3'$ UTR lengths in each group seperately, only overall average values of 769 bp and 457 bp, respectively, were calculated: these values were used in the estimation of mean transcript lengths above and in subsequent calculations.

long as multi-exon genes at the genomic level (again, comparing mean transcript sizes, Table 2) probably means that single-exon genes are over-represented at least twofold in the set $\mathcal{L}$ relative to the true genomic proportion of such genes. In order to correct for this "short gene bias", the proportion of single-exon genes used in derivation of the GENSCAN parameters was taken to be one half of the observed value in each C+G% group. Although this estimate is rather crude and could err significantly in either direction, it is likely to be more accurate than simply ignoring the bias in $\mathcal{L}$. There are undoubtedly other types of biases present in the set $\mathcal{L}$, e.g., a bias toward genes of medical interest, a bias away from genes whose cDNAs are difficult to clone, etc. Since it would be extremely difficult to account for such biases systematically, it was decided simply to live with them.

3.3 Initial probabilities

Since the goal is to model a randomly chosen block of contiguous human genomic DNA (as might be generated by a genome sequencing laboratory), the initial probability of each state should be chosen proportionally to its estimated frequency in bulk human genomic DNA. Because the sequences of the learning set are all centered on genes (and typically contain little or no flanking intergenic DNA), they are not representative of typical genomic fragments. However, by combining previously published estimates of the total number of genes in the human genome and the total size of the genome with estimates of the approximate proportions of DNA and of genes present in each of the isochore compartments, it was possible to estimate the approximate proportion of each of the type-$\mathcal{D}$ state types (intron, intergenic and so on) in genomic DNA for each of the four C+G compositional groups described above. The final results of these calculations and some of the intermediate values are listed in Table 3: a sample calculation is given in the table legend. The specific assumptions made were that the genome contains 65,000 genes (Fields *et al.*, 1994), has a total size of 3,400 Mb (Cavalier-Smith, 1985), and that the DNA amount and gene numbers are distributed essentially as estimated in Mouchiroud *et al.* (1991) — see previous section.

Table 3. Estimation of state initial probabilities

C+G% compositional group	I	II	III	IV
Bulk human genomic DNA				
Estimated proportion of genome	62%	31%	3%	2%
Estimated DNA amount in genome (Mb)	2,074	1,054	102	68
Estimated gene number	22,100	24,700	9,100	9,100
Estimated mean intergenic length (bp)	83,000	36,000	5,400	2,600
Estimated initial probabilities				
Intergenic (N)	0.892	0.867	0.540	0.418
Intron ($I_0^+, I_1^+, I_2^+, I_0^-, I_1^-, I_2^-$)	0.095	0.103	0.338	0.388
$5'$ Untranslated region (F^+, F^-)	0.008	0.018	0.077	0.122
$3'$ Untranslated region (T^+, T^-)	0.005	0.011	0.045	0.072

Legend. The data in the upper portion of the table (see text) were used to estimate the initial probabilities for each of the four type-$\mathcal{D}$ state types shown in the lower portion. The method of calculation is illustrated below for group II sequences (43-51% C+G) — parameters for other groups were estimated similarly. From Table 2, the observed proportion of single-exon genes in group II sequences is $44/115 = 0.38$: correcting for the "short gene bias" gives a revised proportion of 0.19. Therefore, using the mean CDS lengths for single- and multi-exon genes from Table 2, the total amount of coding DNA in this isochore is approximately $24,700 \times (0.19 \times 1,251 \text{ bp} + 0.81 \times 908 \text{ bp}) = 24$ Mb, leaving $1,054 - 24 = 1,030$ Mb of non-coding DNA in this isochore. Using the average intron length and number of introns per multi-exon gene from Table 2, approximately $0.81 \times 24,700 \times 4.9 \times 1,086$ bp $= 106$ Mb of this total is intronic. So, the proportion of group II non-coding DNA which is intronic is approximately 106 Mb / 1,030 Mb = 0.103. Similarly, the amount of $5'$ UTR DNA in this isochore is estimated as $24,700 \times 769$ bp $= 19$ Mb, and the estimated $3'$ UTR amount is $24,700 \times 457$ bp $= 11$ Mb, yielding initial probabilities of $19/1,030 = 0.018$ and $11/1,030 = 0.011$, respectively. Finally, a total of $1,030 - (106 + 19 + 11) = 894$ Mb is intergenic, giving an average between-gene intergenic length of 894 Mb / 24,700 = 36 kb. Estimates for groups III and IV, which correspond to subsets of the H3 isochore, were made assuming that approximately 60% of the DNA and one half of the genes in the H3 isochore belong to group III (51 – 57% C+G), with the remainder in group IV (> 57%).

Note that the differences in initial probabilities are quite dramatic with, for example, the probability of hitting an intergenic region much higher in A+T rich sequences than for C+G rich ones, and conversely for intron states. Since the coding strand of the input sequence is assumed to be unknown a priori, the initial probabilities of the strand-specific states are estimated symmetrically so that, for example, the initial probabilities π_{F+} and π_{F-} for group II sequences are each estimated to be $0.018/2 = 0.009$ (see Table 3). For the intron states, there is also the complication of phase. Interestingly, the three intron phases are not represented equally in human genes. The observed proportions in the learning set were: 41.5% (phase 0), 38.1% (phase 1) and 20.4% (phase 2). Similar proportions have been observed in previous studies of intron phase, e.g., Smith (1988), Fedorov *et al.* (1992), Long *et al.* (1995) and Tomita *et al.* (1996). The initial probabilities of the states I_k^+ (and I_k^-) were estimated using these observed proportions in the obvious way, e.g., for group II sequences, $\pi_{I_0^+} = 0.415 \times 0.103/2 = 0.021$ (see Table 3).

3.4 Transition probabilities

The biologically permissible state transitions are shown as arrows in Fig. 2. Certain transitions are obligatory (e.g., $P^+ \rightarrow F^+$, $T^+ \rightarrow A^+$) and hence are assigned probability one: probabilities for all other transitions are estimated from the learning set as follows. Since all of the genes in $\mathcal{L}$ occur on the "forward" strand (i.e. the strand of the GenBank sequence), all transition probabilities between forward-strand states were assigned values equal to the observed frequency in the learning set (adjusted for the "short gene bias" if appropriate). For example, the probability of an $I_2^+ \rightarrow E_{term}^+$ transition was set equal to the observed fraction of phase 2 introns which are followed by terminal exons. Transitions between reverse-strand states were estimated in the same way from an analogous set $\mathcal{L}^-$ constructed by taking the inverted complement of each sequence of $\mathcal{L}$: it can be easily checked that this results in a model which is "strand-symmetric" in the sense that the product of the transition probabilities corresponding to any particular gene structure will be the same whether the gene occurs on the forward or reverse strand. The probabilities of the transitions $N \rightarrow P^+$

Table 4. Distribution of adjacent intron phases

5′ intron	3′ intron						
	phase 0		phase 1		phase 2		All phases
	O	E	O	E	O	E	O
phase 0	205	(177)	133	(151)	78	(88)	416
phase 1	141	(170)	168	(145)	89	(84)	398
phase 2	87	(86)	68	(73)	47	(43)	202
All phases	433		369		214		1,016

$$\chi^2 = 17.4 \text{ (P} < 0.005, 4 \text{ d.f.)}$$

Legend. Phases of 1,016 pairs of adjacent introns from $\mathcal{L}$: O indicates observed count, E indicates expected count (product of marginals).

("initiating" a forward-strand gene) and $N \rightarrow A^-$ ("initiating" a reverse-strand gene) were set equal to $\frac{1}{2}$ in order to preserve the strand-symmetry of the model, i.e. given an arbitrary stretch of sequence, it is assumed that a gene is equally likely to be encountered in the forward or reverse orientation.

The most interesting property observed in derivation of the transition probabilities was that the phases of successive introns are correlated (this issue arises in derivation of the $E_j^+ \rightarrow I_k^+$ and $I_j^- \rightarrow E_k^-$ transition probabilities). Table 4 shows the distribution of the phases of adjacent intron pairs in the learning set, i.e. of the introns immediately 5′ and 3′ to each internal exon. The χ^2 test reveals significant dependence between the phases of adjacent introns, consistent with the results of Long *et al.* (1995) and Tomita *et al.* (1996). The reasons for this correlation are not entirely clear, but it is worth noting that successive introns will have the same phase if and only if the intervening exon has length which is a multiple of three, so that selection might be acting on exons which have this potentially desireable property. This property might be advantageous for regulation by alternative splicing in the sense that an exon whose lengths is a multiple of three can be alternatively skipped or included without disrupting the reading frames of the adjacent exons or could be important for "exon shuffling" (e.g., Dorit & Gilbert, 1991).

3.5 Length distributions

As discussed in the previous chapter, all type-$\mathcal{D}$ state are assumed to have geometric length distributions, as would be expected if there are few or no functional constraints on the lengths of these features. Since the mean, μ, of a geometric distribution is related to the parameter, q, by the relation $\mu = \frac{1}{q}$, the parameter may be estimated from an observed mean value simply as $q = \frac{1}{\mu}$. For the intergenic state, N, separate geometric parameters were estimated for the four C+G% groups from the estimated mean intergenic lengths in Table 3. Intron length parameters were estimated similarly from the mean lengths in Table 2.[4] For the $5'$ and $3'$ UTR states, the same parameter was used for all four C+G groups (see legend to Table 2). The promoter and polyadenylation states have special length distributions which are discussed in Chapter 4.

3.5.1 Exon lengths

Since exon length distributions differ significantly for different types of exon (Fig. 1), but do not appear to vary substantially between the C+G% compositional groups (e.g., Table 2), it was considered preferable to derive separate distributions for each exon type, but to pool exons from the four C+G groups in order to keep the sample sizes as large as possible. Although the internal exon length distribution (Fig. 1c) looks a bit like a normal (Gaussian) density, the other types of exon (Figs. 1b, 1d) have distributions unlike any standard statistical distribution. Nor is there any obvious reason to expect that they should, given the various types of potential constraints on exon length discussed in Section 2.2.3. An alternative to using standard statistical distributions such as the geometric or normal is to use an empirically-based distribution, estimating the probability of observing each exon length directly from the available data.

Since the number of exon lengths available from the learning set (238 initial, 1,016 internal, and 238 terminal exon lengths) is somewhat limited relative to the set of possible exon lengths, the empirical distribution is fairly sparse and many

[4] Intron length is assumed to be independent of intron phase.

possible lengths were missing (not observed). Of course, simply because no exons of a particular length occurred in the learning set does not mean that the probability of this length should be assumed to be zero: a more likely explanation is that the length was missed simply because of normal sampling fluctuations. In other words, the true distribution of lengths is likely to be much smoother than that derived from a relatively small sample such as the learning set, raising the issue of whether there might be a way in which the smooth underlying distribution can be approximated from a relatively sparse sample. Consideration of a simple model for the evolution of exon lengths leads to a fairly natural smoothing procedure described in the following section. Of course, the simple goal of exon/gene prediction may not require this level of attention to the details of exon lengths. However, the smoothing method described here is fairly general and might be of some independent interest.

3.5.2 A model for exon length evolution

Lengths of coding exons most likely evolve by insertion or deletion of an integral number of DNA triplets, since changing the length by a number of base pairs which is not a multiple of three will change the reading frame and probably have severe consequences at the protein level.[5] It is assumed for simplicity that at each position, only single-codon insertions or deletions occur in a single generation (multiple-codon changes at the same position are still allowed, but they must occur over multiple generations) and that the probabilities per generation of insertion or deletion at any given codon position have the same (extremely small) value p. Making the reasonable assumption that $\lambda p \ll 1$ (e.g., λ is typically on the order of 10^2 and p is probably $< 10^{-9}$), so that the probability of insertion or deletion of multiple codons in an exon in a single generation is negligible, an exon of length λ codons will increase or decrease in length by one codon with probability approximately λp per generation.

If $X_0 = \lambda$ is the initial length of the exon and X_n its length after n generations have elapsed, the sequence $X_0, X_1, ...$ will describe a special type of random walk[6] on

[5] More complex types of mutation, e.g., involving recombination, are not considered in the present model, and selection acting on exon length is assumed to be absent.

[6] This model may also be framed as a branching (birth and death) process, e.g., Feller (1950),

the positive integers (e.g., Karlin & McGregor, 1959), according to:

$$[10] \qquad P\{X_{k+1} = n | X_k = m\} = \begin{cases} pm & \text{if } n = m+1 \text{ or } n = m-1 \\ 1-2pm & \text{if } n = m \\ 0 & \text{if } |n-m| > 1 \end{cases}$$

It is easily seen[7] that $E[X_{k+1}|X_k] = X_k$ for $k \geq 0$ (the martingale property), from which it follows that $E[X_n] = X_0 = \lambda$ for all n. Furthermore, using the notation $V[Y]$ to represent the variance of a random variable Y, it is easily shown that the conditional variance, $V[X_{k+1}|X_k] = E[(X_{k+1} - E[X_{k+1}])^2|X_k] = 2pX_k$. Therefore, the (unconditional) variance of X_{k+1} can be calculated as:

$$\begin{aligned} V[X_{k+1}] &= E[(X_{k+1}-\lambda)^2] \\ &= E[E[(X_{k+1}-\lambda)^2]|X_k] \\ &= E[E[(X_{k+1}-X_k+X_k-\lambda)^2|X_k]] \\ &= E[E[(X_{k+1}-X_k)^2|X_k] + E[(X_k-\lambda)^2|X_k] + 2E[(X_{k+1}-X_k)(X_k-\lambda)|X_k]] \\ &= E[2pX_k+(X_k-\lambda)^2] \\ &= 2p\lambda + V[X_k] \end{aligned}$$

Since $V[X_0] = 0$, it follows that $V[X_n] = 2np\lambda$ for $n \geq 0$, so the variance of the position of the random walk increases linearly with time and, for fixed time, is proportional to the initial position (i.e. the length of the exon at generation 0). Computer simulations of this process show that for reasonable choices of λ (e.g., 50) and p (e.g., $p = 10^{-10}$), after a moderately large number of generations the distribution of lengths, though slightly asymmetric about the mean, becomes almost normal in shape, as might be expected.

These observations may be applied to smooth an empirical length distribution as follows. Assuming that most genes (and hence their component exons) evolve

---

Karlin & McGregor (1958).

[7] $E[X]$ indicates the expected value of the random variable X; $E[X|Y]$ indicates the conditional expected value of X given Y.

by duplication and subsequent divergence (many such examples are known, e.g., the globin, tubulin and Hox families, etc.), and the number of generations n which have elapsed since the duplication giving rise to a typical gene is on the order of or smaller than $1/p \approx 10^{10}$, then the ancestral exon giving rise to an observed exon of length λ codons will typically have a length within $\sigma = \sqrt{2np\lambda}$ ($< \sqrt{2\lambda}$) of λ. For example, an observed exon of length 50 codons likely had an ancestral exon between 40 and 60 codons in length. Taking the learning set to represent a random sample of the genes in the human genome, we can approximate the remaining genes in the genome which are related to this set by at least a very distant common ancestral gene (probably most genes) as follows. Consider each exon of $\mathcal{L}$ not as a single example, but as representing the whole family of exons which evolved from the same common ancestral exon. By the arguments given above, the distribution of the lengths of these exons[8] will typically be approximately normal with mean $\mu \approx \lambda$ and variance $\sigma^2 \approx 2\lambda$, leading to the following smoothing procedure.

3.5.3 Smoothing procedure for sparse length data

The observed length distribution for a particular exon type can be represented as a vector, $\vec{n} = n_1, n_2, ..., n_m$, where n_k is the number of exons observed (possibly zero) of length k codons (lengths are rounded up to the nearest whole number of codons) and m is the maximum length observed. The total number of lengths observed is denoted $N = \sum_{k=1}^{m} n_k$. The "empirical distribution" is the probability density which assigns mass $f_k = \frac{n_k}{N}$ to each point $k = 1, ..., m$: the empirical distribution for terminal exon lengths is shown by the solid vertical lines in Fig. 3. The smoothed distribution is created by replacing the probability mass f_k at position k by a "discretized" normal density[9] with mean $\mu = k$ and variance $\sigma^2 = \frac{2Ck}{n_k}$ (C a positive constant), scaled so

[8] This model could, of course, be tested by examining the distribution of the lengths of the set of corresponding exons from a group of homologous genes.

[9] A normal distribution may be made discrete by assigning to the integer j the mass f_j derived by integrating the normal density from $j - 0.5$ to $j + 0.5$ (which may be conveniently calculated on a computer using the "erf" function). In this context, the discretized normal is restricted to the positive integers by setting the density equal to zero for all non-positive integers and appropriately rescaling so that the total mass is not changed.

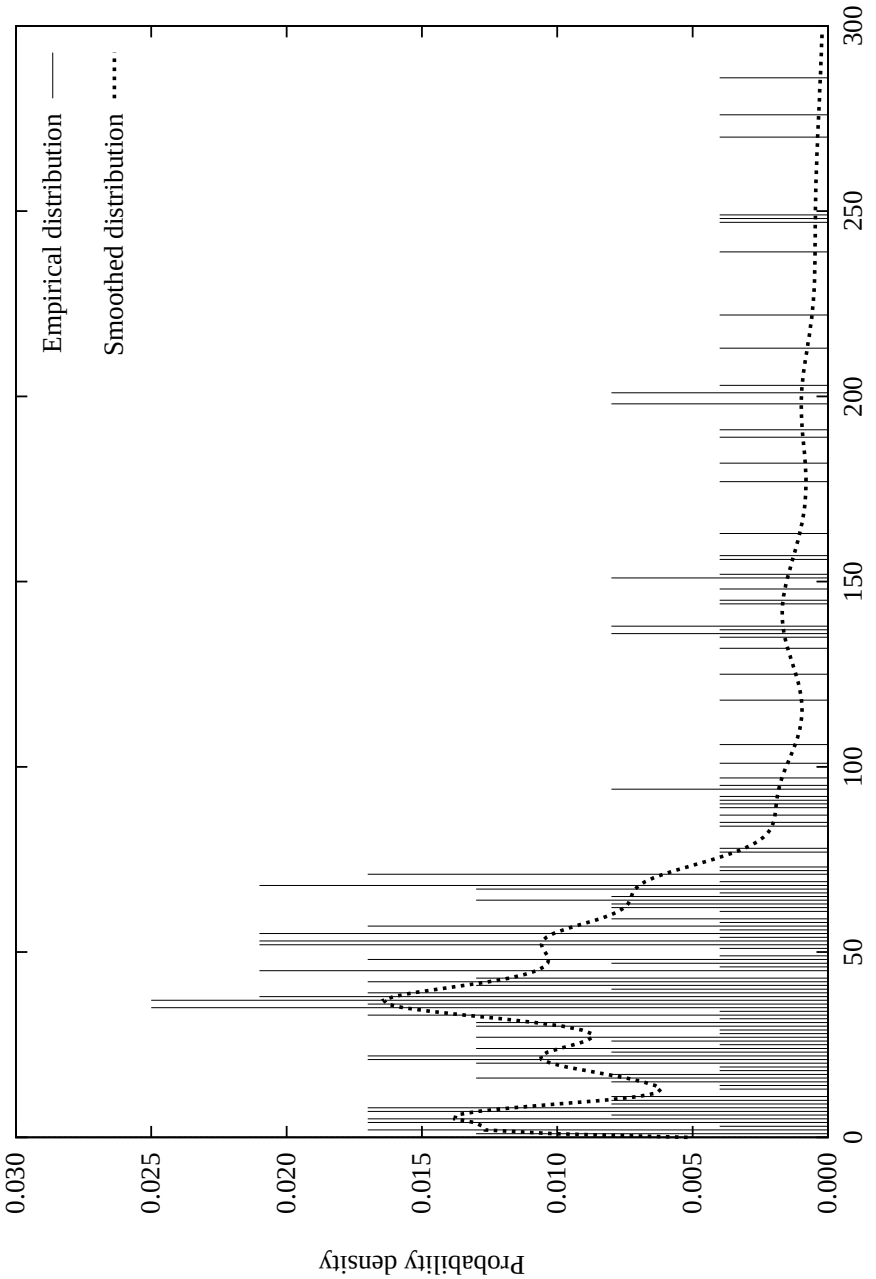
that the total mass is f_k. The resulting smoothed (normal mixture) distribution for terminal exon lengths is shown (for $C = 1$) by the dotted line in Fig. 3.

The previous section justified the replacement of an observed length k by a normal distribution of mean k and variance $2npk$ — here, it is assumed that the number of elapsed generations is very large, so that $C = np \approx 1$. Division of this variance by n_k does not follow directly from the evolutionary model, but is a correction for the reduction in variance which results from increased sample size. This additional factor lends to the smoothed distribution two additional intuitively desireable properties which would otherwise be lacking, namely: (i) the empirical distribution is smoothed more in sparse areas of the length distribution and less in regions where the data is more dense; and (ii) as the total number of samples increases, the smoothed distribution approximates more and more closely the empirical distribution. Of course, even if the evolutionary model is not very realistic, the type of normal-based smoothing described above may still be appropriate in the sense that the resulting smoothed distribution resembles the curve that a reasonable person might draw by hand to describe the data.

3.6 Composition of coding and non-coding DNA

A wide variety of compositional differences between coding and non-coding DNA have been described in the literature, including differences in C+G content, oligonucleotide content, periodic properties, local compositional complexity, and others (summarized in Section 1.1). However, it is not immediately apparent which of these differences are primary and which are simply consequences of other, more fundamental factors. This issue was addressed by Fickett & Tung (1992), who compared more than twenty different measures in terms of their ability to discriminate the coding or non-coding character of sequence segments approximately 100 bp in length. The results were that measures based on frame-specific hexamer composition (see below) were the most discriminatory and that most of the other measures proposed are redundant with respect to these measures in the sense that they reflect features (e.g., amino acid usage, codon usage, di-amino acid usage, dinucleotide bias) which can be derived

Fig. 3. Terminal exon lengths: empirical and smoothed distributions



Legend. Length data for 238 terminal exons from learning set (Appendix A). Empirical and smoothed distributions are as defined in text.

from the frame-specific hexamer composition.

The model of coding regions used here is an inhomogeneous 3-periodic fifth-order Markov model (see Section 2.1), of the sort used by Borodovsky & McIninch (1993) in the widely used GENMARK program for identification of genes in prokaryotic genomes. In this approach, separate fifth-order Markov transition matrices, denoted $C^{(1)}, C^{(2)}, C^{(3)}$, are determined for hexamers ending at codon positions 1, 2 and 3, respectively, so that the probability of generating the next base pair in the sequence is conditional on the codon position as well as the identities of the previous five nucleotides in the sequence. The three transition matrices thus capture biases in dicodon usage (encompassing the lower-order biases in amino acid, di-amino acid and synonymous codon usage), as well as biases in the frequencies of the two types of out-of-frame hexamers in coding sequences.

Since each transition matrix $C^{(i)}$ has $4^6 = 4{,}096$ entries, representing approximately 3,000 independent parameters, a large amount of sequence data is required to reliably estimate these parameters. For this purpose, the coding regions of the genes of the learning set were supplemented by a large set of complete cDNA sequences to form the set $\mathcal{L}_{coding}$, of more than 1,000,000 codons, as described in Section 3.1. Each Markov transition probability is then set equal to its maximum likelihood estimate, which is the corresponding conditional frequency from the set $\mathcal{L}_{coding}$. Since the size of the data set is $> 10^6$ and the total number of independent parameters in the three Markov matrices is $< 10^4$, there is an average of more than 100 data points per parameter, which should be sufficient to give reasonably reliable estimates. Lower-order 3-periodic Markov models were also tried, but gave inferior results (data not shown) — higher-order models are prohibited at present by lack of sufficient sequence data.

These matrices are used to model the codon positions of the exon states in the natural way. For example, the first coding positions of an internal exon of phase $h = 1$ are modeled using matrices $C^{(2)}$, $C^{(3)}$, $C^{(1)}$, $C^{(2)}$, $C^{(3)}$, ..., and so on until the last coding base pair has been generated.[10] This treatment of the coding portions of multi-exon genes is essentially equivalent to the "in-frame scoring" plus "in-frame

[10]The description above does not apply to positions at the edges of the exon which are overlapped by signal models (see Chapter 4).

assembly" approach described by Wu (1996), which he has shown gives somewhat better accuracy than alternative methods of coding region scoring/assembly, e.g., those used by GeneParser (Snyder & Stormo, 1995) and by the gene assembly option of GRAIL II (Xu *et al.*, 1994b).

The non-coding states F, T, N and I_k are modeled using a homogeneous fifth-order Markov matrix, D, with transition probabilities derived from the non-coding portions of the genes in $\mathcal{L}$ (a total of more than 2 Mb of DNA). Despite the presumably much lower degree of selective pressure acting on non-coding sequences, strong biases in hexamer composition nevertheless exist, many of which may reflect biases in mutational or repair processes. For example, very high AAAAA $\rightarrow$ A and TTTTT $\rightarrow$ T transition probabilities of 0.55 and 0.54, respectively, were observed in non-coding regions, putatively because of a bias in polymerase slippage-induced mutation toward extension of pre-existing runs of A or T bases (e.g., Schlotterer & Tautz, 1992). Other transitions with unusually high frequency include GCCCG $\rightarrow$ G (0.66), GGCGT $\rightarrow$ G (0.62) and CACGC $\rightarrow$ C (0.56), all related to hexamers present in consensus *Alu* sequences (Jurka & Smith, 1988). At the other extreme, the greatest biases were seen for the transitions TTTTC $\rightarrow$ G (0.023) and ATTTC $\rightarrow$ G (0.025), reflecting the pronounced suppression of CG dinucleotides in vertebrate non-coding regions (e.g., Karlin & Burge, 1995), which reaches its greatest intensity in A+T rich regions.

3.6.1 Coding differential

The concept of "coding differential", introduced below, provides a convenient way of summarizing the way in which these models of coding and non-coding DNA contribute to prediction. Letting $C^{(i)}[S_j]$ denote[11] the probability of generating nucleotide S_j under the transition matrix $C^{(i)}$, the probability of generating the sequence segment $[a, b]$ as a phase h coding region (i.e. beginning in codon position $h+1$) can be written as:

[11] The dependence on the nucleotides $S_{j-5}, S_{j-4}, S_{j-3}, S_{j-2}, S_{j-1}$ is not made explicit in order to simplify the notation.

[11] $$P^{(h)}(S_{a,b}) = \prod_{i=0}^{b-a} C^{(\,(h+i) \bmod 3 + 1\,)}[S_{a+i}]$$

Similarly, letting $D[S_j]$ denote the probability of generating nucleotide S_j from the transition matrix D for non-coding regions, the probability of generating the segment $[a, b]$ under the non-coding sequence model is given by: $P^D(S_{a,b}) = \prod_{i=0}^{b-a} D[S_{a+i}]$. Recall (Section 2.10) that the critical quantity for prediction of the region $[a, b]$ as an exon or intron is the ratio, $R = \frac{P_E(S_{a,b})}{P_I(S_{a,b})}$, where E represents a particular exon state type, and I the corresponding non-coding (e.g., intron) state type. If the portions of this ratio which correspond to the splicing or translational signals are omitted, we are left with the "coding ratio", $r = \frac{P^{(h)}(S_{a,b})}{P^D(S_{a,b})}$, where h is the phase of the exon[12] or, equivalently, the "coding score", $s = \log_2(r)$. If the sequence contains n coding exons of lengths $\lambda_1, ..., \lambda_n$, whose coding scores are $s_1, ..., s_n$, the average coding score per coding base pair, μ_C, is given by $\mu_C = \frac{\sum_{i=1}^{n} s_i}{\sum_{i=1}^{n} \lambda_i}$. If $\mathcal{N}$ denotes the set of non-coding nucleotide positions in the sequence, then the average coding score per non-coding base pair, μ_D, is defined by:

[12] $$\mu_D = \sum_{i \in \mathcal{N}} \frac{1}{3} \left[\log_2\left(\frac{C^{(1)}[S_i]}{D[S_i]}\right) + \log_2\left(\frac{C^{(2)}[S_i]}{D[S_i]}\right) + \log_2\left(\frac{C^{(3)}[S_i]}{D[S_i]}\right) \right]$$

Note that the coding score is averaged over the three possible reading frames for non-coding sequence positions.

The "coding differential", Δ, is defined as $\Delta = \mu_C - \mu_D$, i.e. the difference in coding score per base pair for coding vs non-coding regions of the sequence. This measure provides a convenient gauge of how well the sequence generating models are able to distinguish coding from non-coding regions. For example, if $\Delta = 0$ for a sequence, then on the whole the coding/non-coding sequence generating models

[12] Initial exons are assigned phase zero: terminal exons are assigned the phase of the previous intron state.

provide no help in distinguishing exons from introns, while if $\Delta < 0$, then these models actually tend to misclassify exons as introns and vice versa. On the other hand, if the coding differentials Δ_1 and Δ_2 of two sequences differ with, say, $\Delta_1 > \Delta_2 > 0$, then on average the model is better able to distinguish coding from non-coding DNA in sequence 1 than in sequence 2.

Figure 4 shows the relation between coding differential and C+G% content for the 380 genes of $\mathcal{L}$. Two features stand out. First, as one would hope, Δ is nearly always positive (only two of 380 sequences have negative values), meaning that hexamer composition almost always helps to distinguish coding from non-coding regions of a gene. Secondly, coding differential is significantly positively associated with C+G% content, as measured by a Pearson (product-moment) correlation coefficient of $\rho = 0.44$ ($P < 0.01$). This association is also seen by comparing the mean Δ values of 0.130, 0.164, 0.184, and 0.197, for groups I, II, III and IV, respectively. In particular, the Δ value for group I genes stands out as substantially lower than the others. This phenomenon may in part explain the observation that gene prediction programs, many of which make use of differences in hexamer composition or related functionals, tend to perform less well on A+T rich sequences (e.g., Xu *et al.*, 1994a; Lopez *et al.*, 1994; Snyder & Stormo, 1995).

A possible explanation for the lower coding differential values in A+T rich sequences is that the set $\mathcal{L}_{coding}$ from which the $C^{(i)}$ matrices were derived (and the set $\mathcal{L}$ from which the D matrix was derived) are heavily biased toward C+G rich genes (e.g., Section 3.2). One way to counteract such a bias might be to construct new matrices specifically for group I sequences, derived from the coding and non-coding portions of A+T rich genes only. Therefore, a new fifth-order Markov transition matrix D_I was derived using only the group I sequences from $\mathcal{L}$ (totaling more than 700 kb of genomic DNA), and new coding matrices, $C_I^{(i)}$ ($1 \leq i \leq 3$) were derived from the set $\mathcal{L}^I_{coding}$, constructed as described below.

Figure 5 shows the relationship between the C+G% content of the CDS (coding sequence) vs that of the genomic sequence for the genes of $\mathcal{L}$: note the very strong correlation ($\rho = 0.85$) and the tendency for the C+G% of the CDS to be about 5% higher than that of the corresponding genomic region (mean difference: 4.15%).

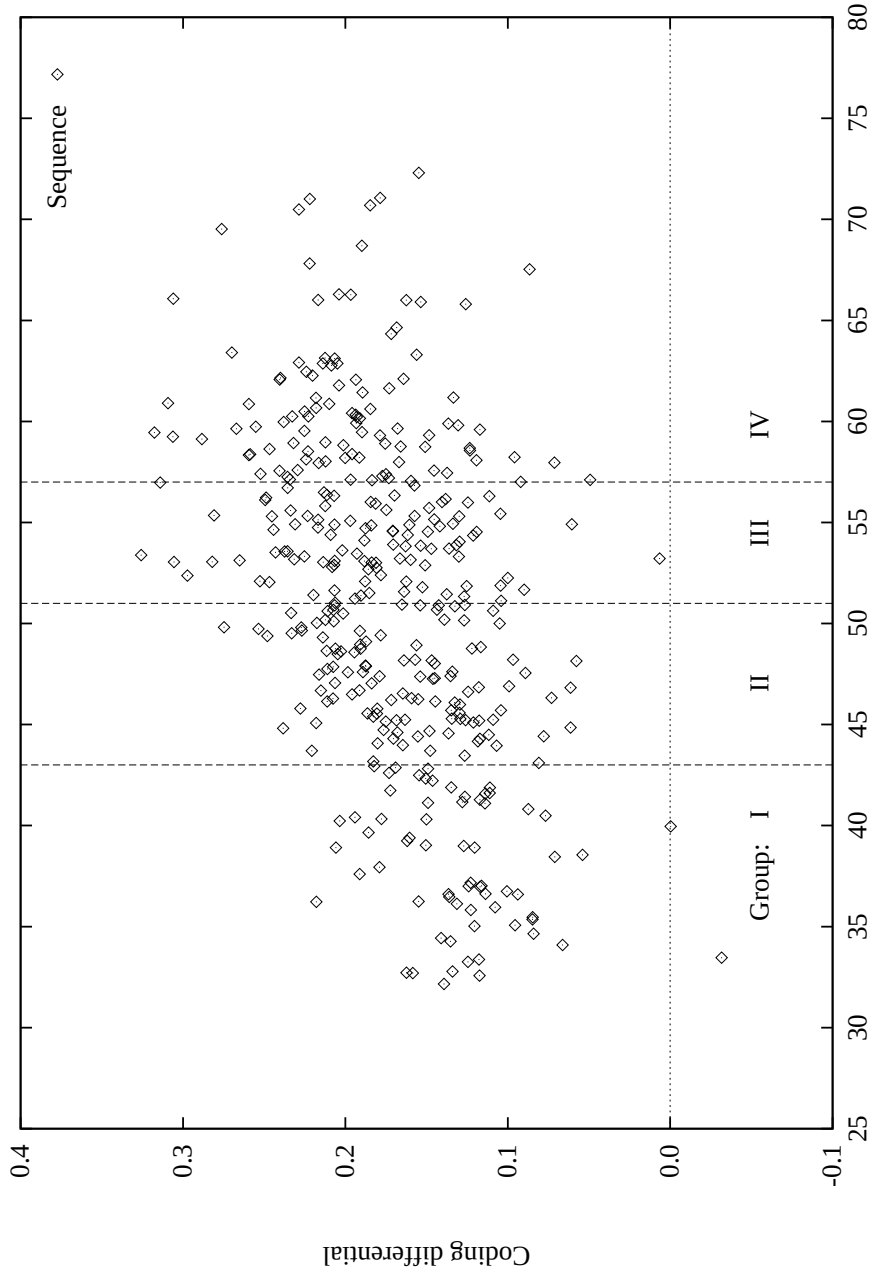


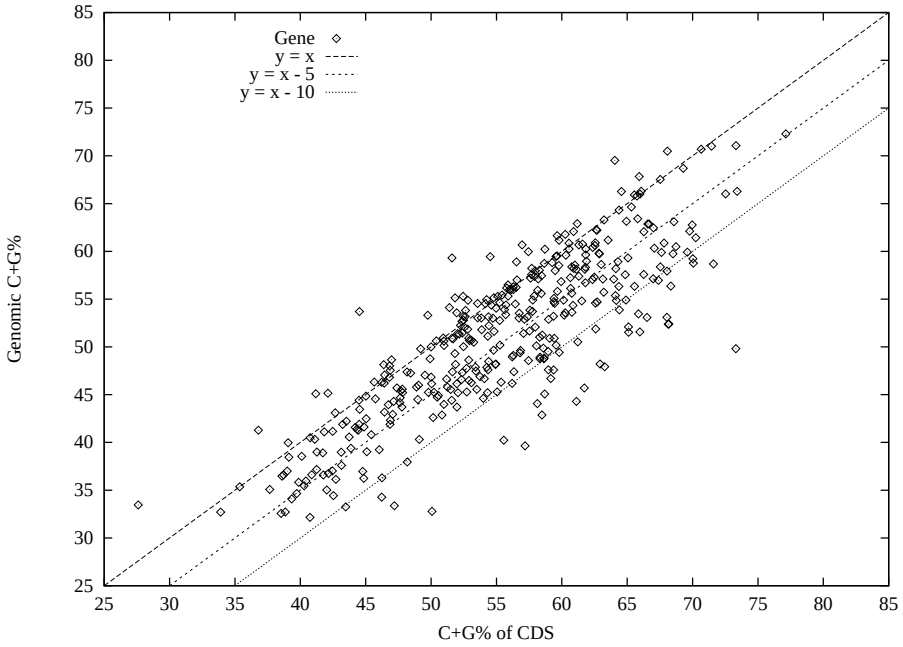
Fig. 4. Coding differential vs C+G% content of learning set genes

Legend. Data for 380 genes of GENSCAN learning set (Appendix A). Vertical dashed lines separate the four C+G% compositional groups described in text.

Specifically, in 75% of cases (96/380), the genomic sequence corresponding to a CDS of C+G% content x had C+G% content in the range [x-10,x]. As a consequence, cDNAs can be categorized into appropriate genomic C+G% compositional groups with reasonable reliability simply by subtracting 5% from the observed C+G content of the cDNA. The set $\mathcal{L}^I_{coding}$ was constructed by combining the coding regions of the group I genes ($< 43\%$ genomic C+G) of $\mathcal{L}$ with the cDNA sequences of less than 48% ($= 43 + 5$) C+G from the set $\mathcal{L}_{cDNA}$. This subset, comprising 638 complete coding sequences, totaled approximately 380,000 codons.

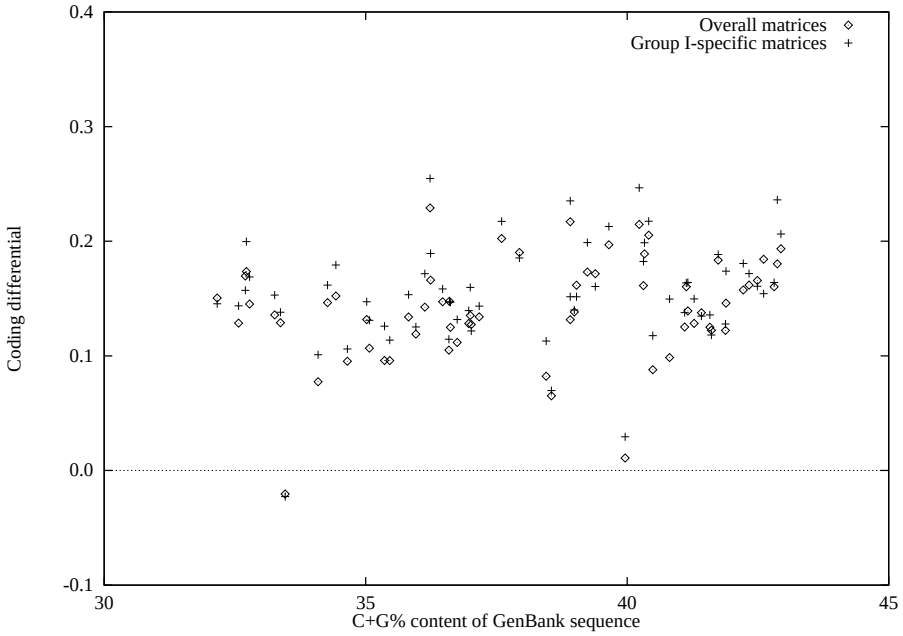
The effect of using the coding/non-coding matrices $C^{(i)}_I$, D_I on the coding differentials of group I genes is illustrated in Fig. 6. The coding differential increases in 53/65 (82%) of the genes of group I, and the mean value increases from 0.130 to 0.143, a 10% increase, suggesting that these group I-specific matrices should provide somewhat better prediction in A+T rich sequences. Performance of GENSCAN using group I-specific vs overall matrices is compared in Section 5.3.2. Derivation of coding and non-coding matrices specifically for other C+G% compositional groups did not result in improved performance (data not shown).

Fig. 5. Genomic C+G% vs CDS C+G% for genes of learning set



Legend. Data for learning set genes. Dashed lines indicate specific C+G% differences.

Fig. 6. Effect of group I-specific matrices on coding differential



Legend. Data for 65 group I genes of GENSCAN learning set (Appendix A).

Chapter 4

BIOLOGICAL SIGNALS

This chapter describes the biological signal models which were constructed, focusing primarily on signals related to pre-mRNA splicing. These signals are undoubtedly the most important elements for accurate prediction of exon boundaries, since all exons begin and/or end with such sites. Section 4.1 briefly reviews current knowledge of the mechanism of constitutive pre-mRNA splicing. The next section discusses models of biological signals in general and the acceptor/branch point signal in particular, introduces a new model for this element, and compares it to two types of models described previously. Section 4.3 discusses the dependencies which exist between positions in donor splice signals and how a new type of model was developed which accounts for many of the most significant of these dependencies. Again, the discriminatory power of this model is compared to models used previously. Finally, the last section discusses transcriptional and translational signals, for which relatively simple models were employed.

4.1 Pre-mRNA splicing

Introns are removed from eukaryotic pre-mRNAs in the nucleus by a complex process catalyzed by a 60S particle known as the spliceosome (e.g., Green, 1991). The spliceosome is composed of five small nuclear RNAs (snRNAs) called U1, U2, U4, U5 and U6, and numerous protein factors. Splice site recognition and spliceosomal assembly

occur simultaneously according to a complex sequence of steps outlined below.[1] The first step appears to be recognition of the donor (5′) splice site at the exon/intron junction: a substantial amount of genetic (e.g., Zhuang & Weiner, 1986, Siliciano & Guthrie, 1988) and biochemical evidence (Heinrichs *et al.*, 1990) has established that this occurs primarily through base pairing with the U1 snRNA over a stretch of approximately nine nucleotides encompassing the last three exonic nucleotides and the first six nucleotides of the intron. The second step in spliceosomal assembly involves recognition of the branch point/acceptor site. This process is more complex, involving binding of U2 auxiliary factor (U2AF) and possibly other proteins to the pyrimidine-rich region immediately upstream of the acceptor site, which directs U2 snRNA binding to the branch point sequence approximately 20 to 40 bp upstream of the intron/exon junction (Green, 1991). The U2 snRNA sequence 3′ GGTG 5′ has been shown to base pair with the branch point signal, consensus 5′ YYRAY 3′, with the unpaired branch point adenosine bulged out of the RNA duplex (Query *et al.*, 1994). Subsequently, a particle containing U4, U5 and U6 snRNAs is added, U5 snRNA possibly interacting with the acceptor site, leading eventually to formation of the mature spliceosome (Konarska & Sharp, 1987).

Splicing itself occurs by two sequential transesterification reactions. First, an unusual 2′-5′ phosphodiester bond (RNA branch) is formed between the 2′ hydroxyl of an adenosine (A) near the 3′ end of the intron (the branch point) and the guanosine (G) at the 5′ end of the intron, resulting in cleavage at the 5′ or donor splice site (exon/intron junction). In the second step, the 3′ or acceptor splice site (intron/exon junction) is cleaved and the two exons are ligated together, causing the intron to be released as an "RNA lariat" which is rapidly degraded *in vivo*. After all introns have been removed, the resulting processed mRNA is exported to the cytoplasm for translation. Despite fairly extensive knowledge of the factors involved in splice site recognition (e.g., McKeown, 1993), the precise mechanisms by which the proper splice sites are distinguished from similar or identical "pseudo-sites" nearby is not well understood.

[1] Two of the most comprehensive reviews of pre-mRNA splicing are Moore *et al.* (1993) and Green (1991); see also McKeown (1992) for a review of alternative splicing.

Table 5. Base composition around intron/exon junctions

a. Branch point region, $[-38, -21]$

Pos	-38	-37	-36	-35	-34	-33	-32	-31	-30	-29	-28	-27	-26	-25	-24	-23	-22	-21
A%	22	20	22	24	21	21	20	22	23	22	21	21	22	23	21	23	20	20
G%	25	26	25	22	23	22	22	21	23	20	20	18	20	16	17	18	17	16
C%	28	28	26	28	28	29	29	29	29	30	30	31	30	31	30	29	31	34
T%	26	27	26	26	28	28	29	28	25	28	28	30	28	31	33	30	32	30
Y%	54	54	52	55	56	57	57	57	55	58	59	61	58	61	63	59	63	64

b. Pyrimidine-rich region, $[-20, -5]$

Pos	-20	-19	-18	-17	-16	-15	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5
A%	20	16	15	14	14	12	9	9	8	8	8	8	8	9	6	7
G%	16	18	18	18	15	12	13	13	12	12	13	13	12	10	6	6
C%	31	32	32	31	35	37	35	34	34	33	33	38	41	41	44	38
T%	34	33	35	37	35	39	42	45	46	47	46	42	39	41	44	48
Y%	65	66	66	68	71	76	78	79	80	80	80	80	80	82	88	87

c. Acceptor site region, $[-4, +3]$

Pos	-4	-3	-2	-1	+1	+2	+3
A%	22	4	100	0	25	25	27
G%	22	0	0	100	52	22	24
C%	33	74	0	0	13	21	27
T%	22	21	0	0	9	32	23
Y%	55	96	0	0	23	53	50

Legend. Compositional data for 1,254 acceptor sites from the 238 multi-exon genes of the learning set (Appendix A). The letter Y indicates either pyrimidine nucleotide (C or T).

4.2 The acceptor / branch point signal

Table 5 displays the base composition at specific positions relative to the intron/exon junction for the 1,254 acceptor splice sites of the learning set. Intron positions are labeled $-38, -37, \ldots$ up to -1 for the last intron nucleotide: exon positions are labeled $+1, +2, +3$. These positions divide fairly naturally into three regions. First, the region $[-38, -21]$ in which the branch point adenosine typically resides is characterized by nearly random (equal) usage of the four nucleotide types, with a weak bias toward pyrimidines (Y = C or T). Second, the region $[-20, -5]$ where U2AF typically binds, is characterized by a pronounced bias toward pyrimidine nucleotides, which increases almost monotonically from 65 to almost 90% immediately upstream of the acceptor site. Third, the region $[-4, +3]$ near the acceptor splice site itself, which is thought to interact with U5 snRNA, U1 snRNA and possibly other factors, displays the motif CAGG, flanked by less conserved positions on either side. Such compositional biases at the acceptor site were first noticed by Breathnach & Chambon (1981) and first systematically tabulated by Mount (1982).

4.2.1 Weight matrix models and generalizations

Numerous models of biological signal sequences such as donor and acceptor splice sites, promoters, etc. have been constructed in the past ten years or so (reviewed in Gelfand, 1995). One of the earliest and most influential approaches to modeling the acceptor splice signal and other biological signals has been the weight matrix method (WMM) introduced by Staden (1984) (but see also Stormo *et al.*, 1982 where a similar idea was introduced). In this approach, nucleotides in a signal of length λ are assumed to be generated independently according to position-specific probability distributions. Under such a model, the probability of generating a particular sequence $X = x_1, x_2, \ldots, x_\lambda$ under the signal model, "+", is given by: $P\{X|+\} = P^+_{WMM}(X) = \prod_{i=1}^{\lambda} p^{(i)}_{x_i}$, where $p^{(i)}_j$ is the probability of generating nucleotide j at position i of the signal, which is typically estimated from the positional frequency $f^{(i)}_j$ observed in a set of aligned signal sequences as in Table 5. Typically, an analogous negative model, "–", corresponding to non-sites is also derived from a set of "pseudo-sites". The probability of generating the sequence X under the negative model will be written $P\{X|-\}$ or $P^-_{WMM}(X)$. Sites may then be discriminated from non-sites by the "signal ratio", $r_{WMM} = P^+_{WMM}(X)/P^-_{WMM}(X)$ or the "signal score", $s_{WMM} = \log_2(r_{WMM})$.

A natural generalization of this method, termed weight array model (WAM), was applied by Zhang & Marr (1993) to model the donor splice signal in the yeast *Schizosaccharomyces pombe*. The WAM model is essentially an inhomogeneous first-order Markov model (Section 2.1) which differs from the WMM model in that it allows for dependencies between adjacent positions. Under this model, the probability of generating the sequence X is: $Pr\{X|+\} = P^+_{WAM}(X) = p^{(1)}_{x_1} \prod_{i=2}^{\lambda} p^{(i-1,i)}_{x_{i-1},x_i}$, where $p^{(i-1,i)}_{j,k}$ is the conditional probability of generating nucleotide k at position i, given nucleotide j at position $i-1$. This quantity is typically estimated from the ratio $f^{(i-1,i)}_{j,k}/f^{(i-1)}_j$, where $f^{(i-1,i)}_{j,k}$ is the frequency of the dinucleotide j,k at positions $i-1, i$ of the signal. An analogous negative model may again be constructed from non-sites: such a model will capture essentially the adjacent nucleotide biases in bulk genomic DNA. Signal ratios and scores may be defined for the WAM model as above.

Both WMM and WAM models of the pyrimidine-rich / acceptor region $[-20, +3]$ were constructed from the acceptor sites of $\mathcal{L}$. These models were then tested on a

disjoint set of 65 human genes, the GENSCAN test set $\mathcal{T}$, described in Appendix B. All segments of length 23 containing the requisite AG dinucleotide at positions $-2, -1$ were scored (segments lacking this AG have probability zero under both models and so need not be considered). The distributions of scores for the true and false splice sites in these sequences under these two models are shown in Fig. 7a,b. Comparison of these two figures shows that the WAM model gives clearly superior separation between true and false sites (see also Table 7). What additional features of the signal does the WAM model capture? The differences are most easily seen by comparing the signal ratios for the two models:

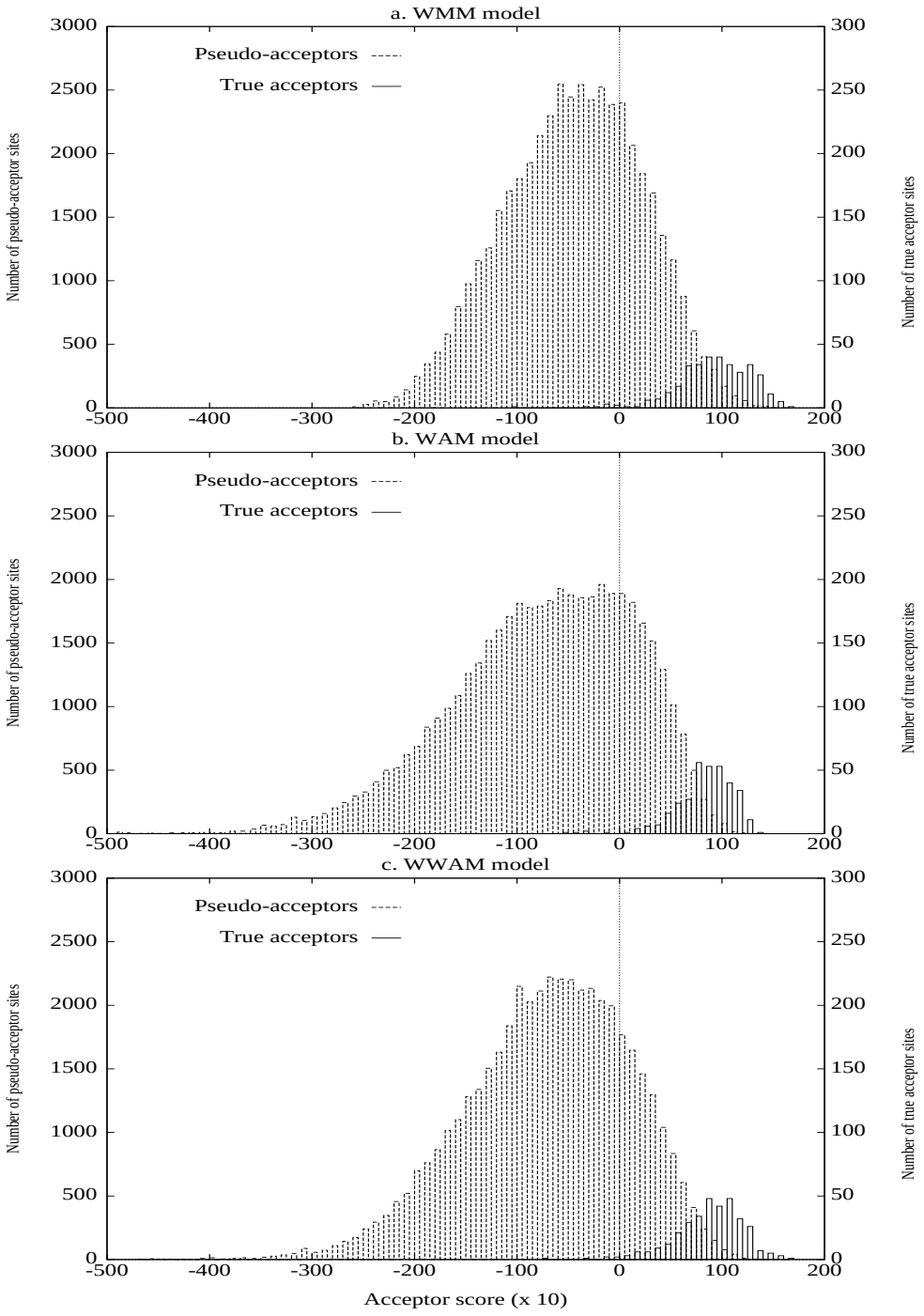
$$[13] \qquad r_{WAM}/r_{WMM} = \frac{P^+_{WAM}(X)}{P^-_{WAM}(X)} / \frac{P^+_{WMM}(X)}{P^-_{WMM}(X)} = \prod_{i=2}^{\lambda} r^{(i-1,i)}_{x_{i-1},x_i} / \hat{r}^{(i-1,i)}_{x_{i-1},x_i} \ ,$$

where $r^{(i)}_{x_{i-1},x_i} = \frac{f^{(i-1,i)}_{j,k}}{f^{(i-1)}_j f^{(i)}_k}$ is the "positional odds ratio" of the dinucleotide j,k ending at position i of the signal and $\hat{r}^{(i)}_{x_{i-1},x_i}$ is the corresponding ratio for non-sites, which will be essentially equal to the "global odds ratio" for the dinucleotide j,k in bulk genomic DNA (e.g., Karlin & Burge, 1995).

4.2.2 Positional odds ratios

Figure 8a shows the positional odds ratios in the branch point/pyrimidine-rich region for all YR and YY dinucleotides (R = A or G), compared to the corresponding global odds ratios (dashed lines) for the genes of the GENSCAN learning set. Several features are of note. First, though many dinucleotides exhibit biases (positional odds ratios different from one), most of these ratios are quite similar to those observed globally in genomic DNA. For example, the CG dinucleotide exhibits very strong negative biases, with odds ratios in the range of approximately 0.25 to 0.45, but this range is centered on the typical global value of 0.36. Thus, it appears that the dinucleotide odds ratios are in some way an intrinsic property of the genome or "genomic signature" (Karlin & Burge, 1995), which tend to gravitate toward typical genomic values even in the presence of fairly strong selective pressures (i.e. the presumably

Fig. 7. Comparison of acceptor splice site models



Legend. Data for sites in genes of GENSCAN test set (Appendix B).

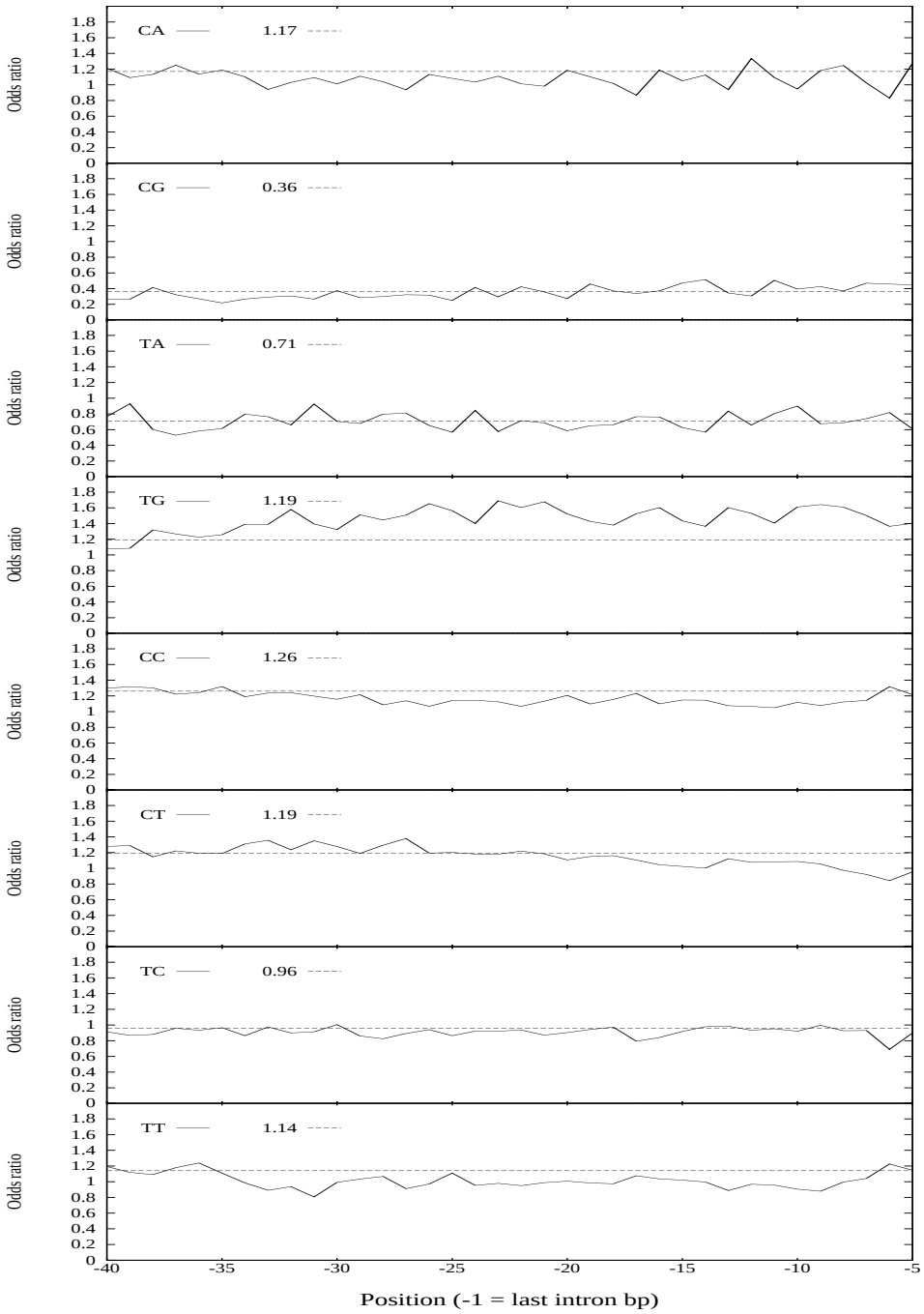
strong selection for pyrimidines in the region $[-20,-5]$). The only doublet in this figure which exhibits positional odds ratios consistently different from the global value is TG, with ratios typically in the range of 1.4 to 1.6, higher than the global average value of 1.19 — the reason for this excess of TG dinucleotides is not immediately apparent.

The positional odds ratios for the RR and RY dinucleotides are displayed in Fig. 8b. Interestingly, three of the four RR doublets exhibit fairly dramatic over- or under-representation relative to their typical genomic values in the region $[-20,-5]$ (but not in the branch point region). In particular, the AG doublet is dramatically avoided in this region. To appreciate just how low the frequency of AG is in this region, note (Table 5) that the product $f_A^{(i-1)} f_G^{(i)}$ is typically around 0.01 (1%) in this region, so that the positional odds ratios of 0.0 to 0.4 observed for AG over much of this region imply that its frequency is not more than 0.004 (0.4%), i.e. almost completely absent (see also Senapathy *et al.*, 1990). The strength of this avoidance suggests that occurrence of an AG in the region $[-20,-5]$ might be extremely deleterious for splicing. The most likely explanation is that, at the time of acceptor site definition, the splicing machinery chooses the first available AG site downstream ($3'$) of the branch point so that presence of an AG in the region $[-20,-5]$ will lead to incorrect acceptor site choice, with probably serious consequences for the translation product of the incorrectly spliced mRNA. There is a significant amount of experimental evidence in support of this idea (e.g., Reed, 1989, Smith *et al.*, 1989, Zhuang & Weiner, 1990). Thus, while both the WMM and WAM models capture the bias toward pyrimidine nucleotides in $[-20,-5]$ and the preferred pattern CAGG at $[-3,+1]$, the improved discrimination observed for the WAM model apparently relates to its ability to capture the biases away from AG and towards AA, GG and TG dinucleotides in the pyrimidine-rich region.

4.2.3 The branch point region

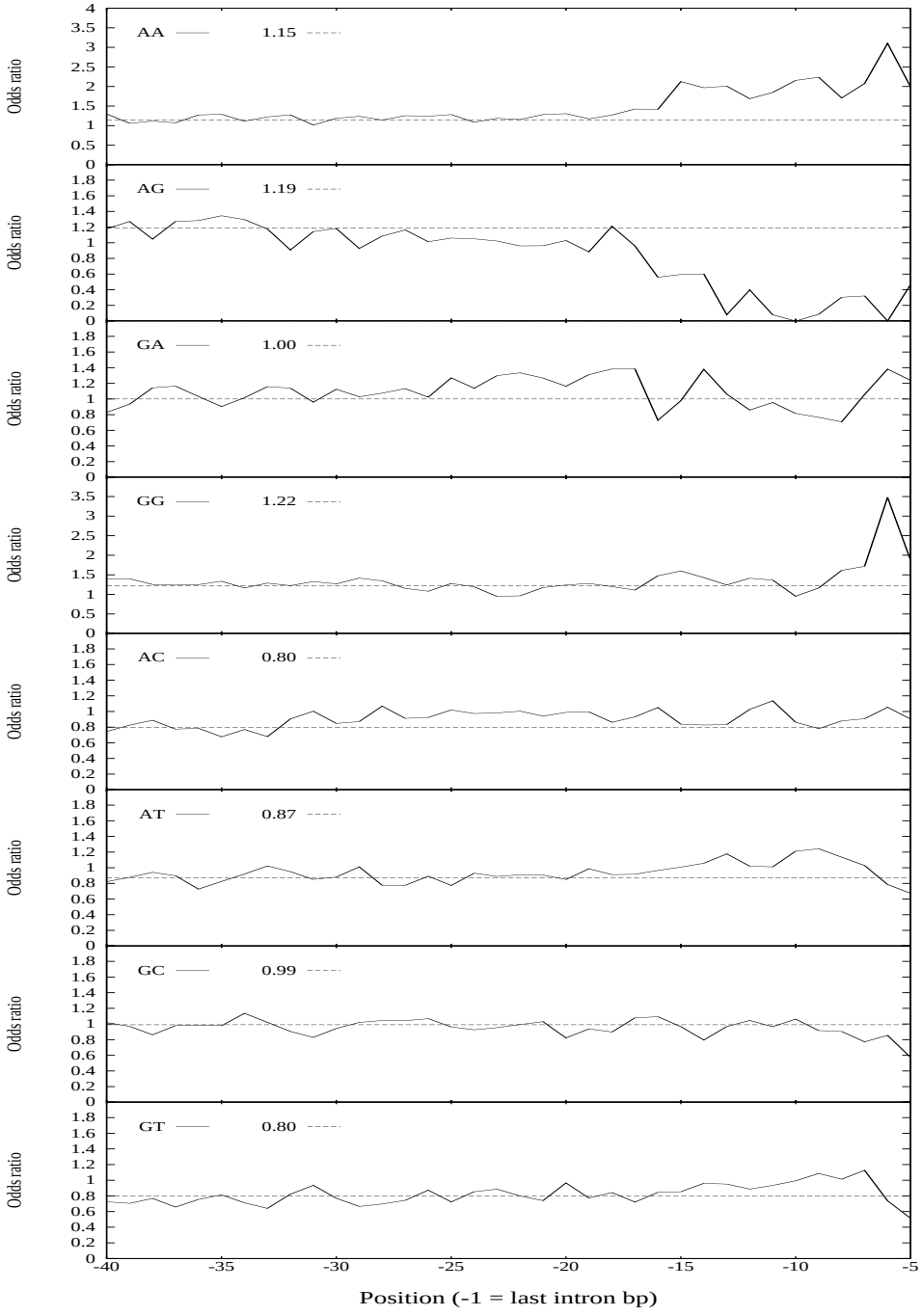
Interestingly, the branch point region $[-38,-21]$ exhibits very few biases in terms of nucleotide (Table 5) or dinucleotide composition (Figs. 8a,b) and, as expected, WMM and WAM models encompassing this region give little or no extra discriminatory

Fig. 8a. Positional odds ratios of YN doublets near acceptor sites



Legend. Data for genes of GENSCAN learning set (Appendix A).

Fig. 8b. Positional odds ratios of RN doublets near acceptor sites



Legend. Data for genes of GENSCAN learning set (Appendix A).

power (data not shown). This lack of positional bias is presumably due at least in part to the variable position of the branch point signal in this region. One approach toward modeling this signal would be to locate and align the branch point signals from each sequence, as was done for the acceptor site. However, this approach is not feasible in practice since branch points are almost never annotated in GenBank sequences (because sequencing laboratories very rarely perform the extra experiments required to localize the branch point). Nor is searching for the branch point consensus, YYRAY, a reliable way of locating branch points, e.g., Harris & Senapathy (1990) found only a very weak tendency toward the branch point consensus in this region. Consistent with these findings, only 30% of the introns in the set $\mathcal{L}$ contained a YYRAY pentamer in the appropriate region, $[-40, -21]$. Thus, any model which required the presence of even such a weak consensus pattern would tend to miss most true sites.

How then to proceed? One approach would be to develop a higher-order WAM model capturing second-order (triplet), third-order (tetramer), or even fourth-order (pentamer) biases at particular positions in the branch point region. While such a model might work in principle, serious problems arise in estimating the increased number of parameters in such models. For example, construction of even a second-order model requires estimation of four transition probabilities conditional on each doublet at each position. For the set of 1,254 acceptor sequences from $\mathcal{L}$, most doublets occur about $n = 1,254/16 = 78$ times per position and some doublets occur significantly less often, e.g., TA occurs about $0.71 \times 78 = 55$ times per position (Fig. 8a) and CG has even lower counts. Unfortunately, this is not sufficient data to reliably estimate model parameters, as is discussed below.

4.2.4 Parameter estimation error

If a sample of size n is randomly chosen from a large population (of sequences), then the count n_i of nucleotide i will be distributed binomially, with mean $\mu_i = np_i$ and variance $\sigma_i^2 = np_i(1 - p_i)$, where p_i is the (true) frequency of nucleotide i in the original population. Therefore, the typical estimation error made in using the

Table 6. Estimation error versus frequency and sample size: $E = \sqrt{\frac{1-p_i}{p_i n}}$

	Sample size, n						
p_i	30	50	100	175	300	500	1000
0.50	18.3%	14.1%	10.0%	7.6%	5.8%	4.5%	3.2%
0.25	31.6%	24.5%	17.3%	13.1%	10.0%	7.7%	5.5%
0.15	43.5%	33.7%	23.8%	18.0%	13.7%	10.6%	7.5%
0.10	54.8%	42.4%	30.0%	22.7%	17.3%	13.4%	9.5%

Legend. Parameter estimation error was calculated as described in text.

observed fraction $f_i = \frac{n_i}{n}$ to estimate p_i will be on the order of $E_i = \frac{\sigma_i}{\mu_i} = \sqrt{\frac{1-p_i}{p_i n}}$. In particular, the error increases with decreasing p_i but decreases (of course) with increasing sample size, n. Values of this error measure for typical ranges of p_i and n are given in Table 6. Thus, estimation errors are quite large for small sample sizes (e.g., $n = 30$ to 50 or less), but become tolerable (in the range of 10 to 20% or so) at around $n = 175$ to 300, depending on the value of p_i. In particular, insufficient data is available to derive a reliable WAM model of order two or higher.

4.2.5 A windowed weight array model

In order to have enough data to describe potential higher-order biases, the approach I chose was to pool data from a "window" of adjacent signal positions, constructing what might be termed a "windowed weight array model" (WWAM). A second-order WWAM was therefore constructed in which data from positions $i-2$, $i-1$, i, $i+1$ and $i+2$ were averaged to give the second-order Markov transition probabilities at position i for $-38 \leq i \leq -21$. This averaging resulted in typical sample sizes of $n = 5 \times 1,254/16 = 392$ per position, which should be sufficient to give reliable parameter estimates (see Table 6). An analogous negative model was also constructed. The score distribution observed for the model derived by combining this branch point WWAM with the WAM derived previously for positions $[-21, +3]$ is shown in Fig. 7c. Further improvement in discrimination is clearly apparent relative to the other two models.

Table 7. Specificity vs sensitivity for acceptor splice signal models

Model	Sensitivity level							
	20%		50%		90%		95%	
	FP	Sp	FP	Sp	FP	Sp	FP	Sp
WMM	68	50.7%	629	21.6%	3,983	7.1%	6,564	4.7%
WAM	65	51.5%	392	30.9%	3,322	8.4%	5,493	5.5%
WWAM	48	58.6%	343	33.8%	3,170	8.8%	5,397	5.6%

Legend. Data for genes of the GENSCAN test set (Appendix B).

A somewhat more systematic way to compare the discriminatory power of the three models is to compare the number of "false positives" (FP) observed at several levels of sensitivity. For a particular sensitivity level p, the score threshold s_p is chosen as the minimum score for which at least $p\%$ of true sites have score $\geq s_p$. Thus, the number of true positives (TP) at this level is approximately pN, where N is the total number of true sites. The number of false sites with score $\geq s_p$ is then determined: improved discrimination corresponds to fewer false positives at a given sensitivity level. Discriminative power is typically measured by either the total number (or percentage) of false positives (FP) or by the specificity, $Sp = TP/(TP + FP)$, at the given sensitivity threshold. These values are tabulated for the three models at selected sensitivity levels in Table 7. The test set $\mathcal{T}$, of total length 600,104 bp, contains a total of $N = 338$ true acceptor sites. Notably, the WWAM model gives consistently lower numbers of false positives (and higher levels of specificity) at each sensitivity level than the WMM or WAM models. A third-order WWAM in which data from the entire branch point region $[-40, -21]$ were used to estimate third-order transition probabilities for all positions in the range $-40 \leq i \leq -21$ was also constructed, as well as several other variations on this theme, but none gave improvement over the second-order WWAM described above.

What accounts for the improved discrimination of the WWAM branch point / acceptor model? Though the patterns of trinucleotide biases in this region are fairly complex, one way to detect triplets which may be particularly favored or disfavored

is to consider the "triplet positional odds ratio",

$$[14] \qquad r^{(i)}_{x,y,z} = f^{(i-2,i-1,i)}_{x,y,z} \quad / \quad f^{(i-2)}_{x} \frac{f^{(i-2,i-1)}_{x,y}}{f^{(i-2)}_{x}} \frac{f^{(i-1,i)}_{y,z}}{f^{(i-1)}_{y}} = \frac{f^{(i-2,i-1,i)}_{x,y,z} f^{(i-1)}_{y}}{f^{(i-2,i-1)}_{x,y} f^{(i-1,i)}_{y,z}}$$

i.e. the ratio of the frequency of the triplet x, y, z ending at position i to its expected frequency under the first-order WAM model. Though such triplet odds ratios fluctuate more than the corresponding dinucleotide ratios (due to smaller sample sizes) and are difficult to evaluate individually, significant deviations which persist across the branch point region can be detected by analyzing the signs of the differences $d_i = r^{(i)}_{x,y,z} - 1$ for a particular triplet. Specifically, for four out of the 64 possible triplets, CAG, CTG, TAA and TTT, all 20 d_i values in the range $[-40, -21]$ were positive. For five triplets, CAA, CTA, CTT, TAG and TTG, all d_i were negative in this range. This consistency is highly statistically significant by the sign test, even considering that 64 tests were performed, since $2^{-20} < 10^{-6}$. The favored triplets CTG and TAA are probably related to the branch point itself, forming portions of the YYRAY consensus. The CAG triplet might represent an alternative branch point pattern, YYRRY or YYARY, since under certain circumstances either of the purine nucleotide positions in the branch signal are capable of acting as the branch nucleophile (Query *et al.*, 1994). The TTT triplet may represent a subclass of acceptor sites whose pyrimidine-rich tracts are longer or more distally located than usual. Notably, all of the avoided triplets are a single transition mutation away from at least one favored triplet.

Of course, none of the models considered above treats the possibility of longer-range interactions, e.g., between the pyrimidine-rich region and the bases around the acceptor site. One way to measure such biases would be to perform χ^2 tests between the variables X_i and X_j (indicating the nucleotides at positions i and j of the acceptor site) for different i, j pairs. However, in many cases the contingency table expected values become too small (e.g., less than 10) for such tests to be reliable. Therefore, a more robust approach is to test the independence of the consensus indicator C_i (1 if the nucleotide at position i matches the consensus at i, 0 otherwise) and the variable X_j. Such tests were performed between the indicators C_{-3} (consensus: C) and C_{+1} (consensus: G) and the set of nucleotide variables X_j, $-19 \leq j \leq -5$. Most of the

positions in this range were found to be strongly dependent on the variable C_{-3}, but independent of C_{+1} (data not shown). Closer scrutiny of the data showed that this dependence is primarily the result of a positive association between usage of T at position -3 and increased T vs C usage in the pyrimidine-rich region. Partitioning the genes according to C+G content (as in Chapter 3) caused these dependencies to largely disappear, suggesting that the dependence is primarily due to mutational forces acting differentially on different C+G% compositional regions of the genome rather than on factors directly related to splicing. For this reason, further models accounting for these dependencies were not developed.

4.3 The donor splice signal

The donor splice signal comprises the last 3 exonic nucleotides (positions -3 to -1) and the first 6 nucleotides of the succeeding intron (positions $+1$ through $+6$), with consensus[2] sequence [c/a]AGGT[a/g]AGt. The GT dinucleotide at positions $+1$, $+2$ is essentially invariant, with only a small number of exceptions known.[3] Most previous probabilistic models of these sites have assumed either independence between positions, e.g., the WMM model of Staden (1984) or dependencies between adjacent positions only, e.g., the WAM model of Zhang & Marr (1993). However, highly significant dependencies exist between *non-adjacent* as well as adjacent positions in the donor splice signal (see below), which are not adequately accounted for by such models and which likely relate to details of donor splice site recognition by U1 snRNP and possibly other factors. For the reasons indicated in the previous section, I focused on dependencies between the consensus indicator C_i and nucleotide variable X_j, rather than on X_i vs X_j comparisons.

Table 8 shows the χ^2 statistics for the variables C_i vs X_j for all pairs i, j with $i \neq j$ in the set of donor sites from the genes of the learning set (positions $+1$ and $+2$ are omitted since they do not exhibit variability in this data set). Strikingly, almost three fourths (31/42) of the i, j pairs exhibit significant χ^2 values even at the relatively

[2]Uppercase letters indicate nucleotides with frequency $> 50\%$ — see Fig. 9.
[3]Donor sites lacking the GT are not considered by the model.

Table 8. Dependencies between positions in human donor splice sites: χ^2 statistic for consensus indicator C_i vs nucleotide X_j.

Pos.		Position j							
i	Consensus	-3	-2	-1	$+3$	$+4$	$+5$	$+6$	Sum
-3	c/a	—	**61.8**	14.9	5.8	**20.2**	11.2	**18.0**	**131.8**
-2	A	**115.6**	—	**40.5**	**20.3**	**57.5**	**59.7**	**42.9**	**336.5**
-1	G	15.4	**82.8**	—	13.0	**61.5**	**41.4**	**96.6**	**310.8**
$+3$	a/g	8.6	**17.5**	13.1	—	**19.3**	1.8	0.1	**60.5**
$+4$	A	**21.8**	**56.0**	**62.1**	**64.1**	—	**56.8**	0.2	**260.9**
$+5$	G	11.6	**60.1**	**41.9**	**93.6**	**146.6**	—	**33.6**	***387.3**
$+6$	t	**22.2**	**40.7**	**103.8**	**26.5**	**17.8**	**32.6**	—	**243.6**

Legend. For each pair of positions $\{i, j\}$ with $i \neq j$, a 2×4 contingency table was constructed for the consensus indicator variable C_i (see text) vs the variable X_j identifying the nucleotide at position j. The consensus nucleotide(s) at each position i are shown in the second column: the invariant positions $+1, +2$ are omitted. For each contingency table, the value of the χ^2 statistic was calculated and is listed in the table above. Those values exceeding 16.3 ($P < 0.001$, 3 df) are displayed in boldface. The last column in the table lists the sum of the values in each row, which is a measure of the dependence between C_i and the vector $\vec{x}_i$ of the nucleotides at the six remaining positions $j \neq i$. All values exceeded 42.3 ($P < 0.001$, 18 df) and so are displayed in boldface: the largest value, for G at position $+5$, is indicated by *.

stringent level of $P < 0.001$, indicating a great deal of dependence between positions in the donor splice site. (The stringent P-value cutoff was used to compensate for the effect of multiple comparisons.) It is also noteworthy and perhaps surprising that many non-adjacent pairs of positions as well as most adjacent pairs exhibit significant dependence, e.g., positions -1 and +6, separated by 5 intervening nucleotides, exhibit the extremely high χ^2 values of 103.8 for C_6 vs X_{-1} and 96.6 for C_{-1} vs X_6. In order to account for such dependencies in a natural way, a new model-building procedure was developed.

4.3.1 Maximal dependence decomposition (MDD)

The goal of the MDD procedure is to generate, from an aligned set of signal sequences of moderate to large size (i.e. at least several hundred or more sequences), a model which captures the most significant dependencies between positions (allowing for non-adjacent as well as adjacent dependencies), essentially by replacing unconditional WMM probabilities by appropriate conditional probabilities provided that sufficient data is available to do so reliably. Given a data set D consisting of n aligned sequences of length λ, the first step is to assign a consensus nucleotide or nucleotides at each position. For each pair of positions, the χ^2 statistic is calculated for C_i vs X_j (as above) for each i, j pair with $i \neq j$. If no significant dependencies are detected (for an appropriate P-value), then a simple WMM should be sufficient. If significant dependencies are detected, but they are exclusively or predominantly between adjacent positions, then a WAM model may be appropriate.

If, however, there are strong dependencies between non-adjacent as well as adjacent positions (as was observed Table 8), then the following procedure is carried out.

1) Calculate, for each position i, the sum $S_i = \sum_{j \neq i} \chi^2(C_i, X_j)$, which is a measure of the amount of dependence between the variable C_i and the nucleotides at the remaining positions of the site (the row sums in Table 8).

2) Choose the value i_1 such that S_{i_1} is maximal and partition D into two subsets: D_{i_1}, all sequences which have the consensus nucleotide(s) at position i_1; and

$D_{\overline{i_1}}$ ($= D \setminus D_{i_1}$), all sequences which do not.

Now repeat steps 1) and 2) on the subsets, D_{i_1} and $D_{\overline{i_1}}$ and on subsets thereof, and so on, yielding a binary subdivision "tree" with (at most) $\lambda - 1$ levels (see Fig. 9). This process of subdivision is carried out successively on each branch of the tree until one of the following three conditions occurs:

(i) The $(\lambda - 1)$'th level of the tree is reached (so that further subdivision is impossible);

(ii) No significant dependencies between positions in a subset are detected (so that further subdivision is not indicated); or

(iii) The number of sequences remaining in a subset falls below a preset minimum value m so that reliable WMM frequencies could not be determined after further subdivision.

Finally, separate WMM models are derived for each subset of the tree, and these are combined to form a composite model as described below.

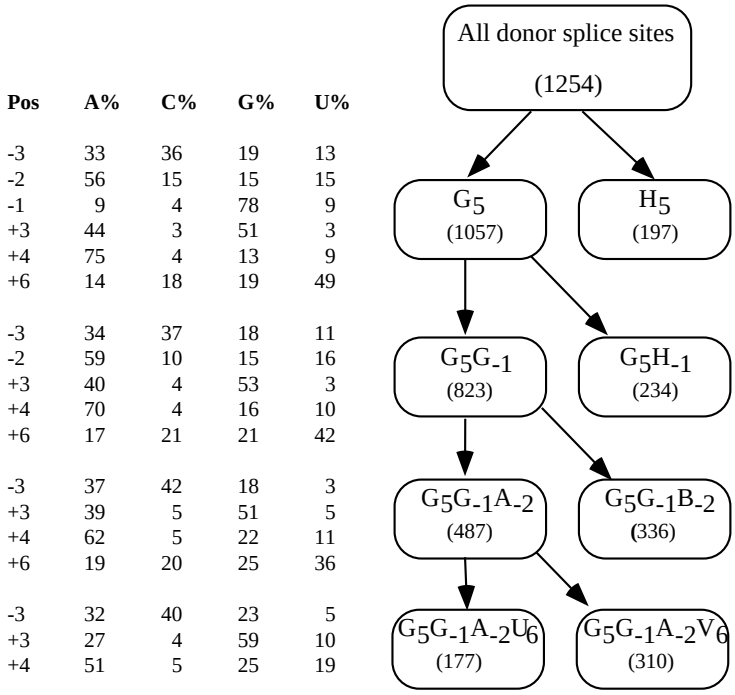
Figure 9 illustrates the MDD procedure applied to the set of 1,254 donor splice sites from $\mathcal{L}$. The initial subdivision is made based on the consensus (G) at position 5 of the donor signal (see Table 8), resulting in subsets G_5 and H_5 (H meaning A, C or U) containing 1,057 and 197 sequences, respectively. Based on the data of Table 6, the value $m = 175$ was chosen as a reaonable minimum sample size (giving typical parameter estimation errors in the range of 7 to 23%), so the set H_5 is not further divided. The subset G_5 is sufficiently large, however, and exhibits significant dependence between positions (data not shown), so it is further subdivided according to the consensus (G) at position -1, yielding subsets G_5G_{-1} and G_5H_{-1}, and so on.

The composite model for generation of signal sequences is then essentially a recapitulation of the subdivision procedure, as described below for the particular case of the donor signal.

0) The (invariant) nucleotides X_1 and X_2 are generated.

1) X_5 is generated from the original WMM for all donor sites combined.

Fig. 9. Maximal dependence decomposition model of human donor splice signal



Pos	A%	C%	G%	U%		Pos	A%	C%	G%	U%
-3	33	36	19	13		-3	35	44	16	6
-2	56	15	15	15		-2	85	4	7	5
-1	9	4	78	9		-1	2	1	97	0
+3	44	3	51	3		+3	81	3	15	2
+4	75	4	13	9		+4	51	28	9	12
+6	14	18	19	49		+6	22	20	30	28
-3	34	37	18	11		-3	29	31	21	18
-2	59	10	15	16		-2	43	30	17	11
+3	40	4	53	3		+3	56	0	43	0
+4	70	4	16	10		+4	93	2	3	3
+6	17	21	21	42		+6	5	10	10	76
-3	37	42	18	3		-3	29	30	18	23
+3	39	5	51	5		+3	42	1	56	1
+4	62	5	22	11		+4	80	4	8	8
+6	19	20	25	36		+6	14	21	16	49
-3	32	40	23	5		-3	39	43	15	2
+3	27	4	59	10		+3	46	6	46	3
+4	51	5	25	19		+4	69	5	20	7

All sites:	----------	**Position**	----------					
Base	**-3**	**-2**	**-1**	**+1**	**+2**	**+3**	**+4**	**+5**	**+6**
A%	**33**	**60**	8	0	0	**49**	**71**	6	15
C%	**37**	13	4	0	0	3	7	5	19
G%	18	14	**81**	**100**	0	**45**	12	**84**	20
U%	12	13	7	0	**100**	3	9	5	**46**

U1 snRNA:	**3'**	**G**	**U**	**C**	**C**	**A**	**U**	**U**	**C**	**A**	**5'**

Legend. Subclassification of the donor sites of the learning set by the MDD procedure is illustrated. Each box represents a subset of donor sites corresponding to a pattern of matches/mismatches to the consensus nucleotide(s) at a set of positions, e.g., G_5G_{-1} is the set of donors with G at positions +5 and -1. H indicates A, C or U; B indicates C, G or U; and V indicates A, C or G. The number of sites in each subset is given in parentheses. The frequencies (percentages) of the four nucleotides at each variable position are indicated for each subset immediately adjacent to the corresponding box. Data for the entire set of 1254 donor sites are given at the bottom of the figure: frequencies of consensus nucleotides are shown in boldface. The sequence near the 5' end of U1 snRNA which has been shown to base-pair with the donor site is shown below in 3' to 5' orientation.

Table 9. Specificity vs sensitivity for donor splice signal models

Model	Sensitivity level							
	20%		50%		90%		95%	
	FP	Sp	FP	Sp	FP	Sp	FP	Sp
WMM	68	50.0%	368	32.0%	2954	9.4%	4185	7.1%
WAM	79	49.6%	350	33.0%	2160	12.4%	4153	7.2%
MDD	59	54.3%	307	36.0%	1985	13.4%	3382	8.7%

Legend. Data for genes of the GENSCAN test set (Appendix B).

2a) If $X_5 \neq G$, then the (conditional) WMM model for subset H_5 is used to generate the nucleotides at the remaining positions in the donor site;

2b) If $X_5 = G$, then X_{-1} is generated from the (conditional) WMM model for the subset G_5.

3a) If ($X_5 = G$ and) $X_{-1} \neq G$, then the WMM model for subset $G_5 H_{-1}$ is used;

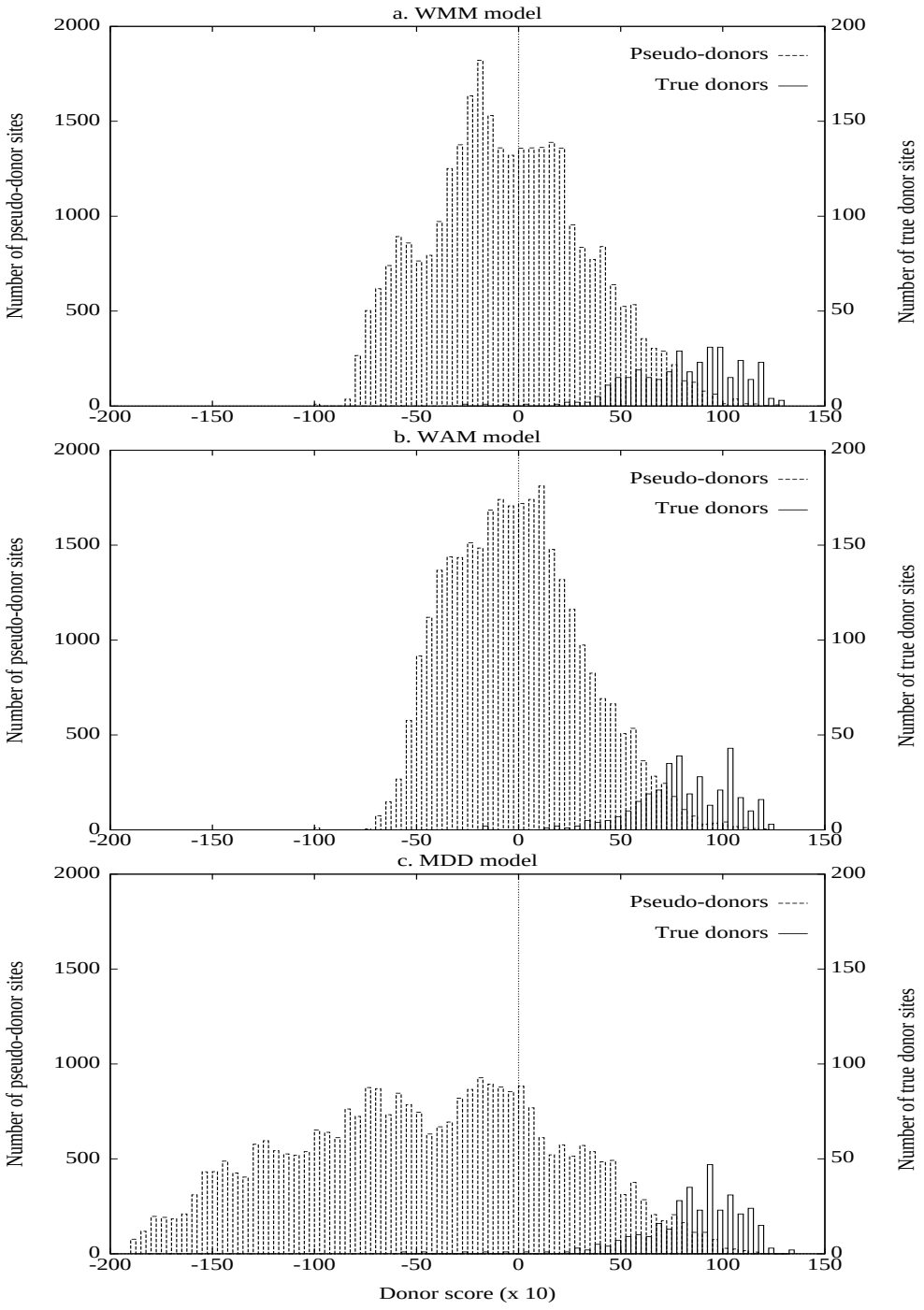
3b) If ($X_5 = G$ and) $X_{-1} = G$, X_{-2} is generated from the model for $G_5 G_{-1}$.

4) ...and so on, until the entire 9 bp sequence has been generated.

The discriminative power of the MDD model is compared to that of the previously used WMM and WAM models in Fig. 10, as tested on the GENSCAN test set $\mathcal{T}$ containing 338 donor splice sites. As for the acceptor site models, the number of false positives (FP) was determined at various levels of sensitivity; these data are given in Table 9. Fig. 10 and Table 9 demonstrate quite significant improvements in discriminative power for the MDD model vs the other two types of models. Most notably, the MDD model appears much better able to distinguish (and give very low scores to) sites with poor potential as donor signals from the remainder of sites (Fig. 10c).

Aside from the improvement in predictive ability demonstrated above, the MDD procedure may also lend insight into the mechanism of donor splice signal recognition. Specifically, the data of Fig. 9 suggest some fairly subtle properties of the U1:donor signal interaction, namely:

Fig. 10. Comparison of donor splice signal models



Legend. Data for sites in genes of GENSCAN test set (Appendix B).

(i) A $5'/3'$ *compensation effect*, in which matches to consensus nucleotides at nearby positions on the same side of the intron/exon junction are positively associated, while poor matching on one side of the junction is almost always compensated by stronger matching on the other side;

(ii) An *adjacent base-pair effect*, in which base pairs at the edge of the donor splice site appear to form only in the presence of adjacent base pairs; and

(iii) A G_3 *preference effect*, in which G is preferred at position +3 only for a subclass of strongly U1-binding donor sites.

The evidence for each of these effects is summarized below.

$5'/3'$ *compensation effect.* First, G_{-1} is almost completely conserved (97%) in H_5 donor sites (those with a non-G nucleotide at position +5) vs 78% in G_5 sites, suggesting that absence of the G:C base pair with U1 snRNA at position +5 can be compensated for by a G:C base pair at position -1, with a virtually absolute requirement for one of these two G:C base pairs (only 5 of 1,254 donor sites lacked both G_5 and G_{-1}). Second, the H_5 subset exhibits substantially higher consensus matching at position -2 ($A_{-2} = 85\%$ in H_5 vs 56% in G_5), while the G_5 subset exhibits stronger matching at positions +4 and +6. Similar compensation is also observed in the G_5G_{-1} vs G_5H_{-1} comparison: the G_5H_{-1} subset exhibits substantially higher consensus matching at positions +6 (76% vs 42%), +4 (93% vs 70%) and +3 (100% R_3 vs 93%). Yet another example of compensation is observed in the $G_5G_{-1}A_{-2}$ vs $G_5G_{-1}B_{-2}$ comparison, with the $G_5G_{-1}B_{-2}$ subset exhibiting increased consensus matching at positions +4 and +6, but somewhat lower matching at position -3. This effect might simply be a consequence of the energetics of RNA helix formation, in which adjacent base pairs contribute greater stability due to favorable base stacking interactions. Another possible interpretation is that this compensation effect results from constraints acting at steps subsequent to the U1:donor interaction. In particular, the two sides (exon and intron) of the donor signal may be recognized separately by U5 and U6 snRNAs, respectively (e.g., Moore *et al.*, 1993), and it could be that at least one of these interactions must be fairly strong for splicing to take place (P. A. Sharp, personal communication).

Adjacent base-pair effect. H_5 splice sites have nearly random (equal) usage of the four nucleotides at position +6, implying that base pairing with U1 at position +6 does not occur (or does not aid in donor recognition) in the absence of a base pair at position +5. The almost random distribution of nucleotides at position −3 of the $G_5G_{-1}B_{-2}$ donor sites also suggests that base pairing with U1 snRNA at position −3 does not occur or is of little import in the absence of a base pair at position −2.

G_3 preference effect. Comparison of the relative usage of A vs G at position +3 in the various subsets reveals interesting features. Perhaps surprisingly, G is almost as frequent as A at position +3 (45% vs 49%) in the entire set of donor sites, despite the expected increased stability of an A:U vs G:U base pair at position +3. Only in subset H_5 is a dramatic preference for A over G at position +3 observed (81% vs 15%), suggesting that only in the absence of the strong G:C base pair at position +5 does the added binding energy of an A:U vs G:U base pair at position +3 become critical to donor site recognition by U1 snRNA. On the other hand, in the most strongly consensus-matching donor site subset, $G_5G_{-1}A_{-2}U_6$, there is actually a strong prefence for G_3 over A_3 (59% vs 27%)! Two possible explanations for this observation seem reasonable: either (a) there is selection to actually weaken the U1:donor interaction in these strongly matching sites so that U1 snRNA can more easily dissociate from the donor site to permit subsequent steps in splicing; or (b) G_3 is preferred over A_3 at some step in splicing subsequent to donor site selection (but this effect is only apparent when the strong constraints of U1 binding are satisfied by consensus matches at many other positions).

In summary, the MDD model not only provides improved discrimination between true and false donor sites by accounting for potentially important non-adjacent as well as adjacent interactions, but may also give some insight into how the donor site is recognized. It may be of interest in the future to apply this method to other biological signals, e.g., transcriptional or translational signals in DNA or even perhaps protein motifs. In many cases, however, this approach will have to be postponed until sufficiently large sets of sequences have accumulated so that complex dependencies can be reliably measured.

4.4 Transcriptional and translational signals

In contrast to the fairly complex models used for splice signals, relatively simple models were used for transcriptional and translation signals. There are two primary reasons for this apparent discrepancy. The first is that accurate detection of splice signals is probably much more important for reliable exon prediction in higher eukaryotes which typically have many introns per gene, so greater effort was invested in modeling these signals. The second reason is that there is much less data available for translational and transcriptional signals than for splice signals, making it difficult or impossible to consider complex dependencies. For example, the learning set $\mathcal{L}$ contains 1,254 introns, hence 1,254 donor and acceptor splice sites, but only 380 genes, hence 380 translation initiation and termination sites and even fewer promoters and poly-adenylation signals since the boundaries of the GenBank sequence often fall within $5'$ and/or $3'$ untranslated regions (and even in cases where the transcription unit is complete, promoter and poly-adenylation signals are not always annotated).

The specific models used are described below. A 6 bp WMM was used for the poly-adenylation signal (consensus: AATAAA) using the GenBank annotated "polyA_signal" features from the sequences of $\mathcal{L}$. A 12 bp WMM model, beginning 6 bp prior to the initiation codon, was used for the translation initiation (Kozak) signal (consensus: gccAcCATGgcg). Few base preferences were detected in the vicinity of the stop codon: for the translation termination signal, then, one of the three stop codons is generated (according to its observed frequency in $\mathcal{L}$) and the next 3 nucleotides are generated according to a WMM. The translation initiation and termination signal models were based on the GenBank "CDS" feature annotation. Similar models of these signals have been used by others (e.g., Guigó *et al.*, 1992, Snyder & Stormo, 1995).

For promoters, a very simplified model was used for what is undoubtedly an extremely complex signal. The primary goal was to construct a model flexible enough so that potential genes would not be missed simply because they lacked a sequence similar to some preconceived notion of what a promoter should look like. Since about 30% of eukaryotic promoters lack an apparent TATA signal, a split model was used in

which a TATA-containing promoter is generated with probability 0.7 and a TATA-less promoter with probability 0.3. The TATA-containing promoter is modeled using a 15 bp TATA-box WMM and an 8 bp cap site WMM, both borrowed from a previous analysis of 502 unrelated eukaryotic promoters (Bucher, 1990). The length between the WMMs is generated uniformly from the range of 14-20 nucleotides, corresponding to a TATA $\rightarrow$ cap site distance of 30-36 bp, from the first T of the TATA-box matrix to the cap site (start of transcription). Intervening bases are generated according to an intergenic-null model, i.e. independently generated from intergenic base frequencies. At present, TATA-less promoters are modeled simply as intergenic-null regions of 40 bp in length. In the future, incorporation of improved promoter models, e.g., perhaps using multiple types of known transcription factor binding sites along the lines of Prestridge's (1995) work, may lead to more accurate promoter recognition.

One other type of signal which was incorporated into the GENSCAN model is the leader or signal peptide (e.g., Randall & Hardy, 1989) which occurs at the N-terminus of secreted, lysosomal and membrane proteins. This signal was modeled using a bipartite "codon-level WMM" (i.e. a WMM which generates a nucleotide triplet at each position rather than a single nucleotide) of nineteen codons in length. The annotated signal peptides (GenBank "sig_peptide" feature) from the genes of the learning set were extracted (78 of 380 $\approx$ 20% of genes had an annotated signal peptide) and used as follows. Codon frequencies for the first four codon positions after the initiation signal were averaged to give the first four columns of the WMM (the basic portion of the signal peptide); and codon frequencies for codon positions 6 through 20 were averaged to give the next 15 columns of the WMM (the hydrophobic portion of the signal peptide).[4] For initial exons and single-exon genes, then, a split model is used to generate the first nineteen codons after the initiation signal: with probability 0.8, these codons are generated using the default 3-periodic fifth-order Markov model; and with probability 0.2, they codons are generated by the composite signal peptide model.[5]

[4]The short stretch hydrophilic stretch which typically follows the long hydrophobic stretch was determined to be too weak and too variable in terms of position to be useful for prediction.

[5]For exons of length $\lambda < 63$ bp, of course, only codons three through $\lambda/3 - 1$ are generated in this fashion.

Chapter 5

IMPLEMENTATION AND TESTING OF GENSCAN

This chapter covers the implementation and testing of the GENSCAN program, addresses some of its strengths and weaknesses, and gives some examples of its application. The first section describes the command-line, email, and web versions of the program and gives examples of the text and graphical output. The next section reviews several measures of the predictive accuracy of gene prediction programs at the nucleotide, exon, and gene levels. Section 5.3 compares the performance of GENSCAN to that of other programs as tested on a large collection of vertebrate sequences and on two smaller sets of human genomic sequences. The following section takes a closer look at the accuracy of the program, addressing the dependence on exon size, gene complexity, exon type, and organism of origin, and discussing the usefulness of the exon probability as a reliability indicator. Finally, the last section describes two of the most interesting applications of the program, namely finding genes in long genomic sequences and prediction of alternative splicing, giving examples of each.

5.1 Implementation of GENSCAN

The GENSCAN program was written in the C programming language (Kernighan & Richie, 1988) in a Unix environment and runs on a Sun workstation under the SunOS

4.x or Solaris operating systems. Since C is a highly portable language, it should be fairly straightforward to adapt the program to run on a PC or Apple Macintosh as well if the need arises. The basic (command-line) version of the program reads two input files: a DNA sequence file in FastA format and a parameter file which contains a complete description of the model including the state transition and initial probabilities, length distributions, and sequence generating models.

5.1.1 Approximations made

In the development of the program, several approximations were made which lead to faster run time, reduced memory usage or other desireable features with little or no effect on the accuracy of the calculations. The most significant of these approximations was to work primarily with "scores", i.e. logarithms of ratios of probabilities, for splice signals, coding/non-coding models, etc. rather than with the raw probabilities. The primary advantage of taking logarithms is that large products of probabilities are converted to (large) sums of log-probabilities, which can lead to substantial computational savings.[1] Specifically, the logarithms (base 2) of appropriate probability ratios were taken and these values multiplied by ten and rounded to the nearest integer: all of the algorithms (Viterbi, forward, backward) were implemented using these scores whenever possible, converting back to probabilities only when necessary. For the poly-adenylation signal, for instance, a matrix of the logarithms (base 2, times 10, rounded) of the probabilities of generating each nucleotide at each position under the positive (signal) model over the negative (non-coding, non-signal) models were used, rather than separate (floating point) weight matrix probabilities for the positive and negative models. In addition to reducing the size of the parameter file by approximately 50%, the main advantage of this approach is that the many large products of (floating point) probabilities which must be calculated become sums of integer-valued scores. Several other approximations were made along similar lines (not described).

[1] The exact amount of time saved depends on the architecture of the computer.

5.1.2 What GENSCAN does

For a given input sequence and parameter matrix, the following sequence of operations is carried out:

1) The input sequence is read, stored in memory, and its C+G% content calculated.

2) The parameter file is read and the set of parameters appropriate to the C+G% content group of the sequence (see Chapter 3) is chosen.

3) Both strands of the sequence are scored using the matrices for coding regions, splice signals, etc. and these values are stored for later use.

4) The Viterbi, forward and backward algorithms are performed (using the precalculated scores) and intermediate values of the α, β and γ arrays (Chapter 2) are stored in memory.

5) For each potential exon, the conditional probability, $P\{\epsilon|S\}$, is calculated using the forward/backward formula (Section 2.8).

Two types of output files are created:

Text output. The locations of the exons in the optimal parse of the sequence together with the corresponding (conceptually translated) peptide sequences are recorded as well as (optionally) all suboptimal exons with conditional probability above a chosen threshold level.

Graphical output. A diagram of the locations of all predicted exons in the sequence is created in PostScript or gif format.

The text and graphical displays of the optimal parse for sequence HSNCAMX1 (GenBank accession # Z29373), the human gene for neural cell adhesion molecule L1, are shown in Figs. 11 and 12, respectively.

The text output is described in detail below (a somewhat briefer explanation accompanies the standard text output of the program).

Fig. 11. GENSCAN text output for GenBank sequence HSNCAMX1

```
Sequence HSNCAMX1 : 16288 bp : 59.53% C+G

Parameter matrix : HumanIso.smat : Isochore 4 (57 - 100% C+G)

PREDICTED OPTIMAL PARSE:

G.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr.. Acc.
---- ---- - ------ ------ ---- -- -- ---- ---- ----- ----- ------ ----

1.01 Init +   1533   1608   76  2  1   86   82   161 0.976  14.70 EXAC
1.02 Intr +   4672   4777  106  2  1  101   73   137 0.998  13.94 EXAC
1.03 Intr +   5015   5217  203  2  2  132   49   356 0.985  36.24 EXAC
1.04 Intr +   6192   6581  390  1  0   85   81   419 0.562  36.91 PART
1.05 Intr +   6871   6982  112  2  1   71   70   136 0.999  10.71 EXAC
1.06 Intr +   7131   7315  185  0  2   89   51   398 0.996  36.84 EXAC
1.07 Intr +   7437   7568  132  1  0  100   84   112 0.999  13.80 EXAC
1.08 Intr +   7708   7851  144  2  0   46   75   319 0.999  27.84 EXAC
1.09 Intr +   8417   8528  112  0  1   93   89   130 0.999  14.21 EXAC
1.10 Intr +   8642   8808  167  2  2   73   77   297 0.946  27.90 EXAC
1.11 Intr +   8911   9067  157  2  1   60   61   101 0.939   4.86 EXAC
1.12 Intr +   9248   9372  125  2  2   53  106   190 0.997  18.70 EXAC
1.13 Intr +   9460   9570  111  2  0  102   52   157 0.619  15.02 EXAC
1.14 Intr +   9817  10014  198  2  0   81  109   246 0.784  26.54 EXAC
1.15 Intr +  10246  10316   71  2  2   78  100   104 0.964  10.65 EXAC
1.16 Intr +  10499  10721  223  1  1   83   74   333 0.960  30.33 EXAC
1.17 Intr +  10798  10961  164  2  2   39   48   104 0.489   2.39 WRNG
1.18 Intr +  11502  11666  165  2  0   21   73   190 0.459  12.15 PART
1.19 Intr +  11870  12071  202  1  1  100  105   288 0.999  31.77 EXAC
1.20 Intr +  12160  12282  123  2  0  108   48   230 0.978  22.73 EXAC
1.21 Intr +  12376  12549  174  2  0  102  107   247 0.998  29.11 EXAC
1.22 Intr +  12666  12785  120  1  0  114   99    86 0.999  13.94 EXAC
1.23 Intr +  12893  13048  156  0  0  120   94   202 0.999  25.17 EXAC
1.24 Intr +  13352  13486  135  0  0   62   68   226 0.974  19.72 EXAC
1.25 Intr +  13820  13892   73  0  1   63   99   112 0.999   9.41 EXAC
1.26 Term +  14475  14706  232  0  1  110   40   484 0.997  42.74 EXAC
1.-- PlyA +  15677  15682    6                                 1.05 ????
```

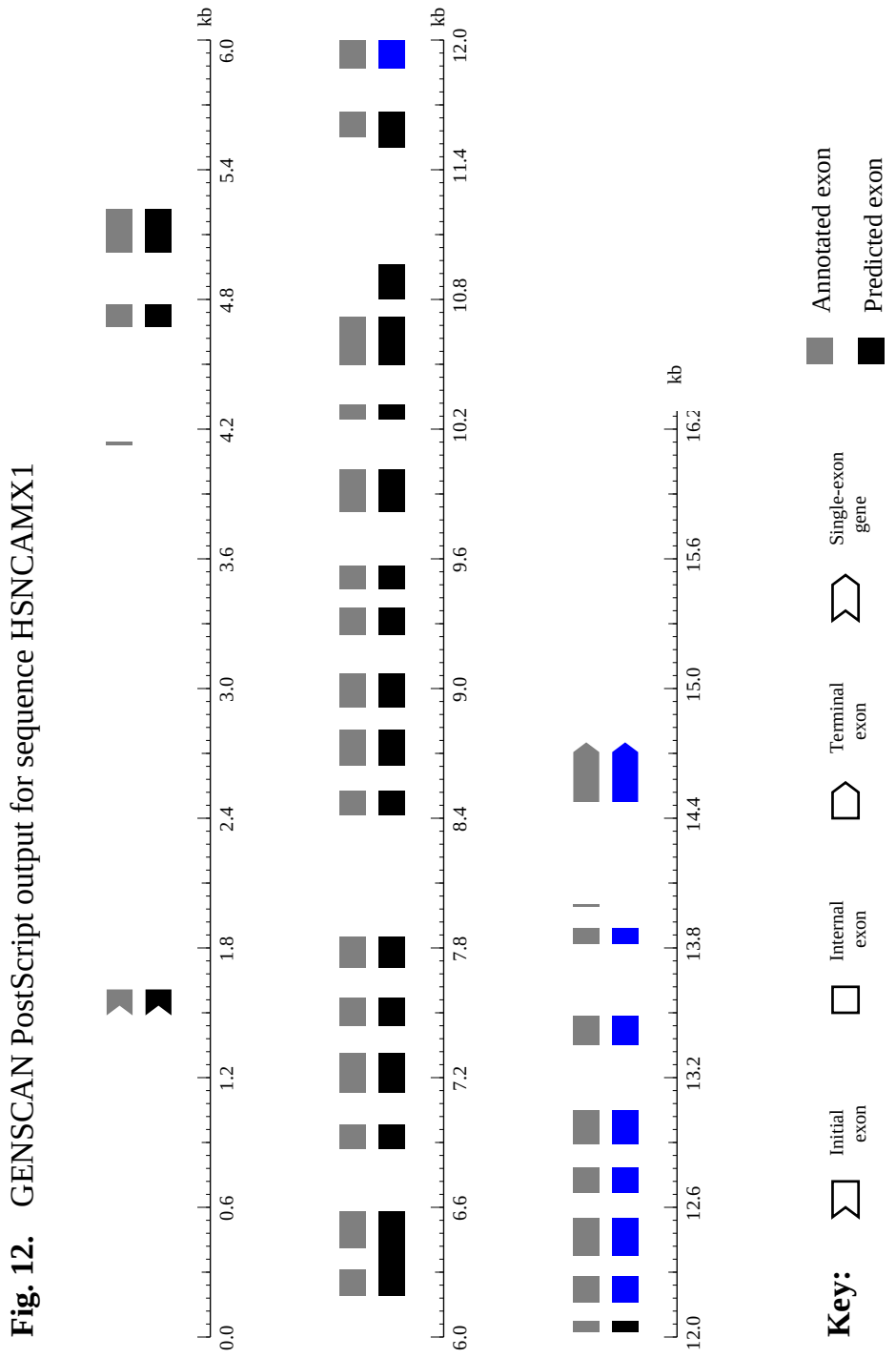


Fig. 12. GENSCAN PostScript output for sequence HSNCAMX1

Col. 1. The gene and exon number of each predicted exon, for reference.[2]

Col. 2. The type of exon or signal: initial exon ("Init"), internal exon ("Intr"), terminal exon ("Term"), single-exon gene ("Sngl"), promoter ("Prom") or polyadenylation signal ("PlyA").[3]

Col. 3. The DNA strand of the predicted feature: "+" for the input strand, "–" for the complementary strand.

Col. 4. The beginning position of the predicted feature (numbered on the input strand).

Col. 5. The ending position of the predicted feature (numbered on the input strand).

Col. 6. The length of the predicted feature (in base pairs).

Col. 7. The "absolute reading frame" of the predicted exon: a codon ending at position x in the sequence has reading frame x mod 3.

Col. 8. The "net phase" of the predicted exon (exon length modulo three).[4]

Col. 9. The score ($\times 10$) of the translation initiation or acceptor signal at the $5'$ end of the predicted exon (Section 4.2).

Col. 10. The score ($\times 10$) of the donor or termination signal at the $3'$ end of the exon (Section 4.3).

Col. 11. The coding score ($\times 10$) of the coding portion of the exon (Section 3.6).

Col. 12. The exon probability, $P\{\epsilon|S\}$ (Section 2.8).

Col. 13. The "exon score" (Section 2.10).

---

[2] In the example, a single complete gene comprising 26 coding exons is predicted.

[3] Type-$\mathcal{D}$ states (intron, intergenic, etc.) are not indicated, since their locations are always implied by the exon/signal locations.

[4] This number may be useful in evaluating potential alternative splices derived by omission of individual exons since any exon of net phase zero can be omitted from a parse without disrupting the reading frames of flanking exons.

Finally, the last column summarizes the accuracy of the predicted exon (see Section 5.2.2 for precise definitions of these terms): exactly correct ("EXAC"); partially correct ("PART"); or wrong ("WRNG"). (Of course, this last column is not normally present in the program output — it is provided here simply for illustrative purposes.) Note that, since the first predicted feature is an inital exon (and there is no prior predicted promoter), the initial state of the parse is 5′ UTR; since the last predicted feature is a poly-adenylation signal, the terminal state of the parse is intergenic. In this example, the annotated NCAM gene contains 28 coding exons, of which 23 were predicted exactly (see Fig. 11), three were predicted partially (annotated exons at [6,192:6,314], [6,411:6,581] and [11,551:11,666]) and two were missed completely (annotated exons at [4,127:4,141] and [13,990:14001]). No promoter or poly-adenylation signals are listed in the GenBank annotation. The predicted poly-adenylation signal at [15,677:15,682] (approximately one kb 3′ of the stop codon) matches the consensus AATAAA and might well be correct.

Though the overall level of accuracy in this example is somewhat higher than average for GENSCAN (see Section 5.3), it is by no means atypical. This example also serves to illustrate some of the strengths and weaknesses of the program. First, it is notable that the probabilities of many of the exons are very high. In particular, 21 of 26 predicted exons have probability > 0.90: of these, all are exactly correct. Of the exons with probability less than 0.90, two are exactly correct, two are partially correct (predicted exons 4 and 18), and one is wrong (predicted exon 17). Taken together, these results suggest that the exon probability may be a useful guide to the reliability of the prediction: this issue is addressed systematically in Section 5.4.5. The types of mistakes made by the program are also of interest. Of the partially correct exons, the donor site of predicted exon 18 is correct, but its acceptor site is 49 bp 5′ of the actual site; both splice sites of predicted exon four are correct but they belong to different annotated exons (see Fig. 12), i.e. the predicted exon erroneously includes a small intron. The predicted exon which is wrong (i.e. not overlapped by any annotated exon) is unusual in several respects: it not only has the lowest probability and exon score of any predicted exon, but it also has unusually weak donor and acceptor splice signal scores (both weaker than any score for a true splice site in this gene). Thus, the

splice signal and exon scores may also provide useful information about the reliability of the prediction. Finally, the most distinctive property of the two annotated exons which were missed is their extremely small size (15 and 12 bp, respectively), raising the issue of the accuracy of prediction as a function of exon length (Section 5.4.1).

5.1.3 Email and web servers

The GENSCAN program has been made available to the scientific community in two forms. First, an electronic mail server was set up which may be accessed by sending a DNA sequence in FastA format to **genscan@gnomic.stanford.edu**. The standard command line version of the program is then run locally and the results (text output) are emailed back to the sender, usually in a few minutes or less. Inclusion of the word "POSTSCRIPT" at the beginning of the mail message causes the program to return the graphical (PostScript) output as well. Second, a web interface for the program was developed [**http://gnomic.stanford.edu/GENSCANW.html**]. This form of the program may be accessed using a web browser such as Netscape Navigator: the genomic sequence to be analyzed is simply "pasted" into the appropriate box on the web page. The sequence is then processed locally by the standard command-line version of GENSCAN and a web page is created which displays the text output and provides links to PostScript and gif images of the predicted exon locations in the sequence. (The gif image may be viewed directly through most web browsers; the PostScript file can downloaded for viewing or printing on the user's computer.) The web server was designed primarily for users who have only one or a few sequences to process or who just want to try out the program. The email server is more appropriate for users who have a large number of sequences to process or who wish to systematically check the accuracy of the program on a large test set. Both versions accept sequences up to 200 kb in length. Computer memory is the only factor which limits the length of sequence which can be processed. If necessary, a program version could be written which can process arbitrarily long sequences by writing intermediate values of the recursion variables and certain other quantities to a file instead of storing them in computer memory. Such a program would, of course, run much more slowly (probably by a factor of ten or more).

5.2 Measuring predictive accuracy

A variety of quantitative measures have been proposed to characterize the accuracy of gene prediction methods (reviewed in Burset & Guigó, 1996). Fundamentally, accuracy is related to the degree of concordance between the predicted and actual (annotated) exon locations. At the nucleotide level, a prediction for a sequence of length L may be represented by the L-vector $\vec{p}$ with $p_i = 1$ if position i is in a predicted coding region, 0 otherwise, for $1 \leq i \leq L$. The actual set of coding exons in the sequence is represented by an analogous L-vector $\vec{a}$ obtained from the sequence annotation (GenBank "CDS" feature). The following quantities may then be calculated from these vectors:

Predicted positives (predicted coding bp): $PP = \sum_{i=1}^{L} p_i$

Predicted negatives (predicted non-coding bp): $PN = L - PP$

Actual positives (annotated coding bp): $AP = \sum_{i=1}^{L} a_i$

Actual negatives (annotated non-coding bp): $AN = L - AP$

True positives[5]: $TP = \vec{p} \cdot \vec{a} = \sum_{i=1}^{L} p_i a_i$

True negatives[6]: $TN = (\vec{1} - \vec{p}) \cdot (\vec{1} - \vec{a}) = \sum_{i=1}^{L} (1 - p_i)(1 - a_i)$

5.2.1 Nucleotide-level accuracy

Two commonly used measures of accuracy at the nucleotide level are sensitivity, $Sn = \frac{TP}{AP}$, the proportion of actual coding nucleotides which were predicted to be coding (undefined if $AP = 0$); and specificity, $Sp = \frac{TP}{PP}$, the proportion of predicted coding nucleotides which were correct (undefined if $PP = 0$). Both sensitivity and specificity range from 0 to 1, with perfect prediction occurring if and only if both Sn and Sp are unity. However, neither measure by itself is a sufficient description of

[5]Here, $\vec{x} \cdot \vec{y}$ is the standard dot product between two vectors.

[6]$\vec{1}$ represents the L-vector all of whose components are unity.

accuracy. For example, $Sn \equiv 1$ (is identically equal to unity) for the trivial program which predicts all nucleotides of any input sequence to be coding (but $Sp \equiv 0$ in this case), and $Sp \equiv 1$ (but $Sn \equiv 0$) for a program which always predicts all nucleotides as non-coding.

A single measure which captures aspects of both sensitivity and specificity is the correlation coefficient, CC, defined as the Pearson (product-moment) correlation between the vectors $\vec{p}$ and $\vec{a}$, which may be calculated as: $CC = \frac{(TP)(TN)-(FP)(FN)}{\sqrt{(PP)(PN)(AP)(AN)}}$. When defined, CC may be the best single measure of accuracy at the nucleotide level. However, CC is not defined when any of the quantities PP, PN, AP, AN are zero, e.g., if there is either no actual gene or no predicted gene in a sequence, making it unsuitable for use in all cases. For this reason, Burset & Guigó (1996) introduced the "average conditional probability" (ACP) measure, defined as the average of Sn, Sp, and the quantities $Sn' = \frac{TN}{AN}$, $Sp' = \frac{TN}{PN}$ corresponding to the sensitivity and specifity for non-coding positions. The average is taken over all conditional probabilities which are defined (always at least one). For example, if all such terms are defined, $ACP = \frac{1}{4}[\frac{TP}{PP} + \frac{TP}{AP} + \frac{TN}{PN} + \frac{TN}{AN}]$. Since ACP is an average of (conditional) probabilities, it ranges between 0 and 1. The quantity $AC = 2 \times ACP - 1$, termed "approximate correlation", ranges from -1 to 1 and can be compared to the correlation coefficient, CC. In practice, AC is usually close to CC whenever CC is defined (data not shown). Further discussion of the advantages and disadvantages of each of these measures may be found in Burset & Guigó (1996).

5.2.2 Exon-level accuracy

The nucleotide level accuracy measures Sn, Sp, CC and AC give an overall sense of how closely the predicted and actual coding regions in a sequence align, but do not accurately reflect the identification of precise exon boundaries (splice sites). For this purpose, a related set of exon-level accuracy measures are defined. Predicted exons can be divided into four categories: *exact* (identical to an annotated exon); *partial* (one or both endpoints correct, but not identical to an actual exon, e.g., exons 4 and 18 in Fig. 11); *overlapping* (neither endpoint correct, but overlaps an actual

exon); or *wrong* (not overlapped by any annotated exon). Similarly, annotated exons can be divided into those which are *exactly* predicted, *partially* predicted, *overlapped*, or *missed* (not overlapped by any predicted exon). For a test sequence, predicted exons, PE, are compared to actual (annotated) exons, AE: true exons, TE, is the number of predicted exons which *exactly* match an actual exon. Accuracy is then measured by exon-level sensitivity, $ESn = \frac{TE}{AE}$ and exon-level specificity, $ESp = \frac{TE}{PE}$. The average of these two quantities, $EA = \frac{ESn+ESp}{2}$ ("average exon-level accuracy"), is typically used as an overall measure of accuracy at the exon level in lieu of a correlation measure. Other useful measures (Burset & Guigó, 1996) include: ME, the proportion of annotated exons in a sequence which were missed; and WE, the proportion of predicted exons which are wrong. For these last two measures, of course, lower values indicate improved prediction.

5.2.3 Accuracy for a set of sequences

The above measures can be applied to any sequence containing a single gene (or multiple genes on the same DNA strand). To summarize nucleotide level accuracy for a set of sequences, there are two conventions: either (i) calculate each measure separately for each sequence and average over all sequences for which the measure is defined; or (ii) sum the basic quantities PP, TP, PE, etc. as if all the sequences were concatenated, and calculate accuracy measures from these overall numbers. In the examples below, accuracy is calculated by the first convention unless otherwise indicated.[7] At the exon level, the convention of Burset & Guigó (1996) is to calculate the quantities ESn, ESp, ME, etc. for each sequence and average these measures over all sequences for which they are defined. Snyder & Stormo (1995) use somewhat different measures, namely: EC, the proportion of actual exons in the *set* of sequences which were predicted exactly; and EO, the proportion of actual exons in the set which were at least overlapped by a predicted exon. Finally, given a set of sequences each of which contains a single complete gene, one may define "gene level accuracy", GA, as the proportion of actual genes which were predicted exactly, i.e. all exons

[7] In practice, both averaging conventions tend to give very similar results — see next section.

exactly correct and no false exons predicted in the transcription unit (in practice, the GenBank file). This measure, the most stringent of any proposed to date, reflects a combination of both sensitivity and specificity since it penalizes for both missed and wrong (or partially correct) exons.

5.3 Accuracy of GENSCAN vs other programs

The performance of GENSCAN has been measured on several different sets of human and vertebrate genomic sequences. First, it was tested on the set of 570 vertebrate genes constructed by Burset & Guigó (1996) as a standard for comparison of gene prediction methods. This set, available by anonymous ftp[8], is by far the largest set of genes ever constructed for this purpose and has been carefully screened to remove pseudo-genes, alternatively spliced genes and sequences with ambiguous, partial or inconsistent annotation. Another advantage of this set is that results have been compiled for virtually all available gene finding programs (a highly complex undertaking, given the varied system requirements and output formats of these programs). A disadvantage is that the set includes only multi-exon (intron-containing) genes and no single-exon genes. In addition, since this set contains the vast majority of available vertebrate genomic sequences with reliable annotation, it was not possible to construct a truly independent training set of sufficient size to give reliable parameter values. For this reason, the smaller GeneParser test sets of 28 and 34 human genes (Snyder & Stormo, 1995) were set aside for testing at the beginning and all sequences $> 25\%$ identical to any of these genes were removed from the GENSCAN learning set, as described in Section 3.1. Performance of the program turned out to be quite robust with respect to different sequence sets, giving similar results for the Burset/Guigó and both GeneParser test sets (see below).

[8] See [http://www.imim.es/GeneIdentification/Evaluation/Index.html].

5.3.1 Burset/Guigó test set

Table 10 displays the standard measures of predictive accuracy at the nucleotide and exon levels for GENSCAN and other programs as tested on the Burset/Guigó sequence set. Comparison of the accuracy data in this table shows that GENSCAN is significantly more accurate at both the nucleotide and the exon level by *all* measures of accuracy than *all* existing programs which do not use protein sequence homology information (those in the upper portion of Table 10). This is probably the most important result obtained in this thesis. At the nucleotide level, substantial improvements are seen in terms of Sensitivity ($Sn = 0.93$ versus 0.77 for the next best program, FGENEH), Approximate Correlation ($AC = 0.91$ versus 0.78 for FGENEH), and Correlation Coefficient ($CC = 0.92$ versus 0.80 for FGENEH). At the exon level, dramatic improvements are seen across the board, both in terms of Sensitivity ($Sn = 0.78$ versus 0.61 for FGENEH) and Specificity ($Sp = 0.81$ versus 0.64 for FGENEH), as well as Missed Exons (0.09 versus 0.15 for FGENEH) and Wrong Exons (0.05 versus 0.11 for GRAIL). At the gene level, 243 out of 570 genes were predicted exactly by GENSCAN (so that $GA = \frac{243}{570} = 0.43$), demonstrating that the program is indeed capable of predicting complete multi-exon gene structures with a reasonable degree of success. The most complex multi-exon gene predicted exactly by GENSCAN was the human gastric (H+, K+)-ATPase gene (GenBank accession # J05451), containing 22 coding exons.

GENSCAN was also found to be more accurate by almost all measures than the two programs, GeneID+ (Guigó & Knudsen, unpublished) and GeneParser3 (Snyder & Stormo, 1995), which make use of protein sequence homology information (Table 10). In particular, all nucleotide-level measures were higher for GENSCAN. At the exon level, ESn and ESp values were substantially higher for GENSCAN and wrong exons (WE) substantially lower; only in the category of missed exons (ME) did GeneID+ do better (0.07 versus 0.09 for GENSCAN). Since GENSCAN is intended for use as a first-pass screening technique for newly sequenced genomic regions, peptide sequences predicted by the program can then be searched against the protein sequence databases using a program such as BLASTP (Altschul *et al.*, 1990) and these results can provide independent confirmation or refinement of the prediction.

Table 10. Comparison of gene prediction programs: Burset/Guigó test set

		Accuracy per bp				Accuracy per exon				
Program	No. seq.	*Sn*	*Sp*	*AC*	*CC*	*ESn*	*ESp*	*EA*	*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	0.76	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	0.66	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	0.71	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

Legend. For each sequence in the Burset/Guigó test set, the forward-strand exons in the default GENSCAN output (the optimal parse) were compared to the annotated exons (GenBank "CDS" key). The standard measures of predictive accuracy per nucleotide (bp) and per exon (see text) were calculated for each sequence and averaged over all sequences for which they were defined. Results for the other programs are from Table 1 of Burset & Guigó (1996), except Genie, for which results are from Kulp *et al.* (1996) — *CC* was not calculated for this program. Under the heading "No. seq.", the number of sequences (out of 570) effectively analyzed by each program is given (some programs failed to run on certain sequences), followed by the number of sequences for which no gene was predicted, in parentheses. (Some programs, e.g., Genie, always predict a gene in any input sequence.) Performance of the two programs which make use of amino acid similarity searches, GeneID+ and GeneParser3, are shown separately at the bottom of the table; these programs were tested only on sequences less than 8 kb in length.

Table 11. GENSCAN accuracy versus C+G content: Burset/Guigó test set

C+G%		Accuracy per bp				Accuracy per exon				
range	No. seq.	Sn	Sp	AC	CC	ESn	ESp	EA	ME	WE
> 57	104 (1)	0.95	0.90	0.89	0.90	0.76	0.77	0.77	0.08	0.07
51 – 57	139 (2)	0.93	0.94	0.91	0.92	0.75	0.76	0.76	0.07	0.05
43 – 51	164 (1)	0.94	0.91	0.91	0.91	0.80	0.82	0.81	0.07	0.05
< 43	163 (4)	0.92	0.94	0.91	0.93	0.79	0.86	0.84	0.12	0.05
< 43*	163 (6)	0.87	0.96	0.88	0.92	0.75	0.83	0.84	0.17	0.03

Legend. The Burset/Guigó set was partitioned according to the C+G% content (first column) of the GenBank sequence. The first four rows show results of the default GENSCAN program, which uses overall coding/non-coding hexamer matrices (see Chapter 3) for sequences in groups II, III and IV, but uses group I-specific matrices for sequences of $< 43\%$ C+G. The last row, designated *, shows data for sequences of $< 43\%$ C+G scored with the overall hexamer matrices. The number of sequences in each subgroup is given under the heading "No. seq.", followed by the number of sequences for which no gene was predicted, in parentheses.

Thus, the overall accuracy of an integrated gene finding approach involving GENSCAN might well be significantly higher than the values listed in Table 10 would indicate. The genomic sequence may also be screened against the EST database (Boguski, 1995) using BLASTN (Altschul *et al.*, 1990) or TBLASTX (Gish & States, 1993): an example of the use of this information is given in Section 5.5.

5.3.2 Accuracy versus C+G% content

It was also of interest to determine whether or not the steps taken to account for the structural properties of genes in different compositional regions (Chapter 3) were indeed successful in achieving the stated goal (Section 1.1) of predictive performance which is independent of sequence composition. Table 11 shows the accuracy of GENSCAN for different C+G% compositional subsets of the Burset/Guigó test set. The results show that overall performance, as measured by AC or CC, is indeed almost independent of composition, e.g., CC values of 0.93, 0.91, 0.92 and 0.90 were obtained for seqeunces of < 43, 43-51, 51-57, and $> 57\%$ C+G, respectively, and AC values were even more homogeneous. Comparison of the last two rows of the table shows

Table 12. GENSCAN accuracy: GeneParser test sets

Test		Accuracy per bp				Accuracy per exon				
set	No. seq.	*Sn*	*Sp*	*AC*	*CC*	*ESn*	*ESp*	*EA*	*ME*	*WE*
I	28 (0)	0.98	0.90	0.93	0.93	0.77	0.78	0.77	0.04	0.04
II	34 (1)	0.90	0.93	0.89	0.91	0.69	0.78	0.75	0.13	0.03

Legend. Data is shown for GeneParser test sets I (28 human genes) and II (34 human genes), which are described in Snyder & Stormo (1995). Accuracy measures were calculated as described in the legend to Table 11 (see also text), using the Burset/Guigó measures.

that the group I-specific scores (Section 3.6) do indeed provide better accuracy in A+T rich sequences by almost all measures. However, there were some subtle differences in the types of errors made by the program in A+T rich sequences versus others. Specifically, the number of genes missed (4) was higher than for the other groups as was the Missed Exons statistic (0.12 versus 0.08 or less for the other three groups). However, this is compensated for by a fairly large increase in exon-level specificity, resulting in overall exon accuracy levels which are actually highest in this subset. Nevertheless, these differences are generally rather slight and are less substantial than the differences observed for most other programs for distinct C+G% compositional subsets of this set (Burset & Guigó, 1996) or for the GeneParser test sets (see below).

5.3.3 GeneParser test sets

It was of particular interest to determine whether or not the relatively high level of accuracy observed for GENSCAN on the Burset/Guigó set would also hold for a truly independent test set not overlapped by any genes of the learning set. GENSCAN accuracy for the GeneParser test sets, as calculated using the conventions of Burset & Guigó, is shown in Table 12. At the nucleotide level, correlation coefficient (CC) values of 0.93 and 0.91 for the GeneParser test sets are very close to the value of 0.92 observed in the Burset/Guigó set, and similarly for the approximate correlation, AC. For GeneParser test set II, exon-level sensitivity and specificity values were a bit lower than for the Burset/Guigó set and ME was higher, but the proportion of wrong exons

was lower (0.03 versus 0.05). This difference may be explained in part by the fact that a higher proportion ($1/34 \approx 3\%$) of the genes in this set was completely missed by GENSCAN, as compared to 8 out of 570 (1.4%) of genes in the Burset/Guigó set. For GeneParser test set I, by contrast, no gene was missed by GENSCAN. In this set, exon-level ESn and ESp values are close to the Burset/Guigó values and both ME and WE are lower, with a particularly low value observed for missed exons (0.04 versus 0.09). Thus, the overall level of accuracy for GENSCAN on the GeneParser sets was quite comparable to that observed in the much larger Burset/Guigó set.

In their paper, Snyder & Stormo (1995) used somewhat different accuracy measures (see previous section) to compare the performance of their program, GeneParser, with that of GeneID (Guigó *et al.*, 1992) and "GRAIL3" (Xu *et al.*, 1994b), i.e. the X-windows version of GRAIL II with the "gene assembly" option. GENSCAN accuracy for these sets, as measured using their conventions, is shown in Table 13 alongside the corresponding values for the other programs. Several features of this data are notable. First, both measurement conventions give nearly identical values for the nucleotide-level accuracy measures Sn, Sp and CC (comparing Tables 12 and 13), and the EC statistic is similar to the Burset/Guigó ESn statistic. Thus it probably does not make a great deal of difference which averaging convention is used to summarize the accuracy of a gene prediction program for a set of sequences. Secondly, it is clear that the performance of GRAIL, and to a lesser extent GeneParser, is sensitive to C+G% content, with lower levels of accuracy observed in A+T rich sequences (e.g., comparing CC or EC values), while GeneID appears less sensitive to C+G content, but has overall accuracy which is significantly lower than for the other programs. GENSCAN's performance, as was observed for the Burset/Guigó set, is quite stable with respect to C+G content and is consistently better than the other programs, sometimes by a fairly wide margin.

5.4 Accuracy of GENSCAN: a closer look

In this section, the accuracy of GENSCAN is examined from various points of view in order to gain a better understanding of the factors contributing to its performance and

Table 13. Comparison of gene prediction programs: GeneParser test sets

	Program							
	GeneID		GRAIL3		GP2		GENSCAN	
DATASET	I	II	I	II	I	II	I	II
All sequences								
Sn (bp level)	0.69	0.50	0.83	0.68	0.87	0.82	0.98	0.95
Sp (bp level)	0.77	0.75	0.87	0.91	0.76	0.86	0.90	0.94
CC (bp level)	0.69	0.55	0.83	0.75	0.78	0.80	0.93	0.93
EC (exons correct)	0.42	0.33	0.52	0.31	0.47	0.46	0.79	0.76
EO (exons overlapped)	0.73	0.64	0.81	0.58	0.87	0.76	0.96	0.91
High C+G								
Sn (bp level)	0.72	0.85	0.87	0.80	0.90	0.65	1.00	0.98
Sp (bp level)	0.73	0.73	0.95	0.88	0.93	0.87	0.91	0.98
CC (bp level)	0.65	0.73	0.88	0.80	0.89	0.71	0.94	0.98
EC (exons correct)	0.38	0.43	0.67	0.50	0.64	0.57	0.76	0.64
EO (exons overlapped)	0.80	0.86	0.89	0.79	0.96	0.79	1.00	0.93
Medium C+G								
Sn (bp level)	0.65	0.47	0.86	0.68	0.86	0.84	0.97	0.95
Sp (bp level)	0.77	0.76	0.84	0.91	0.70	0.87	0.90	0.95
CC (bp level)	0.67	0.52	0.83	0.75	0.75	0.82	0.93	0.94
EC (exons correct)	0.37	0.29	0.51	0.32	0.41	0.46	0.79	0.79
EO (exons overlapped)	0.67	0.62	0.83	0.38	0.84	0.79	0.96	0.93
Low C+G								
Sn (bp level)	0.82	0.56	0.51	0.45	0.79	0.71	0.93	0.80
Sp (bp level)	0.85	0.71	0.87	0.89	0.75	0.67	0.94	0.84
CC (bp level)	0.81	0.62	0.62	0.62	0.72	0.67	0.92	0.81
EC (exons correct)	0.80	0.47	0.25	0.16	0.40	0.37	0.85	0.68
EO (exons overlapped)	0.85	0.63	0.55	0.42	0.85	0.58	0.85	0.74

Legend. Performance data is shown for GeneID, GRAIL3 (GRAIL II + "assembly"), GeneParser2 (GP2), and GENSCAN for GeneParser test sets I and II. Sequence sets and performance data for programs other than GENSCAN are from Snyder & Stormo (1995). Nucleotide- and exon-level accuracy statistics were calculated using the conventions of Snyder & Stormo (see text). Each test set was divided into three subsets according to the C+G content of the GenBank sequence: low C+G (< 45%); medium C+G (45 - 60%); and high C+G (> 60%).

to assess its strengths and weaknesses. The approach taken is to examine the accuracy as a function of several independent variables, including exon size, gene complexity, exon type and organism of origin. Finally, the last subsection addresses predictive accuracy as a function of the exon forward-backward probability (Section 2.8), in a sense asking how "self-critical" the program is, i.e. how well it can distinguish which parts of its predictions are reliable and which are not.

5.4.1 Accuracy as a function of exon length

The NCAM example (Figs. 11 and 12) raised the issue of whether or not GENSCAN is capable of finding very small exons. To measure this effect systematically, annotated and predicted exons were grouped into ten different length ranges, and accuracy statistics were calculated separately for each group (Table 14). This data shows that the proportion of exons missed is indeed much higher for very small exons, but decreases steadily with increasing exon length reaching very low levels (one or two percent) for exons of average size or above (say $>$ 150 bp). The most obvious explanation for this phenomenon is that the amount of information derived from the coding/non-coding portion of the exon score increases (linearly) with exon length (Section 3.6). Interestingly, there is a considerable amount of biochemical evidence suggesting that such extremely small exons are inefficiently spliced and/or require the presence of special splicing activating sequences in the flanking introns (e.g., Dominski & Kole, 1991, Black, 1991). Building models of such signals into the overall gene model architecture (after the nature and function of these signals is more clearly worked out) might allow more accurate identification of short exons.

At the other extreme, although very large exons (say $>$ 300 bp) are almost never missed completely, they are somewhat less likely to be predicted exactly and more likely to be predicted partially than medium-sized exons. The explanation for this phenomenon is not clear, but might relate to the increased number of potential splice sites in long exons or to other factors. Apropos, there is some biochemical evidence that long exons ($>$ 300 or 500 bp) are less efficiently recognized by the splicing machinery and may be skipped *in vitro* (Robberson *et al.*, 1990) and *in vivo* (Sterner

Table 14. GENSCAN accuracy versus exon length: Burset/Guigó set

Length range (bp)	Annotated exons				Predicted exons			
	#	%Exac	%Part	%Miss	#	%Exac	%Part	%Wrng
≤ 24	89	38	8	52	44	77	11	11
25 – 49	163	58	15	25	124	76	6	18
50 – 74	248	70	12	16	204	85	9	6
75 – 99	382	85	8	6	389	84	6	10
100 – 124	351	84	9	7	366	81	8	11
125 – 149	425	88	8	4	460	81	10	7
150 – 174	261	88	9	2	283	81	11	7
175 – 199	167	91	7	2	188	81	12	7
200 – 299	353	90	8	1	390	82	8	8
≥ 300	211	66	19	1	204	69	20	10
Total	2650	81	10	8	2678	81	10	9

Legend. For each range of lengths (column 1), the number of annotated and predicted exons in the genes of the Burset/Guigó test set are given in columns 2 and 6, respectively. Columns 3, 4 and 5 give the percentage of annotated exons predicted exactly, predicted partially, or missed, respectively. Columns 7, 8 and 9 give the percentage of predicted exons which were exactly correct, partially correct, or wrong, respectively. Overall totals for the Burset/Guigó set are given in the bottom row. The percentage of exons overlapped is not given because: 1) this number is negligible in most cases; and 2) in any case, it can be easily calculated from the data in the other columns.

et al., 1996); the lengths of flanking introns may also be important (Sterner *et al.*, 1996). Finally, it is notable that *predicted* small exons (say < 50 bp) are almost as likely as medium or large predicted exons to be exactly correct. Thus, although the program does miss quite a few small exons, when it does predict a small exon it is likely to be correct.

5.4.2 Accuracy as a function of gene complexity

One of the problems with the Burset/Guigó test set, as for the GENSCAN learning set (Section 3.1), is that it is biased toward relatively short sequences containing genes of relatively low complexity.[9] Specifically, the average number of exons per gene in the Burset/Guigó set is only 4.6, probably significantly less than the average for human genes, and the GeneParser sets have a similar bias toward short genes with relatively few exons. This raises the obvious issue of whether the results obtained for GENSCAN and other programs on this set will carry over to more typical human genomic contigs containing genes of greater average complexity. To address this issue, the genes of the Burset/Guigó set were divided into four groups according to the number of coding exons and results tabulated separately for each such group (Table 15). Contrary to what one might expect, the accuracy of GENSCAN was actually found to *increase* with increasing gene complexity, reaching its highest levels by all measures for genes with ten or more exons! This surprising result raises the possibility that predictive accuracy might actually be higher for typical genomic sequences than for the Burset/Guigó set. The most likely explanation for this phenomenon has to do with the the differing accuracy for different exon types (see below).

5.4.3 Accuracy as a function of exon type

The accuracy of GENSCAN for different types of exon is summarized in Table 16. The results show quite clearly that internal exons are the most accurately predicted, with

[9] Here, the term "gene complexity" refers simply to the number of exons/introns the gene contains.

Table 15. GENSCAN accuracy versus gene complexity: Burset/Guigó set

# of		Accuracy per bp				Accuracy per exon				
exons	No. seq.	*Sn*	*Sp*	*AC*	*CC*	*ESn*	*ESp*	*EA*	*ME*	*WE*
2 – 3	284 (7)	0.92	0.93	0.89	0.92	0.73	0.77	0.76	0.10	0.05
4 – 5	147 (1)	0.94	0.91	0.90	0.91	0.81	0.83	0.83	0.09	0.07
6 – 9	103 (0)	0.96	0.93	0.93	0.93	0.83	0.85	0.84	0.07	0.05
≥ 10	36 (0)	0.96	0.93	0.94	0.93	0.87	0.87	0.87	0.05	0.05

Legend. The organization of the table is similar to that of Table 11, except that genes of the Burset/Guigó set were divided according to the number of coding exons (column 1) rather than C+G% content. Accuracy measures are as described in text. The number of sequences in each subgroup is given under the heading "No. seq.", followed by the number of sequences for which no gene was predicted, in parentheses.

Table 16. GENSCAN accuracy versus exon type: Burset/Guigó set

Exon	Annotated exons				Predicted exons			
type	#	%Exac	%Part	%Miss	#	%Exac	%Part	%Wrng
Initial	570	65	25	9	457	81	9	10
Internal	1,510	90	5	4	1,707	80	11	8
Terminal	570	76	8	15	509	84	6	8
Total	2,650	81	10	8	2,678	81	10	9

Legend. The organization of the table is the same as for Table 14, except that annotated and predicted exons were grouped by exon type rather than length. Again, the proportion of overlapped exons is not given and totals are listed in the bottom row. A total of five single-exon genes were predicted in the Burset/Guigó sequences (not shown as a separate row, but included in the totals).

an impressive 90% of all annotated internal exons identified exactly. This finding suggests that the splice signal models contribute strongly to GENSCAN's performance, and offers an explanation for why the program performs better on more complex genes. Obviously, the more coding exons a gene has, the higher the proportion of internal exons relative to initial and terminal exons, e.g., a gene with three coding exons has only 33% internal exons, while this proportion increases to 80% or more for genes with ten or more coding exons.

Terminal exons were the most likely to be missed (15% versus 9% or less for the other types), which may be explained by the much greater average information content of the donor splice signal at the 3′ terminus of initial and internal exons versus the stop signal at the 3′ end of terminal exons. Two other somewhat puzzling features of this data turned out to be closely related. First, initial exons were the least likely to be predicted exactly (65% versus more than 75% for the other two types), but were less likely to be missed than terminal exons, with a much higher proportion of partially predicted exons (25%) than for the other exon types. Second, GENSCAN seems to have a tendency to overpredict internal exons (1,707 predicted versus 1,510 actual), but to underpredict initial exons (457 versus 570).

Further examination of the predictions for the Burset/Guigó set showed that one of the most common mistakes made by GENSCAN was to predict initial exons as longer internal exons ending at the correct donor site, but beginning at an acceptor site 5′ to the translation initiation site, and that many of the partially predicted initial exons are of this type. In some cases these internal exons were actually correct spliceosomal exons, i.e. the predicted acceptor site was correct for an intron upstream (5′) of the translation start site. Such a prediction is incorrect in the sense that it will lead to a predicted protein which has some extra amino acid residues at its amino terminus, but might still be perfectly acceptable for certain purposes, e.g., design of PCR primers to amplify a cDNA from a cDNA library. It was not possible to quantify precisely how many such predicted internal exons were correct in this sense because the GenBank annotation often does not indicate exons and introns outside of the coding region (CDS). Finally, similar levels of accuracy were observed when *predicted* exons were grouped by exon type (right half of Table 16), in contrast to the

Table 17. GENSCAN accuracy for different organisms

Organism		Accuracy per bp				Accuracy per exon				
or group	No. seq.	*Sn*	*Sp*	*AC*	*CC*	*ESn*	*ESp*	*EA*	*ME*	*WE*
Primates	237 (1)	0.96	0.94	0.93	0.94	0.81	0.82	0.82	0.07	0.05
Rodents	191 (4)	0.90	0.93	0.89	0.91	0.75	0.80	0.78	0.11	0.05
Non-mam.	72 (2)	0.93	0.93	0.90	0.93	0.81	0.85	0.84	0.11	0.06
Drosophila	202 (1)	0.96	0.92	0.89	0.90	0.68	0.68	0.68	0.11	0.10
Maize	41 (0)	0.94	0.93	0.90	0.90	0.67	0.71	0.69	0.09	0.08

Legend. The first three rows show results of GENSCAN for different subsets of the Burset/Guigó test set, divided by the organism of origin. Classification by organism was based on the GenBank "ORGANISM" key: the primate set comprised mostly human sequences; the rodent set, mostly mouse and rat; and the non-mammalian set, a diverse group of vertebrates comprising 22 fish, 17 amphibian, 5 reptilian and 28 avian sequences. The last two rows show accuracy for sets of *Drosophila melanogaster* and *Zea mays* sequences, respectively (see text). The number of sequences in each subgroup is given under the heading "No. seq.", followed by the number of sequences for which no gene was predicted, in parentheses. Accuracy measures are as described in text.

case for annotated exons.

5.4.4 Accuracy for different organisms

An issue of obvious importance is the phylogenetic generality of the program. To address this question, sequences from the Burset/Guigó set were divided into primates, rodents, and non-mammalian vertebrates[10] (Table 17). Overall, accuracy was fairly similar for these groups, but subtle differences were apparent. In particular, the program seems to perform slightly less well on rodent sequences than for primates, but this difference cannot be explained simply on the basis of the greater phylogenetic distance of the rodent sequences to the human sequences from which GENSCAN parameters were derived, since accuracy was higher for the (more distant) non-mammalian vertebrate sequences by all measures (Table 17). The explanation for this effect is not clear, but might relate to biases in the set of available rodent sequences or other

[10]There were also some non-primate, non-rodent mammalian sequences, but not enough to make any other coherent taxonomic groupings.

non-biological effects. Other programs exhibited variable dependence on vertebrate group (Burset & Guigó, 1996).

The program was also tested on two sets of non-vertebrate sequences: a set of 202 complete *Drosophila* genes, based on a set constructed by D. Kulp and M. G. Reese (Appendix C), and a set of 41 nonredundant complete maize genes constructed by V. Brendel (Appendix D). Surprisingly, nucleotide-level accuracy was found to be almost as high for *Drosophila* and maize as for human, but exon-level accuracy was somewhat lower. Even so, the fact that the accuracy was not dramatically different for these two organisms than for vertebrates suggests that the overall model architecture introduced here may be sufficiently general so as to be applicable to most or all higher eukaryotes. These results can also be interpreted as evidence that the signals recognized by the basic splicing machinery are fundamentally similar across vertebrates, invertebrates and plants. However, the program is not completely general, e.g., users have reported rather poor performance for *C. elegans* genomic sequences. It is possible that *trans*-splicing or some other nematode-specific feature may confuse the gene model in this case. A natural follow-up project would be to compare gene structural and compositional properties among vertebrates, monocot and dicot plants, nematodes, arthropods, etc. and perhaps to develop separate parameter files or program versions appropriate to different classes or phyla.[11]

5.4.5 Accuracy as a function of exon probabilities

Finally, it was of interest from at least two points of view to determine how closely the exon (forward-backward) probabilities (Section 2.8) reflect the accuracy of predicted exons. First, as a practical matter, one would like to know how useful the exon probability is as a guide to the reliability of the prediction. Secondly, it is of interest to determine how "self-critical" the program is, i.e. the extent to which the model structure can distinguish which parts of a sequence are well described and which are uncertain. For this purpose, the predicted exons in the optimal parse of each Burset/Guigó sequence were grouped by exon probability, $P\{\epsilon|S\}$ and then

[11] Preliminary efforts in this direction have yielded promising results (data not shown).

Table 18. Accuracy versus exon probability: Burset/Guigó test set

Probability range	Predicted exons	Accuracy class			
		Exact	Partial	Overlap	Wrong
< 0.50	248	29.8%	27.8%	4.0%	38.3%
$0.50 - 0.75$	362	54.1%	26.2%	2.2%	17.4%
$0.75 - 0.90$	337	74.8%	16.0%	1.2%	8.0%
$0.90 - 0.95$	263	87.8%	6.1%	0.4%	5.7%
$0.95 - 0.99$	551	92.4%	3.4%	0.2%	4.0%
> 0.99	917	97.7%	0.9%	0.0%	1.4%
Total	2,678	80.6%	9.7%	0.9%	8.8%

Legend. GENSCAN was run on the Burset/Guigó test set of 570 vertebrate genes and predicted exons were grouped according to their probability, $P\{\epsilon|S\}$ (first column). For each such group, the proportion of exons which were exactly correct, partially correct, overlapped a true exon, or were wrong, are given in columns 3 to 6, respectively. The total number of predicted exons in each group is shown in column 2.

compared to the sequence annotation (Table 18). The results indicate quite clearly that the exon probability is a very useful guide to the likelihood that a predicted exon is correct, with the proportion of exact predictions increasing monotonically as a function of exon probability. In particular, fully 97.7% of the highest probability predicted exons ($P\{\epsilon|S\} > 0.99$) were exactly correct, and this set is by no means negligible, comprising more than a third of all predicted exons. As a consequence, any predicted gene with six or more exons is likely to contain at least two exons with probability > 0.99 (both of which are almost certainly correct), from which PCR primers could be designed to amplify the cDNA with an extremely high degree of confidence. Furthermore, slightly more than half of predicted exons have probability > 0.95, and of these more than 95% are exactly correct (weighted average of data in rows five and six), so that any gene with at least four coding exons is likely to have at least two exons with $P\{\epsilon|S\} > 0.95$. For this reason, GENSCAN high probability exons in particular should prove very useful in combined computational/experimental gene identification efforts applied to newly sequenced human genomic regions. At the other extreme, approximately 9% of predicted exons have probability < 0.50: such exons are highly unreliable and should be treated with due caution in analyzing a genomic sequence. Thus, in a certain sense GENSCAN "knows" how much weight to

give to each of its predictions or, less anthropomorphically, the probabilistic model of gene structure employed bears a surprisingly close relationship to reality.

5.5 Applications of GENSCAN

This section addresses two of the most promising applications of GENSCAN, namely finding genes in newly sequenced genomic regions (even those in the publicly available databases), and the use of suboptimal exons to explore alternative gene structures in a sequence, which may in some cases indicate alternative splicing patterns or exons missed by the optimal parse.

5.5.1 Finding genes

A shortcoming of the Burset/Guigó and GeneParser test sets is that, by construction, they include only sequences containing single complete genes, usually with little $5'$ or $3'$ flanking sequence. Only one systematic test of a gene prediction program (GRAIL) on long human contigs (some containing multiple genes) has so far been reported in the literature (Lopez *et al.*, 1994), and the authors encountered a number of difficulties in carrying out this test. In particular, it was not always clear whether predicted exons not matching the annotation were false positives or might indeed represent real exons which had not been found by the original submitters of the sequence. For this reason, it was desireable to find long sequences containing multiple genes which had been well studied experimentally and were well annotated. The best such example found was GenBank sequence HSU47924 (accession # U47924), a recently sequenced contig 117 kb in length from human chromosome 12p13, in which six genes had been detected using a combination of computational and experimental approaches (Ansari-Lari *et al.*, 1996). Annotated genes, GENSCAN predicted genes, and GRAIL II predicted exons are displayed for this sequence for this sequence in Fig. 13. GENSCAN predicted genes are labeled GS1 through GS8; annotated genes are labeled with the gene name; GRAIL II, of course, predicts only exons, not genes. Forward strand exons are indicated above the sequence line, reverse-strand exons

below.[12]

GENSCAN predicted genes which are similar or identical to annotated (known) genes are as follows:

GS1 corresponds closely to the CD4 gene.[13]

GS2 is identical to one of the alternatively spliced forms of Gene A (function unknown).

GS3 contains several exons from both Gene B (function unknown) and GNB3 (G-protein beta-3 subunit).

GS5 is identical to ISOT (isopeptidase T), except for the addition of a predicted exon at around 74 kb.

GS6 is identical to TPI (triosephosphate isomerase), except with a different translation start site.

This leaves GS4, GS7 and GS8 as potential false positives, which do not correspond to any annotated gene, of which GS7 and GS8 are overlapped by GRAIL predicted exons.

A BLASTP (Altschul *et al.*, 1990) search of the predicted peptides corresponding to GS4, GS7 and GS8 against the nonredundant protein sequence databases revealed that:

GS8 is substantially identical (BLAST score 419, P = 2.6 E-57) to mouse 60S ribosomal protein (SwissProt accession # P47963).

GS7 is highly similar (BLAST score 150, P = 2.8 E-32) to *C. elegans* predicted protein C26E6.5 (GenBank accession # 532806).

GS4 is not similar to any known protein (no BLASTP hit with P < 0.01).

[12]This is the standard convention used for the graphical output of GENSCAN.
[13]The predicted exon at around 1.5 kb is actually a non-coding exon of CD4.

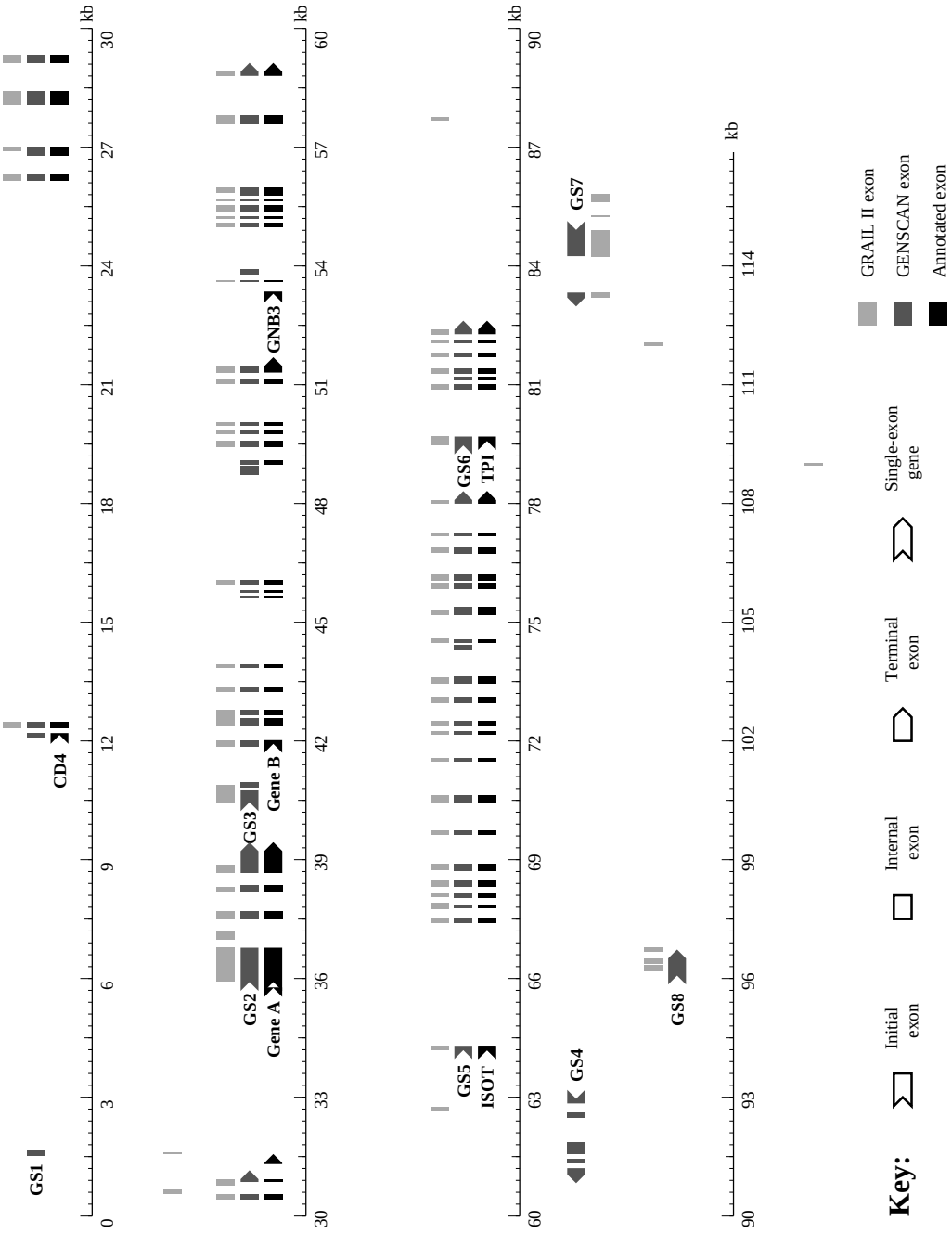


Fig. 13. GENSCAN and GRAIL II predictions for sequence HSU47924

Examination of the genomic sequence around GS8 suggests that this is probably a 60S ribosomal protein pseudogene. Predicted gene GS7 might be an expressed gene, but no hits were detected in the database of expressed sequence tags (dbEST) to confirm this, so it remains uncertain. However, several ESTs were found which were substantially identical to the predicted 3′ UTR and exons of GS4 (GenBank accession #s AA070439, W92850, AA055898, R82668, AA070534, W93300 and others), strongly implying that this is indeed an expressed human gene which was missed by the submitters of this sequence (probably because GRAIL did not detect it). Since all five predicted exons of GS4 had probabilities > 0.98, I considered it a virtual certainty that the predicted gene structure was substantially correct. This information was communicated to Dr. Martin Dyer (Academic Dept. of Haematology, Institute of Cancer Research, Surrey, U.K., visiting Stanford at the time), whose laboratory has since sequenced the full-length cDNA of this gene in both human and mouse, confirming the exact locations of all five predicted coding exons (Dyer & Burge, in preparation).

Aside from the prediction of this novel gene (which was not found by GRAIL and could not have been found by protein sequence homology search), this example also illustrates the potential of GENSCAN to predict the number of genes in a sequence fairly well. In particular, of the eight genes predicted, seven correspond closely to known or putative genes and only one (GS3) corresponds to a fusion of exons from two known genes. This example also illustrates some of the difficulties which arise in testing gene finding programs on long genomic contigs, since even in this relatively well characterized region, several errors (omissions) were found in the GenBank annotation. Proper tests of the ability of GENSCAN to predict the number of genes in a sequence will have to await construction of reliably annotated datasets of long genomic contigs.

5.5.2 Suboptimal exons and alternative splices

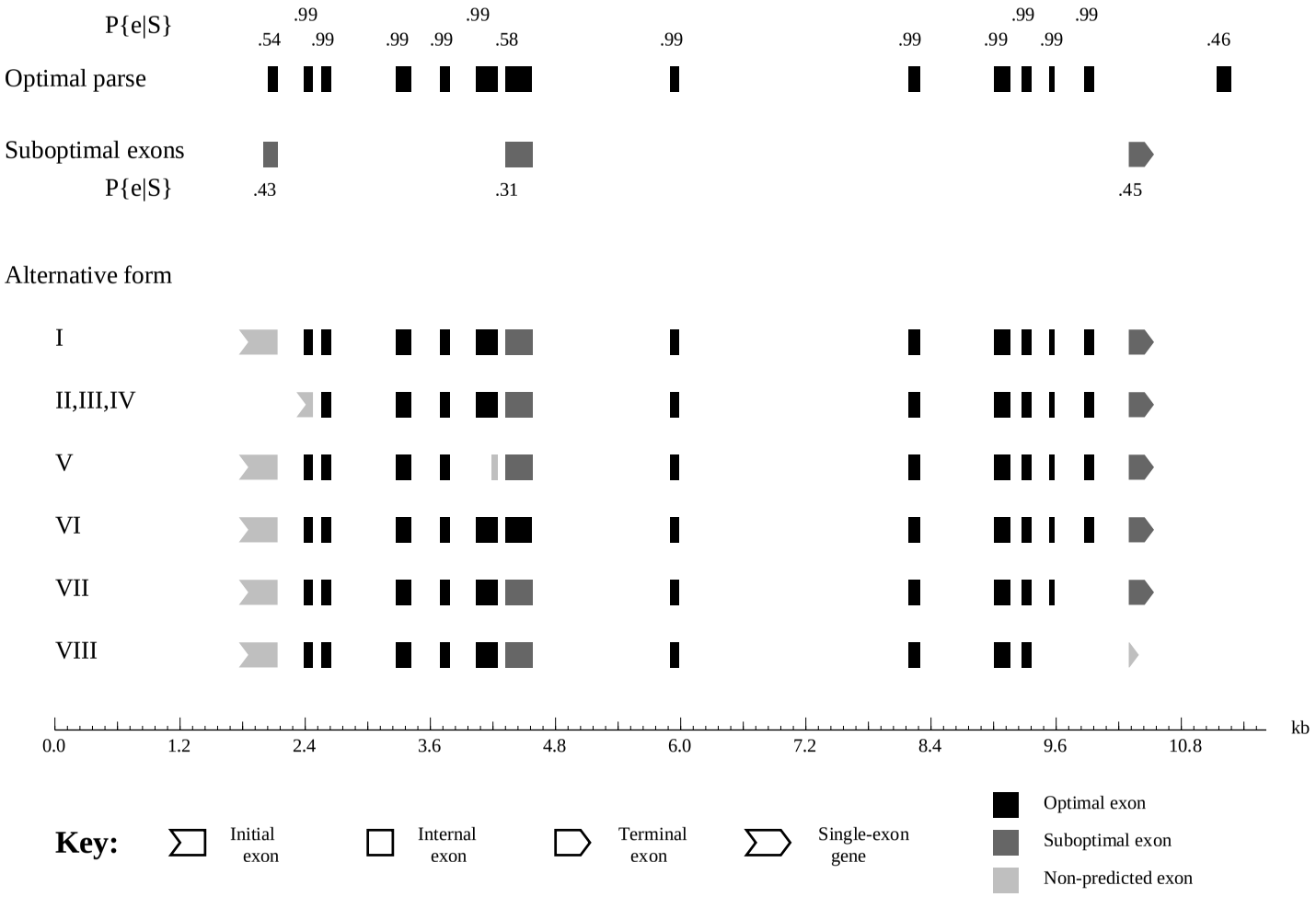
Two other issues of interest in the practical application of GENSCAN are the usefulness of suboptimal exons and the potential for prediction of alternative splicing

patterns of genes. These issues are again illustrated by means of an example, in this case GenBank sequence HUMPROK (accession # L00727), the genomic sequence of the human myotonin kinase (Mt-PK) gene. This gene, mutations of which are associated with the adult form of myotonic dystrophy (Fu *et al.*, 1993), was chosen because of the large number of known alternatively spliced forms (at least eight) and the high quality of the sequence annotation. Figure 14 displays the optimal GENSCAN parse of this sequence (black) together with all (three) suboptimal exons of probability greater than 0.25 (dark gray), and all annotated alternatively spliced forms.[14] Annotated exons were colored to match the predicted exons when exactly predicted, or shown in light gray otherwise. The exon probabilities are shown adjacent to each predicted exon for reference.

Several features are notable. First, the optimal parse consists mostly of very high probability exons ($P\{\epsilon|S\} > 0.99$), of which all are correct in some or all alternatively spliced forms. Of the three lower probability exons: the first overlaps the most commonly used initial exon; the second is an alternative internal exon; and the last is (apparently) incorrect (located in the $3'$ UTR for some isoforms and in an untranscribed region for others). Each of these lower probability exons has an associated suboptimal exon with which it is mutually incompatible: the first suboptimal exon overlaps the most commonly used initial exon; the second is an alternative internal exon (differing by only 15 bp from the other alternative); and the third is in fact the most commonly used terminal exon. Thus, suboptimal exons may be useful for at least two purposes: identification of exons missed by the optimal parse and, in some cases, indication of potential alternatively spliced regions of a gene.

[14]Of the eight known alternative splices, three (numbers II, III and IV) are identical in the coding region, only differing in the locations of introns in the $5'$ UTR.

Fig. 14. Optimal and suboptimal exons vs alternative splices: HUMPROK



Chapter 6

CONCLUSIONS

Previous chapters have described the development of a probabilistic model of the (gene) structural and sequence compositional properties of human genomic DNA and the application of this model to the problem of gene identification in unannotated genomic regions. The main accomplishment was the development of a new gene identification program, GENSCAN, which has several significant advantages over existing gene finding algorithms. Most importantly, predictive accuracy has been shown to be substantially higher for GENSCAN than than for any other available method when tested on standardized sets of human and vertebrate genomic sequences. In particular, the program is able to identify 70 to 80% of exons in a genomic sequence precisely, with even higher levels of accuracy observed for complex genes containing ten or more exons. Furthermore, consistently high levels of accuracy have been attained for sequences of differing C+G% content and the program performs almost as well on rodent and non-mammalian vertebrate sequences as for human sequences. The program version developed for human genomic sequences is even capable of identifying approximately two thirds of exons exactly in maize and *Drosophila* sequences, implying that the model incorporates gene structural and sequence properties which are of fundamental importance in most or all higher eukaryotes.

Other important novelties are the ability to treat partial as well as complete genes and the ability to predict multiple genes, occurring on either or both DNA strands,

in a single sequence. These properties should make the program particularly useful for analysis of the long genomic contigs which are being generated at an increasing rate by the Human Genome Project and related genome sequencing efforts. Another noteworthy feature of the program is its ability to assign a meaningful reliability measure, the exon probability ($P\{\epsilon|S\}$), to each predicted exon, which gives the user a highly informative guide as to the degree of confidence which should be ascribed to each aspect of a prediction. In some cases at least, suboptimal exons can indicate potential alternatively spliced regions of a predicted gene. Finally, the potential use in gene finding was demonstrated by the detection of a novel gene not homologous to any known protein in a published human genomic sequence. (Several other such examples are currently being investigated.)

Of existing algorithms, GENSCAN is most similar in its overall architecture to the recently developed Genie program (Kulp *et al.*, 1996), which uses a "generalized Hidden Markov Model" of gene structure which is fundamentally similar to the model described here. Genie, developed at the same time as GENSCAN in a collaboration between groups at U. C. Santa Cruz and Lawrence Berkeley National Laboratories, is not as general as GENSCAN, however, in that it does not include: 1) promoter or poly-adenylation signals or intergenic regions; 2) intronless genes; 3) partial genes; 4) multiple genes in the same sequence; 5) signal peptides; or 6) differences in gene structure between different isochore compartments. There are other substantial differences as well, e.g., Genie uses neural network models for coding regions and splice signals rather than strictly probabilistic models. Finally, as discussed previously, the accuracy of GENSCAN is much higher than that reported for Genie (Table 10), although recent improvements in Genie have narrowed this gap (M. G. Reese, personal communication).

Aside from the specific goal of gene prediction, other aspects of this work may be of some independent interest. Specifically, the Maximal Dependence Decomposition (MDD) method (Section 4.3), developed to capture dependencies between positions in the donor splice site signal, may prove useful in modeling other biological signals in nucleic acid or protein sequences. The MDD procedure has certain advantages over alternative methods such as (artificial) neural networks (e.g., Brunak *et al.*, 1991) in

that the method clearly indicates which dependencies are of primary and secondary importance and since all parameters are explicit and can be easily interpreted. In the case of the donor splice signal, at least, these dependencies are quite interesting in themselves and suggest subtle features of the mechanism of donor site selection, some of which could be tested experimentally.

The technique devised for smoothing sparse empirical length distributions (Section 3.4) may also be a useful statistical tool in a variety of contexts, e.g., in estimating smooth underlying distributions of protein lengths or in studying the evolutionary divergence of the lengths of other biological structures such as introns and repetitive elements. The studies of the relationships between structural and compositional properties of human genes (Chapter 3) may also be of interest with regard to certain questions about the evolution of genome organization. In particular, the dramatic increase in the sizes of introns and intergenic regions in A+T rich regions of the genome suggests that special mutational, repair or selective forces may be at work to expand such regions or, conversely, to reduce the lengths of intronic and intergenic regions in C+G rich portions of the genome.

It is worthwhile at this point to consider this work from a somewhat broader point of view, which leads naturally to some other potentially interesting applications of the probabilistic framework to areas beyond gene finding. From this point of view, Chapters 3 and 4 can be represented by the relation, $\{S, \Phi\} \rightarrow M$, in the sense that a set of sequences, S, with known gene locations (Φ) were used to derive a specific model description, M. In this framework, use of the model for gene prediction (described in Chapter 2), is represented by the relation $\{M, S\} \rightarrow \Phi$, in the sense that the optimization (Viterbi) algorithm determines a specific parse (ϕ_{opt}) for a given sequence, S, under the model specification M. This raises the obvious question of whether the third permutation, $\{M, \Phi\} \rightarrow S$, has any significance. Since, in general, many sequences are consistent with a given parse and model specification, the interpretation of this relation should be the generation of a (random) sequence or sequences (using the probabilities specified by the model, M) corresponding to a pre-specified parse.

Of what use might this be? Consider the computational experiment of generating a large number of such random sequences corresponding to a set of actual gene specifications: compositional differences between such randomly generated sequences and actual sequences could then provide information about signals or other biological properties not adequately represented by the model, M. Such randomly generated sequences might have other uses as well. For instance, if it were of interest to determine whether the distribution of a particular sequence motif (e.g., secondary structural element, enhancer signal, etc.) in a genomic sequence is intrinsically unusual or only appears unusual due to its differing frequency of occurrence in coding vs non-coding regions, then using the model to generate random sequences consistent with the given gene locations might provide a much more realistic type of control than would otherwise be available.

Other applications of the model framework may also be of interest. In particular, consider deriving two (or more) sets of model parameters, M_1 and M_2, perhaps from different organisms or different classes of genes. Then, given a sequence S, the probabilities $P\{S|M_1\}$ and $P\{S|M_2\}$ could be calculated, using the "forward" algorithm (Section 2.7) and used to predict which organism or class of genes the sequence belongs to. This application might prove useful, for instance, in the detection of genes which have been acquired by horizontal transfer from one genome to another, an issue of obvious importance in studying the evolution of genomes and in estimating phylogenetic divergence times. Thus, this work can be considered either as another step toward the goal of identifying all human genes (and all genes from a variety of model organisms), or in the more general context of methods for classification, analysis and comparison of genes and genomes.

APPENDIX A

GENSCAN LEARNING SET

The sets of GenBank loci constructed as a standard for training and testing of gene finding methods by D. Kulp (University of California at Santa Cruz) and M. G. Reese (Lawrence Berkeley National Laboratories), August 22, 1995 [ftp://ftp.cse.ucsc.edu/pub/dna/genes] were used as a starting point for construction of the GENSCAN learning set. These sets were derived by screening GenBank (release 89, 1995) for all sequences meeting the following criteria:

ORGANISM Homo sapiens

exactly one CDS feature (to avoid alternatively spliced genes)

first exon begins with ATG

last exon ends with stop codon: no other in-frame stop codons

all splice sites match minimal consensus (acceptor: AG, donor: GT)

Separate single- and multi-exon gene sets were derived and culled of redundant or substantially similar (BLAST score $\geq$ 100) entries using BLASTP (Altschul *et al.*, 1990) with default parameters. I further cleaned this set by removing genes whose annotation indicated any of the following:

alternative splicing

partial or putative CDS location or ORF designation

result of non-productive rearrangement

viral or mitochondrial origin

submitted to GenBank by NCBI staff, not original sequencer of gene

suspiciously short, even intron lengths (suggesting incomplete sequencing of introns)

This procedure resulted in a penultimate set of 428 sequences. From this set were removed all genes $\geq 25\%$ identical at the amino acid level to any gene from GeneParser test sets I and II (Snyder & Stormo, 1995) using the PROSET program (Brendel, 1992) with default parameters. The final resulting set containing 380 genes is designated the GENSCAN learning set, $\mathcal{L}$. The 238 multi-exon genes of the learning set are listed below, followed by the 142 single-exon genes.

GenBank Locus	Accession	Definition
HS14B7	Z49258	Human DNA sequence from cosmid 14B7 in Xq28 containing
HS1D3HLH	X73428	H.sapiens Id3 gene for HLH type transcription factor.
HS2OXOC	X66114	H.sapiens gene for 2-oxoglutarate carrier protein.
HSACKI10	X14487	Human gene for acidic (type I) cytokeratin 10.
HSALADG	X64467	H.sapiens ALAD gene for porphobilinogen synthase.
HSAPC3A	X01392	Human apolipoprotein CIII gene and apo AI-apo CIII intergenic
HSAPOA2	X04898	Human gene for apolipoprotein AII.
HSAPOAIA	X01038	Human fetal gene for apolipoprotein AI precursor.
HSAPOC2G	X05151	Human apoC-II gene for preproapolipoprotein C-II.
HSARYLA	X52150	Human DNA for arylsulphatase A (EC 3.1.6.1).
HSASML	X63600	H.sapiens genes for acid sphingomyelinase ASM.
HSAT3	X68793	H.sapiens gene for antithrombin III.
HSATPCP1	X69907	H.sapiens gene for mitochondrial ATP synthase c subunit (P1 form).
HSB3A	X72861	H.sapiens gene for beta-3-adrenergic receptor.
HSBCDIFFI	X12706	H.sapiens gene for B cell differentiation factor I.
HSBGPG	X04143	Human gene for bone gla protein (BGP).
HSBSF2	Y00081	Human (BSF-2/IL6) gene for B cell stimulatory factor-2.
HSC1INHIB	X54486	Human gene for C1-inhibitor.
HSCBMYHC	X52889	Human gene for cardiac beta myosin heavy chain.
HSCD14G	X06882	Human gene for CD14 differentiation antigen.
HSCD1R3	X14974	Human CD1 R3 gene for MHC-related antigen.
HSCD7	X06180	Human mRNA for CD7 antigen (gp40).
HSCKBG	X15334	Human gene for creatine kinase B (EC 2.7.3.2).
HSCKIIBE	X57152	Human gene for casein kinase II subunit beta (EC 2.7.1.37).
HSCOMT2	Z26491	H.sapiens gene for catechol O-methyltransferase.
HSCOSE	X62891	H.sapiens mutant coseg gene for vasopressin-neurophysin precursor.
HSCPH70	X52851	Human cyclophilin gene for cyclophilin (EC 5.2.1.8).
HSCSF1PO	X14720	Human c-fms proto-oncogene for CSF-1 receptor.
HSCST3G	X52255	Human CST3 gene for cystatin C.
HSCTAS	Z18859	H.sapiens gene for cone transducin alpha subunit.
HSCYCLA	X68303	H.sapiens cycA gene for cyclin A.
HSCYP450	X02612	Human gene for cytochrome P(1)-450.
HSDAO	X78212	H.sapiens diamine oxidase gene.
HSDNAMIA	X84707	H.sapiens MIA gene.
HSENAGENO	Z46254	H.sapiens gene for neutrophil-activating peptide 78 (ENA-78).
HSENO2	X51056	Human ENO2 gene for neuron-specific (gamma) enolase

HSGTRH	X15215	Human gene for gonadotropin-releasing hormone.
HSHCF1	X79198	H.sapiens HCF-1 gene.
HSHH3X3B	Z48950	H.sapiens hH3.3B gene for histone H3.3.
HSHLADMBG	X76776	H.sapiens HLA-DMB gene.
HSHLADZA	X02882	Human HLA class II alpha chain gene DZ-alpha.
HSHLIA	X00492	Human gene for histocompatibility antigen HLA-A3.
HSHOX3D	X61755	Human HOX3D gene for homeoprotein HOX3D.
HSHSC70	Y00371	Human hsc70 gene for 71 kd heat shock cognate protein.
HSIFNAR	X60459	Human IFNAR gene for interferon alpha/beta receptor.
HSIFNG	V00536	Human immune interferon (IFN-gamma) gene.
HSIGK12	Z00010	H.sapiens germ line pseudogene for immunoglobulin kappa light chain
HSIL05	X00695	Human interleukin-2 (IL-2) gene and 5'-flanking region.
HSIL1AG	X03833	Human gene for interleukin 1 alpha (IL-1 alpha).
HSIL1B	X04500	Human gene for prointerleukin 1 beta.
HSIL1RECA	X64532	H.sapiens gene for interleukin-1 receptor antagonist.
HSINSU	V00565	Human gene for preproinsulin, from chromosome 11. Includes a highly
HSINT1G	X03072	Human int-1 mammary oncogene.
HSINT2	X14445	Human int-2 proto-oncogene.
HSL7A	X52138	Human L7a gene for large ribosomal subunit component (L7a).
HSLCATG	X04981	H.sapiens gene for lecithin-cholesterol acyltransferase (LCAT).
HSMECDAG	X62654	H.sapiens gene for Me491/CD63 antigen.
HSMED	Y00477	Human bone marrow serine protease gene (medullasin) (leukocyte
HSMGSAG	X54489	Human gene for melanoma growth stimulatory activity (MGSA).
HSMHCPU15	Z14977	H.sapiens gene for major histocompatibility complex encoded
HSMOGG	Z48051	H.sapiens gene for myelin oligodendrocyte glycoprotein (MOG).
HSMTS1G	Z33457	H.sapiens mts1 gene.
HSNCAMX1	Z29373	H.sapiens gene for neural cell adhesion molecule L1.
HSNFM	Y00067	Human gene for neurofilament subunit M (NF-M).
HSODCG	X16277	Human gene for ornithine decarboxylase ODC (EC 4.1.1.17).
HSODF2	X74614	H.sapiens ODF2 (allele 2) gene for outer dense fiber protein.
HSP53G	X54156	Human p53 gene for transformation related protein p53 (also called
HSPAT133	X69438	H.sapiens zinc finger gene pAT133.
HSPR264SC	X75755	H.sapiens PR264 gene.
HSPRB3L	X07881	Human gene PRB3L for proline-rich protein G1.
HSPROPG	X70872	H.sapiens gene for properdin.
HSRODPDE	X62025	H.sapiens rod cG-PDE G gene for 3', 5'-cyclic nucleotide
HSRPII145	Z23102	H.sapiens gene for RNA polymerase II 14.5 kDa subunit.
HSRPS6G	X67309	H.sapiens gene for ribosomal protein S6.

HSU08198	U08198	Human complement C8 gamma subunit precursor (C8G) gene, complete
HSU09954	U09954	Human ribosomal protein L9 gene, 5' region and complete cds.
HSU12421	U12421	Human mitochondrial benzodiazepine receptor (MBR) gene, complete
HSU17969	U17969	Human initiation factor eIF-5A gene, complete cds.
HSU19765	U19765	Human nucleic acid binding protein gene, complete cds.
HSU19816	U19816	Human thyroid transcription factor-1 (TTF-1) gene, complete cds.
HSU20325	U20325	Human cocaine and amphetamine regulated transcript CART (hCART)
HSU20499	U20499	Human thermolabile phenol sulfotransferase (stm) gene, complete
HSU20982	U20982	Human insulin-like growth factor binding protein-4 (IGFBP4) gene,
HSU23143	U23143	Human mitochondrial serine hydroxymethyltransferase gene, nuclear
HSU23853	U23853	Human dual-specific phosphoprotein phosphatase (PAC1) gene,
HSU25826	U25826	Human transcription factor (SC1) gene, complete cds.
HSU26425	U26425	Human phospholipase C-beta-3 (PLCB3) gene, complete cds.
HSUBA52G	X56997	Human UbA52 gene coding for ubiquitin-52 amino acid fusion protein.
HSUBR	X76498	H.sapiens gene for uterine bombesin receptor.
HSZNGP1	X69953	H.sapiens gene for ZN-alpha-2-glycoprotein.
HUM6PTS	D25234	Human gene for 6-pyruvoyl-tetrahydropterin synthase.
HUMA1ATP	K02212	Human alpha-1-antitrypsin gene (S variant), complete cds.
HUMA1GLY2	M21540	Human alpha-1-acid glycoprotein 2 (AGP2) gene, complete cds.
HUMACCYBB	M10277	Human cytoplasmic beta-actin gene, complete cds.
HUMADAG	M13792	Human adenosine deaminase gene, complete cds.
HUMADPRF02	M74493	Human ADP-ribosylation factor 3 gene, exons 2-5.
HUMAFP	M16110	Human alpha-fetoprotein gene, complete cds.
HUMAHCY	M61831	Human S-adenosylhomocysteine hydrolase (AHCY) mRNA, complete cds.
HUMAK1	J04809	Human cytosolic adenylate kinase (AK1) gene, complete cds.
HUMALIFA	M63420	Human leukemia inhibitory factor (LIF) gene, complete cds.
HUMANFA	K02043	Human atrial natriuretic factor (PND) gene, complete cds.
HUMANT1	J04982	Human heart/skeletal muscle ATP/ADP translocator (ANT1) gene,
HUMAPOA4C	M14642	Human apolipoprotein A4 (APOA4) gene, exons 1, 2 and 3.
HUMAPOCIA	M20902	Human apolipoprotein C-I (VLDL) gene, complete cds.
HUMAPOE4	M10065	Human apolipoprotein E (epsilon-4 allele) gene, complete cds.
HUMATP1A2	J05096	Human Na,K-ATPase subunit alpha 2 (ATP1A2) gene, complete cds.
HUMATPSAS	D28126	Human gene for ATP synthase alpha subunit.
HUMATPSYB	M27132	Human ATP synthase beta subunit (ATPSB) gene, complete cds.
HUMBFXIII	M64554	Human factor XIII b subunit gene, complete cds.
HUMBHSD	M38180	Human 3-beta-hydroxysteroid dehydrogenase/delta-5-delta-4-isomerase
HUMBLYM1	K01884	Human Blym-1 transforming gene, complete coding region.
HUMBNPA	M31776	Human brain natriuretic peptide (BNP) gene, complete cds.

HUMDS	D26535	Human gene for dihydrolipoamide succinyltransferase, complete cds
HUMEDHB17	M27138	Human estradiol 17 beta-dehydrogenase gene, complete cds.
HUMEDN1B	J05008	Homo sapiens endothelin-1 (EDN1) gene, complete cds.
HUMEF1A	J04617	Human elongation factor EF-1-alpha gene, complete cds.
HUMELAFIN	D13156	Human gene for elafin, complete cds.
HUMEPOHYDD	L29766	Homo sapiens epoxide hydrolase (EPHX) gene, complete cds.
HUMFABP	M18079	Human, intestinal fatty acid binding protein gene, complete cds,
HUMFIXG	K02402	Human factor IX gene, complete cds.
HUMG0S8PP	L13391	Human helix-loop-helix basic phosphoprotein (G0S8) gene, complete
HUMGAD45A	L24498	Human gadd45 gene, complete cds.
HUMGARE	L10822	Human gastrin receptor gene, complete cds.
HUMGAST2	K01254	Human gastrin gene, complete cds.
HUMGCAPB	L36861	Homo sapiens guanylate cyclase activating protein (GCAP) gene exons
HUMGCB1	J03059	Human glucocerebrosidase (GCB) gene, complete cds.
HUMGCK	M93280	Human glucokinase (GCK) gene, exons 1a-10.
HUMGLUT4B	M91463	Human glucose transporter (GLUT4) gene, complete cds.
HUMHCF2	M58600	Human heparin cofactor II (HCF2) gene, exons 1 through 5.
HUMHIS102	L04132	Human histatin 1 (HIS1) gene exons 1-5, complete cds.
HUMHMG14A	M21339	Human non-histone chromosomal protein HMG-14 gene, complete cds.
HUMHMG2A	M83665	Human high mobility group 2 protein (HMG-2) gene, complete cds.
HUMHMGIY	L17131	Human high mobility group protein (HMG-I(Y)) gene exons 1-8,
HUMHPRTB	M26434	Human hypoxanthine phosphoribosyltransferase (HPRT) gene, complete
HUMHSKPQZ7	M81806	Human housekeeping (Q1Z 7F5) gene, exons 2 through 7, complete cds.
HUMHSP27X	L39370	Human heat shock protein 27 (HSPB1) gene exons 1-3, complete cds.
HUMHSP89KD	M27024	Homo sapiens heat shock protein (HSP89-alpha) gene, complete cds.
HUMIDS	L35485	Homo sapiens iduronate sulphate sulphatase (IDS) gene, complete
HUMIFNRF1A	L05072	Homo sapiens interferon regulatory factor 1 gene, complete cds.
HUMIGERA	L14075	Homo sapiens immunoglobulin receptor alpha chain gene, complete
HUMIL2RGA	L19546	Human (IL2RG) gene, complete cds with repeats.
HUMIL4A	M23442	Human interleukin 4 (IL-4) gene, complete cds.
HUMIL9RA	L39064	Homo sapiens interleukin 9 receptor (IL9R) gene, complete cds.
HUMIMPDH	L33842	Homo sapiens (clone FFE-7) type II inosine monophosphate
HUMIRBPG	J05253	Human interstitial retinol-binding protein (IRBP) gene, complete
HUMLHDC	D16583	Human gene for L-histidine decarboxylase, complete cds.
HUMLYL1B	M22638	Human LYL-1 protein gene, complete cds.
HUMLYTOXBB	L11016	Homo sapiens lymphotoxin-beta gene, complete cds.
HUMMCHEMP	M37719	Human monocyte chemotactic protein gene, complete cds.
HUMMHCD8A	M27161	Human MHC class I CD8 alpha chain (Leu-2/T8) gene, complete cds.

HUMPBIPB	D49493	Human gene for human prepro bone inducing protein.
HUMPCNA	J04718	Human proliferating cell nuclear antigen (PCNA) gene, complete cds.
HUMPDHAL	D90084	Human pyruvate dehydrogenase (EC 1.2.4.1) alpha subunit gene, exons
HUMPEPYYA	L25648	Human peptide YY gene, complete cds.
HUMPGAMMG	J05073	Human phosphoglycerate mutase (PGAM-M) gene, complete cds.
HUMPHOSA	L12760	Human phosphoenolpyruvate carboxykinase (PCK1) gene, complete cds
HUMPIM1A	M27903	Human pim-1 proto-oncogene gene, complete cds.
HUMPKD1GEN	L39891	Homo sapiens polycystic kidney disease-associated protein (PKD1)
HUMPPPA	M11726	Human pancreatic polypeptide gene, complete cds.
HUMPRF1A	M31951	Human perforin (PRF1) gene, complete cds.
HUMPSAP	M30838	Human pulmonary surfactant apoprotein (PSAP) gene, complete cds.
HUMPTH2	J00301	Human parathyroid (pth) gene: coding region and 3'flank.
HUMRBPA	L34219	Homo sapiens retinaldehyde-binding protein (CRALBP) gene, complete
HUMRCC1	D00591	Human RCC1 gene, complete cds.
HUMRETBLAS	L11910	Human retinoblastoma susceptibility gene exons 1-27, complete cds.
HUMRIGA	M32405	Human homologue of rat insulinoma gene (rig), exons 1-4.
HUMRIGBCHA	M89796	Human high affinity IgE receptor beta chain gene, complete cds.
HUMROD1X	M96759	Human rod outer segment membrane protein 1 (ROM1) gene exons 1-3,
HUMRPS17A	M18000	Human ribosomal protein S17 gene, complete cds.
HUMSEMI	M81650	Human semenogelin I (SEMGI) gene, complete cds.
HUMSOMI	J00306	Human somatostatin I gene and flanks.
HUMSPBAA	M24461	Human pulmonary surfactant-associated protein SP-B (SFTP3) mRNA,
HUMSPERSYN	M64231	Human spermidine synthase gene, complete cds.
HUMSTATH2	M32639	Human salivary statherin gene, exons 2-6.
HUMTA	D32046	Human gene for thrombopoietin.
HUMTBGA	L13470	Human thyroxine-binding globulin gene, complete cds.
HUMTDGF1A	M96955	Human (clone CR) teratocarcinoma-derived growth factor 1 (TDGF1)
HUMTFPB	J02846	Human tissue factor gene, complete cds.
HUMTHY1A	M11749	Human Thy-1 glycoprotein gene, complete cds.
HUMTNP1	M59924	Human transition protein 1 gene, complete cds.
HUMTPA	K03021	Human tissue plasminogen activator (t-PA) gene, complete cds.
HUMTPALBU	L14927	Human tear prealbumin (TP) gene, complete cds and promoter region.
HUMTROC	M37984	Human slow twitch skeletal muscle/cardiac muscle troponin C gene,
HUMTS1	D00596	Human thymidylate syntase (EC 2.1.1.45) gene, complete cds.
HUMTSHB2	M21024	Human thyrotropin beta (TSH-beta) subunit gene, exons 2 and 3.
HUMUBILP	J03589	Human ubiquitin-like protein (GdX) gene, complete cds.
HUMVCAM1A	M73255	Human vascular cell adhesion molecule-1 (VCAM1) gene, complete CDS.
HUMVIPAA	M33987	Human vasoactive intestinal peptide/PHM-27 gene, exons 1-6.

Single-exon genes. Format: LOCUS (ACCESSION).

HS0MGP (X57436)
HS1433PR (X80536)
HSAACT (X14672)
HSACTHR (X65633)
HSACTREC (X63128)
HSADSS (X66503)
HSANTENII (Z11162)
HSBAR (Y00106)
HSBFCRII (X52473)
HSBNGF (V01511)
HSBPIP (X68790)
HSCAR27 (X65784)
HSCENPB (X55039)
HSCIC1MCC (Z25587)
HSCKIIAL (X70251)
HSCNTFG (X60542)
HSCOLA1X (X60382)
HSCREBA (X55545)
HSD1DO (X55760)
HSDNAJ (X62421)
HSDRK1 (X68302)
HSEAR2 (X12794)
HSECP1 (X16545)
HSFKBPA (X55741)
HSGDF5 (X80915)
HSGLUDP1 (X66310)
HSGPV (Z23091)
HSH11 (X57130)
HSH2B1 (X57127)
HSH4AHIS (X60481)
HSHAIL8G (X65858)
HSHB2A (X63337)
HSHGM071 (X64994)
HSHGMP07J (X64995)
HSHIS10G (X03473)
HSHISH1 (X76786)
HSHISH2A (X00089)
HSHM3 (X15265)
HSIFD2 (V00532)
HSKERUHS (X55293)
HSNFIL6 (X52560)
HSNTFR (X60201)
HSOTF3CG (Z11901)
HSP3 (X12458)
HSPCCBA (X73424)
HSPCRF (V00571)
HSPGK2G (X05246)
HSPLMERE (X13556)
HSPRP2 (X83416)
HSPRPE2 (X53065)
HSRIB1 (X79235)
HSRNAP14K (Z27113)
HSSECONC (X52259)
HSSIAL (X52075)
HSSOX3 (X71135)
HSSPHAR (X82554)
HSTREB5A (X55543)
HSTRELFA (X73534)
HSTYRPH (X82676)
HSU01212 (U01212)
HSU03486 (U03486)
HSU03735 (U03735)
HSU10116 (U10116)
HSU10273 (U10273)
HSU10360 (U10360)
HSU10554 (U10554)
HSU11424 (U11424)
HSU13666 (U13666)
HSU13695 (U13695)
HSU16812 (U16812)
HSU17894 (U17894)
HSU18548 (U18548)
HSU20734 (U20734)
HSU21051 (U21051)
HSU22346 (U22346)
HUM25RNASE (L10381)
HUMA2CIIA (D13538)
HUMABRA (L19704)
HUMAGG (M11567)
HUMANONYMO (L18972)
HUMASPA (L37019)
HUMATCT4A (M35160)
HUMB1LYM (M27394)
HUMBTFC (M90355)
HUMBTFD (M90356)
HUMCALCHAA (M92269)
HUMCDR34 (M31423)
HUMCMOS (J00119)
HUMCNGCCA (L15296)
HUMCSPC (M28170)
HUMCSYNA (M14333)
HUMENIGMA (L35240)
HUMEP2AA (M60119)
HUMEPC1X (M90439)
HUMEVI22 (M55267)
HUMEVI2B3P (M60830)
HUMFPR1A (L10820)
HUMFSRSA (D16826)
HUMG0S2A (M69199)
HUMGA733A (J04152)
HUMGLUDECA (M86522)
HUMGPA (M16514)
HUMGPIBAA (M22403)
HUMGPIX (M80478)
HUMGPR5A (L36149)
HUMH1T (M60094)
HUMHEN2A (M97508)
HUMHISH2R (M64799)
HUMHLGS (D29685)
HUMIL2AB (M22005)
HUMISK (M26685)

APPENDIX B

GENSCAN TEST SET

The sets of GenBank loci constructed by D. Kulp (University of California at Santa Cruz) and M. G. Reese (Lawrence Berkeley National Laboratories), were updated in the summer of 1996 [http://www-hgc.lbl.gov/inf/genesets.html] using sequences from GenBank release 95. The criteria for inclusion were similar to those used for the original set (see Appendix A), except that only multi-exon genes were included. This set was subsequently cleaned by removing all genes $> 25\%$ identical at the protein level to any gene of the previous set using the PROSET program (Brendel, 1992) with default parameters. This resulted in a set of 65 genes termed the GENSCAN test set, $\mathcal{T}$, listed below.

GenBank Locus	Accession	Definition
D13752	D13752	Human CYP11B2 gene for steroid 18-hydroxylase, complete cds.
D83195	D83195	Human DNA for Deoxyribonuclease I precursor.
HSALDOA	X12447	Human aldolase A gene (EC 4.1.2.13).
HSCYTOK20	X73501	H.sapiens gene for cytokeratin 20.
HSDNAAMHI	X89013	H.sapiens gene for anti-mullerian hormone type II receptor.
HSHCC1GEN	Z49269	H.sapiens gene for chemokine HCC-1.
HSMB1GENE	X95586	H.sapiens MB1 gene.
HSNFLG	X05608	Human gene for neurofilament subunit NF-L.
HSPACAP	X60435	H.sapiens gene PACAP for pituitary adenylate cyclase activating
HSQC8B6	Z68193	Human DNA sequence from cosmid QC8B6, on chromosome Xq28,
HSRA36	Z69720	Human DNA sequence from cosmid RA36 from a contig from the tip of
HSRPS3AGE	X87373	H.sapiens RPS3a gene.
HSU07807	U07807	Human metallothionein IV (MTIV) gene, complete cds.
HSU10307	U10307	Human interleukin 13 (IL13) gene, complete cds.
HSU16720	U16720	Human interleukin 10 (IL10) gene, complete cds.
HSU19906	U19906	Human arginine vasopressin receptor 1 (AVPR1) gene, complete cds.
HSU22027	U22027	Human cytochrome P450 (CYP2A6V2) gene, complete cds.
HSU24685	U24685	Human anti-B cell autoantibody IgM heavy chain variable V-D-J
HSU30787	U30787	Human uroporphyrinogen decarboxylase (URO-D) gene, complete cds.
HSU31767	U31767	Human neuronatin gene, complete cds.
HSU31929	U31929	Human orphan nuclear receptor (DAX1) gene, complete cds.
HSU32323	U32323	Human interleukin-11 receptor alpha chain gene, complete cds.
HSU32576	U32576	Human apolipoprotein apoC-IV (APOC4) gene, complete cds.
HSU33446	U33446	Human prostasin gene, complete cds.
HSU37022	U37022	Human cyclin-dependent kinase 4 (CDK4) gene, complete cds.
HSU43415	U43415	Human obese (ob) gene, complete cds.
HSU43572	U43572	Human alpha-N-acetylglucosaminidase (NAGLU) gene, complete cds.
HSU43901	U43901	Human 37 kD laminin receptor precursor/p40 ribosome associated
HSU46692	U46692	Human cystatin B gene, complete cds.
HSU46920	U46920	Human metaxin (MTX) gene, complete cds.
HSU48795	U48795	Human antimicrobial protein CAP18 precursor gene, complete cds.
HSU48869	U48869	Human cdk-inhibitor p57/KIP2 (CDKN1C) gene, complete cds.

HSU50136	U50136	Human leukotriene C4 synthase (LTC4S) gene, complete cds.
HSU50871	U50871	Human familial Alzheimer's disease (STM2) gene, complete cds.
HSU51899	U51899	Human kappa-casein gene, complete cds.
HSUNGGENE	X89398	H.sapiens ung gene for uracil DNA-glycosylase.
HUMAZCDI	M96326	Human azurocidin gene, complete cds.
HUMBETGLOA	L26462	Human haplotype C4 beta-globin gene, complete cds.
HUMCACY	J02763	Human calcyclin gene, complete cds.
HUMCHYMASE	M64269	Human mast cell chymase gene, complete cds.
HUMCOL2A1Z	L10347	Human pro-alpha1 type II collagen (COL2A1) gene exons 1-54,
HUMCOX5B	M59250	Homo sapiens cytochrome c oxidase subunit Vb (COX5B) gene, complete
HUMCYP2DG	M33189	Human debrisoquine 4-hydroxylase mutant allele (CYP2D6-MA1) gene,
HUMCYPIIE	J02843	Human cytochrome P450IIE1 (ethanol-inducible) gene, complete cds.
HUMDKERB	M34482	Human cytokeratin 8 (CK8) gene, complete cds.
HUMDNL1L	L40817	Homo sapiens muscle-specific DNase I-like (DNL1L) gene, exons 1-9,
HUMDODDA	L39874	Homo sapiens deoxycytidylate deaminase gene, complete cds.
HUMG0S19A	M23178	Human homologue-1 of gene encoding alpha subunit of murine cytokine
HUMGALT54X	L48714	Homo sapiens galactose-1-phosphate uridyl transferase (GALT) mutant
HUMHA2WC	D31846	Human gene for aquaporin-2 water channel.
HUMHOX4A	D11117	Human homeobox HOX 4A gene for homeodomain protein, complete cds.
HUMHPD	D31628	Human gene for 4-hydroxyphenylpyruvic acid dioxygenase (HPD),
HUMIBP3	M35878	Human insulin-like growth factor-binding protein-3 gene, complete
HUMKALLIST	L28101	Homo sapiens kallistatin (PI4) gene, exons 1-4, complete cds.
HUMMKXX	M94250	Human retinoic acid inducible factor (MK) gene exons 1-5, complete
HUMNUCLEO	M60858	Human nucleolin gene, complete cds.
HUMP45C17	M19489	Human P450XVIIA-1 (steroid 17-alpha-hydroxylase/17,20 lyase) gene,
HUMPCBD	L41560	Homo sapiens (clones HGPCD2 and HGPCD15) pterin-4a-carbinolamine
HUMPCI	M68516	Human protein C inhibitor gene, complete cds.
HUMPF4V1A	M26167	Human platelet factor 4 varation 1 (PF4var1) gene, complete cds.
HUMPRCA	M11228	Human protein C gene, complete cds.
HUMREGB	J05412	Human regenerating protein (reg) gene, complete cds.
HUMSAP01	D00097	Human serum amyloid P component (SAP) gene with upstream promoter.
HUMSFRS	L41887	Homo sapiens splicing factor, arginine/serine-rich 7 (SFRS7) gene,
HUMTNP2SS	L03378	Homo sapiens transition protein 2 (TNP2) gene, complete cds.

APPENDIX C

Test set of 202 *Drosophila melanogaster* genomic sequences.

This set of 202 GenBank loci [ftp://www-hgc.lbl.gov/pub/genesets/dro] was constructed by D. Kulp (U. C. Santa Cruz) and M. G. Reese (Lawrence Berkeley National Laboratories) on 12 Dec. 1996 as a standard for training and testing of gene finding programs. Criteria used in construction of this set were similar to those used for human genes (Appendices A and B).

Format: LOCUS (ACESSION).

DMADH (X78384) DMANPG (X56726) DMANX (X78323)
DMATTACIN (Z46893) DMAURG (X83466) DMBCDG (X07870)
DMBJ1G (X58530) DMBSG25D (X04896) DMBTDGN (Z29361)
DMBX200 (X13168) DMCALRET (X64461) DMCHORS16 (X16715)
DMCOPIAV (X54147) DMCSDUC (X77936) DMCYP4D2 (X75955)
DMCYSTA (X55178) DMCZSUDMA (Z19591) DMDEADBXA (Z23266)
DMDNAMIN (X91853) DMDNARPL9 (X94613) DMDRCIV2 (X16968)
DMDTFIIAS (X83271) DMEF1AF2 (X06870) DMEHAB (X72303)
DMELGG (X68259) DMFBP1 (X69965) DMFUSED (X80468)
DMGIANT (X61148) DMGTPBP (X71866) DMH2AVDG (X15549)
DMHAIRG (X15904) DMHGSG2 (X07311) DMK10G (X12836)
DMKA12ADH (X60791) DMKNIRPS (X13331) DMKR (X03414)
DML2AMD (X04695) DMLAMIN (X16275) DMLAMINC (X75886)
DMLETHAL2 (Z48443) DMMBNGEN (Z47722) DMMGN (U03559)
DMMP20 (Y00795) DMMTNG (X03758) DMMTOG (X52098)
DMP11 (X59691) DMPCGENE (X55702) DMPER (X03636)
DMPGKG (Z14029) DMPPGENE (X69828) DMPRUNEG (Z12141)
DMPS35 (X62285) DMPUFFSP (X64536) DMR118C (X16962)
DMRAFPO (X07181) DMRLB1A (X73216) DMRLC1B (X73218)
DMRNPOL2 (X05709) DMRP128 (X58826) DMRP49 (X00848)
DMRPL19 (X74776) DMRPL7A (X82782) DMRPS3 (X72921)
DMSAL (X57474) DMSGS5 (X04269) DMSPXGENE (X97197)
DMSSRP2GN (X89811) DMSTELL (X15899) DMSUHW (Y00228)
DMSWAL (X56023) DMTFIIB (U02879) DMTOPII (X61209)
DMTORSO (X15150) DMTPIG (X57576) DMTRA2W (X57484)
DMTRFG (X70838) DMTSLG (Z30342) DMTU36B (X15008)
DMU03276 (U03276) DMU03986 (U03986) DMU04239 (U04239)
DMU04822 (U04822) DMU06861 (U06861) DMU07799 (U07799)
DMU11718 (U11718) DMU15928 (U15928) DMU18401 (U18401)
DMU19731 (U19731) DMU19742 (U19742) DMU20542 (U20542)
DMU20543 (U20543) DMU20566 (U20566) DMU21552 (U21552)
DMU24676 (U24676) DMU27181 (U27181) DMU28044 (U28044)
DMU33747 (U33747) DMU34039 (U34039) DMU35631 (U35631)
DMU35816 (U35816) DMU38951 (U38951) DMU39739 (U39739)
DMU43588 (U43588) DMU43737 (U43737) DMU43786 (U43786)

DMU46009 (U46009)
DMU51043 (U51043)
DMU51045 (U51045)
DMU51046 (U51046)
DMU51047 (U51047)
DMU51053 (U51053)
DMU52952 (U52952)
DMU56393 (U56393)
DMUROX (X51940)
DMW13 (X66270)
DMWHITE (X02974)
DMXDH (Y00308)
DMYELLOW (X04427)
DMYEMA (X63503)
DMYOLK (V00248)
DMYP3G (X04754)
DMZESTE (Y00049)
DROACT79B (M18829)
DROAFLL (M61127)
DROAPRTZ (L06280)
DROARF (L14923)
DROARF2A (L25062)
DROARF3B (L25064)
DROARRA (M30140)
DROBROWNPR (L05635)
DROBSHHB (L06475)
DROCDPR (L32839)
DROCOL4G (M96575)
DRODAPR (L23764)
DRODCDRK (D16402)
DRODEADA (M74824)
DRODFUR2X (L33831)
DRODGQ (M58016)
DRODHORO (L00964)
DRODMRBA (D37788)
DRODOXA2 (M63010)
DRODROSOPH (M23391)
DRODSOR1 (D13782)
DROECDINME (M97259)
DROEDG78A (M71247)
DROEDG84A (M71249)
DROEDG91A (M71250)
DROESCOMBS (L41867)
DROEST6A (J04167)
DROEVE (M14767)
DROFASI (M32311)
DROGAS02 (M23094)
DROGLDGMC (M29298)
DROGLTFAC (L17340)
DROHP1 (M57574)
DROIMPDEH (L14847)
DROLAMAA (M96388)
DROLAMB2A (M58417)
DROMDR50A (L07065)
DROMEX1A (M63626)
DROMNSO (L34276)
DROMSP316 (M32022)
DROMYLA (M11947)
DRONANOS (M72421)
DRONINAA (M22851)
DRONOD (M94188)
DROOPSA (K02315)
DROOPSAA (M12896)
DROOSKAR (M65178)
DROOTUA (M30825)
DROP40A (M90422)
DROPCNA (M33950)
DROPCXGEN (M74329)
DROPFK (L27653)
DROPGD (M80598)
DROPGLY (L27654)
DROPOLA (D90310)
DROPOLYABA (L13037)
DROPPP (M32141)
DROPRD (M14548)
DRORBP1A (L04929)
DRORNAHEL (L06018)
DROROUGH (M23629)
DRORPRIIA (M27431)
DRORPS17 (M22142)
DROSEV (J03158)
DROSNF (L29521)
DROSO7LESA (M77501)
DROSSL (L49382)
DROSUSG (M57889)
DROTRP (M34394)
DROTUBA1 (M14643)
DROTUBA4 (M14646)
DROVERM (M34147)
DROVITB (M18281)
DROXPACDR (D31892)
S57693 (S57693)
S66801 (S66801)
SMCECCG (Z11167)
U00145 (U00145)
U00683 (U00683)
U00790 (U00790)

APPENDIX D

Test set of 41 nonredundant maize GenBank sequences constructed by V. Brendel.

Format: LOCUS (ACCESSION)

MZEACT1G (J01238)	MZECAT3GN (L05934)	MZECDPKX (L27484)
MZECPN60A (L21007)	MZEMYBAA (M37153)	MZEOMTH (M73235)
MZETNENSPM (M25427)	MZETRNMU (M76978)	MZETRNMU (M76978)
MZEWISHR2 (M81603)	MZKN1GENE (X00000)	ZMA1G (X05068)
ZMADH1FA (X04050)	ZMALDOAR (X12872)	ZMALPTUB (X15704)
ZMANTG1 (X15711)	ZMAUX311 (X56737)	ZMB1TUB (X52878)
ZMBZMCC (X07940)	ZMC2CS (X60205)	ZMCATA1 (X60135)
ZMCP71C1G (X81828)	ZMFNRBP (Z26824)	ZMGLB1LG (X59083)
ZMGPA1 (X15408)	ZMGPC1 (X15596)	ZMGRP (X12564)
ZMLACDEHG (Z11754)	ZMOPA2 (X15544)	ZMPDCMRNA (X59546)
ZMPEP (X15642)	ZMR28DNA (X59138)	ZMRAB17G (X15994)
ZMRBCS (Y00322)	ZMSUCS2 (X02382)	ZMTRPA (X76713)
ZMU09989 (U09989)	ZMU14599 (U14599)	ZMU20450 (U20450)
ZMWAXY (X03935)	ZMZMCIIGE (X87126)	

REFERENCES

Altschul, S. F., Gish, W., Myers, E. W. & Lipman, D. J. (1990) Basic local alignment search tool. *J. Mol. Biol.* **215,** 403–410.

Ansari-Lari, M. A., Muzny, D. M., Lu, J., Lu, F., Lilley, C. E., Spanos, S., Malley, T. & Gibbs, R. A. (1996) A gene-rich cluster between the CD4 and triosephosphate isomerase genes at human chromosome 12p13. *Genome Res.* **6,** 314–326.

Arques, D. G. & Michel, C. J. (1990) Periodicities in coding and noncoding regions of the genes. *J. Theor. Biol.* **143,** 307–318.

Baum, L. E. & Petrie, T. (1966) Statistical inference for probabilistic functions of finite state Markov chains. *Ann. Math. Stat.* **37,** 1554–1563.

Bernardi, G. (1989) The isochore organization of the human genome. *Annu. Rev. Genet.* **23,** 637–661.

Bernardi, G., Olofsson, B., Filipski, J., Zerial, M. Salinas, J., Cuny, G., Meunier-Rotival, M. & Rodier, F. (1985) The mosaic genome of warm-blooded vertebrates. *Science* **228,** 953–958.

Black, D. L. (1991) Does steric interference between splice sites block the splicing of a short c-src neuron-specific exon in non-neuronal cells? *Genes & Dev.* **5,** 389–402.

Boguski, M. S. (1995) The turning point in genome research. *Trends Biochem. Sci.* **20,** 295–296.

Borodovsky, M. & McIninch, J. (1993) GENMARK: parallel gene recognition for both DNA strands. *Comp. Chem.* **17,** 123–133.

Breathnach, R. & Chambon, P. (1981) Organization and expression of eucaryotic split genes coding for proteins. *Annu. Rev. Biochem.* **50,** 349–383.

Brendel, V. (1992) PROSET - a fast procedure to create non-redundant sets of protein sequences. *Math. Comp. Modeling* **16 (6/7),** 37–43.

Brunak, S., Engelbrecht, J. & Knudsen, S. (1991) Prediction of human mRNA donor and acceptor sites from the DNA sequence. *J. Mol. Biol.* **220,** 49–65.

Bucher, P. (1990) Weight matrix descriptions of four eukaryotic RNA polymerase II promoter elements derived from 502 unrelated promoter sequences. *J. Mol. Biol.* **212,** 563-578.

Burge, C. & Karlin, S. (1997) Prediction of complete gene structures in human genomic DNA. *J. Mol. Biol.* in press.

Burset, M. & Guigó, R. (1996) Evaluation of gene structure prediction programs. *Genomics* **34,** 353–367.

Cavalier-Smith, T. (1985) Eukaryotic gene numbers, non-coding DNA and genome size. In *The Evolution of Genome Size*, Cavalier-Smith, T. ed., Wiley, London, pp. 69–103.

Chen, I. T. & Chasin, L. A. (1994) Large exon size does not limit splicing in vivo. *Mol. Cell. Biol.* **14,** 2140–2146.

Claverie, J. M. & Bougueleret, L. (1986) Heuristic informational analysis of sequences. *Nucl. Acids Res.* **14,** 179–196.

Cuny, G., Soriano, P., Macaya, G. & Bernardi, G. (1981) The major components of the mouse and human genomes. 1. Preparation, basic properties and compositional heterogeneity. *Eur. J. Biochem.* **115,** 227–233.

Dominski, Z. & Kole, R. (1991) Selection of splice sites in pre-mRNAs with short internal exons. *Mol. Cell. Biol.* **11,** 6075–6083.

Dong, S. & Searls, D. B. (1994) Gene structure prediction by linguistic methods. *Genomics* **23,** 540–551.

Dorit, R. L. & Gilbert, W. (1991) The limited universe of exons. *Curr. Opin. Genet. Dev.* **1,** 464–469.

Duret, L., Mouchiroud, D. & Gautier, C. (1995) Statistical analysis of vertebrate sequences reveals that long genes are scarce in CG-rich isochores. *J. Mol. Evol.* **40,** 308–317.

Fedorov, A., Suboch, G. Bujakov, M. & Fedorova, L. (1992) Analysis of nonuniformity in intron phase distribution. *Nucl. Acids Res.* **20,** 2553–2557.

Feller, W. (1950) *Probability Theory and its Applications, Vol. I,* John Wiley & Sons, New York, pp. 371–375.

Fickett, J. W. (1982) Recognition of protein coding regions in DNA sequences. *Nucl. Acids Res.* **10,** 5503–5518.

Fickett, J. W. (1996) Finding genes by computer: the state of the art. *Trends Genet.* **12(8),** 316–320.

Fickett, J. W. & Tung, C.-S. (1992) Assessment of protein coding measures. *Nucl. Acids Res.* **20,** 6441–6450.

Fields, C., Adams, M. D., White, O. & Venter, J. C. (1994) How many genes in the human genome? *Nat. Genet.* **7,** 345–346.

Fields, C. A. & Soderlund (1990) gm: A practical tool for automating DNA sequence analysis. *Comp. Appl. Biosci.* **6,** 263-270.

Forney, G. D. (1973) The Viterbi algorithm. *Proc. IEEE* **61,** 268–278.

Freedman, D. (1983) *Markov Chains*, Springer-Verlag, New York.

Fu, Y.-H., Friedman, D. L., Richards, S., Pearlman, J. A., Gibbs, R. A., Pizzuti, A., Ashizawa, T., Perryman, M. B., Scarlato, G., Fenwick, R. G., Jr. & Caskey, C. T. (1993) Decreased expression of myotonin-protein kinase messenger RNA and protein in adult form of myotonic dystrophy. *Science* **260,** 235–238.

Gardiner, K. (1996) Base composition and gene distribution: critical patterns in mammalian genome organization. *Trends Genet.* **12,** 519–524.

Gelfand, M. S. (1990) Computer prediction of exon-intron structure of mammalian pre-mRNAs. *Nucl. Acids Res.* **18,** 5865–5869.

Gelfand, M. S. (1995) Prediction of function in DNA sequence analysis. *J. Comp. Biol.* **2(1),** 87–115.

Gelfand, M. S. & Roytberg, M. A. (1993) Prediction of the intron-exon structure by a dynamic programming approach. *BioSystems* **30,** 173–182.

Gelfand, M. S., Mironov, A. A. & Pevzner, P. (1996) Gene recognition via spliced alignment. *Proc. Natl. Acad. Sci. USA* **93,** 9061–9066.

Gish, W. & States, D. J. (1993) Identification of protein coding regions by database similarity search. *Nat. Genet.* **3,** 266-272.

Green, M. R. (1991) Biochemical mechanisms of constitutive and regulated pre-mRNA splicing. *Annu. Rev. Cell Biol.* **7,** 559-599.

Guigó, R., Knudsen, S., Drake, N. & Smith, T. (1992) Prediction of gene structure. *J. Mol. Biol.* **226,** 141–157.

Harris, N. L. & Senapathy, P. (1990) Distribution and consensus of branch point signals in eukaryotic genes: a computerized statistical analysis. *Nucl. Acids Res.* **18,** 3015–3019.

Hawkins, J. D. (1988) A survey on intron and exon lengths. *Nucl. Acids Res.* **16,** 9893–9908.

Heinrichs, V., Bach, M., Winkelmann, G. & Luhrmann, R. (1990) U1-specific protein C needed for efficient complex formation of U1 snRNP with a 5′ splice site. *Science* **247,** 69–72.

Howard, R. A. (1971a) *Dynamic Probabilistic Systems Vol. I: Markov Models*, John Wiley & Sons, New York.

Howard, R. A. (1971b) *Dynamic Probabilistic Systems Vol. II: Semi-Markov and Decision Processes*, John Wiley & Sons, New York.

Hutchinson, G. B. & Hayden, M. R. (1992) The prediction of exons through an analysis of spliceable open reading frames. *Nucl. Acids Res.* **20,** 3453–3462.

Jurka, J. & Smith, T. (1988) A fundamental division in the ALU family of repeated sequences. *Proc. Natl. Acad. Sci. USA* **85,** 4775–4778.

Jurka, J., Klonowski, P., Dagman, V., Pelton, P. (1996) CENSOR - a program for identification and elimination of repetitive elements from DNA sequences. *Comp. Chem.* **20(1):** 119-122.

Karlin, S. & Burge, C. (1995) Dinucleotide relative abundance extremes: a genomic signature. *Trends Genet.* **11,** 283–290.

Karlin, S. & McGregor, J. (1958) Linear growth, birth and death processes. *J. Math. and Mech.* **7,** 643–662.

Karlin, S. & McGregor, J. (1959) Random walks. *Ill. J. Math.* **3,** 66–81.

Karlin, S. & Taylor, H. M. (1975) *A first course in stochastic processes.* Academic Press Inc., San Diego, CA.

Kernighan, B. W. & Richie, D. M. (1988) *The C programming language, 2nd ed.* Prentice Hall, Englewood Cliffs, NJ.

Konarska, M. M. & Sharp, P. A. (1987) Interactions between small nuclear ribonucleoprotein particles in formation of spliceosomes. *Cell* **49,** 763–774.

Konopka, A. K. & Owens, J. (1990) Complexity charts can be used to map functional domains in DNA. *Genet. Anal. Tech. Appl.* **7,** 35–38.

Krogh, A., Mian, I. S. & Haussler, D. (1994) A hidden Markov model that finds genes in *E. coli* DNA. *Nucl. Acids Res.* **22,** 4768–4778.

Kulp, D., Haussler, D., Reese, M.G. & Eeckman, F. H. (1996) A generalized hidden Markov model for the recognition of human genes in DNA. In *Proceedings of the Fourth International Conference on Intelligent Systems for Molecular Biology*, AAAI Press, Menlo Park, CA.

Levinson, S. E. (1986) Continuosly variable duration hidden Markov models for automatic speech recognition. *Comp. Speech and Lang.* **1,** 29–45.

Long, M. Y., Desouza, S. L. & Gilbert, W. (1995) Intron phase correlations and the evolution of the intron-exon structure of genes. *Proc. Natl. Acad. Sci. USA* **92,** 12495–12499.

Lopez, R., Larsen, F. & Prydz, H. (1994) Evaluation of the exon predictions of the GRAIL software. *Genomics* **24,** 133–136.

Maquat, L. E. (1995) When cells stop making sense: effects of nonsense codons on RNA metabolism in vertebrate cells. *RNA* **1,** 453–465.

McKeown, M. (1993) The role of small nuclear RNAs in RNA splicing. *Curr. Opin. Cell Biol.* **5,** 448–454.

McKeown, M. (1992) Alternative mRNA splicing. *Annu. Rev. Cell Biol.* **8,** 133–155.

Michel, C. J. (1986) New statistical approach to discriminate between protein coding and non-coding regions in DNA sequences and its evaluation. *J. Theor. Biol.* **120,** 223-236.

Moore, M. J., Query, C. C. & Sharp, P. A. (1993) Splicing of precursors to mRNAs by the spliceosome. In *RNA World*, eds. Gesteland, R. F. & Atkins, J. F., Cold Spring Harbor Lab. Press, Plainview, NY, pp. 305–358.

Mouchiroud, D., D'Onofrio, G., Aïssani, B., Macaya, G., Gautier, C. & Bernardi, G. (1991) The distribution of genes in the human genome. *Gene* **100,** 181–187.

Mount, S. M. (1982) A catalogue of splice junction sequences. *Nucl. Acids Res.* **10,** 459–472.

Prestridge, D. S. (1995) Predicting Pol II promoter sequences using transcription factor binding sites. *J. Mol. Biol.* **249,** 923–932.

Query, C. C., Moore, M. J. & Sharp, P. A. (1994) Branch nucleophile selection in pre-mRNA splicing: evidence for the bulged duplex model. *Genes Dev.* **8,** 587–597.

Rabiner, L. (1989) A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE.* **77(2),** 257–285.

Randall, L. L. & Hardy, S. J. S. (1989) Unity in function in the absence of consensus in sequence: role of leader peptides in export. *Science* **243,** 1156–1159.

Reed, R. (1989) The organization of 3′ splice-site sequences in mammalian introns. *Genes Dev.* **3,** 2113–2123.

Robberson, B. L., Cote, G. J. & Berget, S. M. (1990) Exon definition may facilitate splice site selection in RNAs with multiple exons. *Mol. Cell. Biol.* **10,** 84–94.

Sankoff, D. (1992) Efficient optimal decomposition of a sequence into disjoint regions, each matched to some template in an inventory. *Math. Biosci.* **111,** 279–293.

Schlotterer, C. & Tautz, D. (1992) Slippage synthesis of simple sequences. *Nucl. Acids Res.* **20,** 211–215.

Senapathy, P., Shapiro, M. B. & Harris, N. L. (1990) Splice junctions, branch point sites, and exons: sequence statistics, identification, and applications to genome project. *Meth. Enzymol.* **183,** 252–278.

Shepherd, J. C. W. (1981) Method to determine the reading frame of a protein from the purine/pyrimidine genome sequence and ist possible evolutionary justification. *Proc. Natl. Acad. Sci. USA* **78,** 1596–1600.

Siliciano, P. G. & Guthrie, C. (1988) 5′ splice site selection in yeast: genetic alterations in base-pairing with U1 reveal additional requirements. *Genes Dev.* **2,** 1258–1267.

Silverman, B. D. & Linsker, R. (1986) A measure of DNA periodicity. *J. Theor. Biol.* **118,** 295–300.

Smith, M. W. (1988) Structure of vertebrate genes: a statistical analysis implicating selection. *J. Mol. Evol.* **27,** 45–55.

Smith, C. W., Porro, E. G., Patton, J. G. & Nadal-Ginard, B. (1989) Scanning from an independently specified branch point defines 3′ splice site of mammalian introns. *Nature* **342,** 243–247.

Snyder, E. E. & Stormo, G. D. (1993) Identification of coding regions in genomic DNA sequences: an application of dynamic programming and neural networks. *Nucl. Acids Res.* **21,** 607–613.

Snyder, E. E. & Stormo, G. D. (1995) Identification of protein coding regions in genomic DNA. *J. Mol. Biol.* **248,** 1–18.

Solovyev, V. V., Salamov, A. A. & Lawrence, C. B. (1994) Predicting internal exons by oligonucleotide composition and discriminant analysis of spliceable open reading frames. *Nucl. Acid. Res.* **22,** 5156–5163.

Staden, R. & McLachlan, A. D. (1982) Codon preference and its use in identifying protein coding regions in long DNA sequences. *Nucl. Acids Res.* **10,** 141-156.

Staden, R. (1984) Computer methods to locate signals in nucleic acid sequences. *Nucl. Acids Res.* **12,** 505–519.

Sterner, D. A., Carlo, T. & Berget, S. M. (1996) Architectural limits on split genes *Proc. Natl. Acad. Sci. USA* **93,** 15081–15085.

Stormo, G. D. & Haussler, D. (1994) Optimally parsing a sequence into different classes based on multiple types of evidence. In *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pp. 47–55, AAAI Press, Menlo Park, CA.

Stormo, G. D., Schneider, T. D., Gold, L. & Ehrenfeucht, A. (1982) Use of the 'Perceptron' algorithm to distinguish translational initiation sites in E. coli. *Nucl. Acids Res.* **10,** 2997–3011.

Thomas, A. & Skolnick, M. H. (1994) A probabilistic model for detecting coding regions in DNA sequences. *IMA J. Math. Appl. Med. Biol.* **11,** 149–160.

Tomita, M., Shimizu, N. & Brutlag, D. (1996) Introns and reading frames: correlation between splicing sites and their codon positions. *Mol. Biol. Evol.* **13,** 1219–1223.

Uberbacher, E. & Mural, J. (1991) Locating protein coding regions in human DNA sequences by a multiple sensor-neural network approach. *Proc. Natl. Acad. Sci. USA* **88,** 11261–11265.

Viterbi, A. J. (1967) Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Trans. Informat. Theory* **IT-13,** 260-269.

Watson, J. D. (1992) In *The Code of Codes*, eds. Kevles, J. D. & Hood, L., Harvard University Press, Cambridge, MA, pp. 164–173.

Wieringa, B., Hofer, E. & Weissmann, C. (1984) A minimal intron length but no specific internal sequence is required for splicing the large rabbit B-globin intron. *Cell* **37,** 915–925.

Wu, T. (1996) A segment-based dynamic programming algorithm for predicting gene structure. *J. Comp. Biol.* **3(3),** 375–394.

Xu, Y., Einstein, J. R., Mural, R. J., Shah, M. & Uberbacher, E. C. (1994a) An improved system for exon recognition and gene modeling in human DNA sequences. In *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pp. 376–384, AAAI Press, Menlo Park, CA.

Xu, Y., Mural, R. J., & Uberbacher, E. C. (1994b) Constructing gene models from accurately predicted exons: an application of dynamic programming. *Comp. Appl. Biosci.* **10,** 613–623.

Zerial, M., Salinas, J., Filikski, J. & Bernardi, G. (1986) Gene distribution and nucleotide sequence organization in the human genome. *Eur. J. Biochem.* **160,** 479–185.

Zhang, M. Q. & Marr, T. G. (1993) A weight array method for splicing signal analysis. *Comp. Appl. Biosci.* **9(5),** 499–509.

Zhuang, Y. & Weiner, A. M. (1990) The conserved dinucleotide AG of the 3′ splice site may be recognized twice during in vitro splicing of mammalian mRNA precursors. *Gene* **90,** 263–269.

Zuker, M. (1991) Suboptimal sequence alignment in molecular biology: alignment with error analysis. *J. Mol. Biol.* **221,** 403–420.