COMPUTATIONAL PREDICTION OF EUKARYOTIC PROTEIN-CODING GENES

Michael Q. Zhang

The human genome sequence is the book of our life. Buried in this large volume are our genes, which are scattered as small DNA fragments throughout the genome and comprise a small percentage of the total text. Finding these indistinct 'needles' in a vast genomic 'haystack' can be extremely challenging. In response to this challenge, computational prediction approaches have proliferated in recent years that predict the location and structure of genes. Here, I discuss these approaches and explain why they have become essential for the analyses of newly sequenced genomes.

Biology has entered the genomic era. The celebrated draft human genome is already one year old, and a publicly available draft of the mouse genome has recently been assembled (see links to the Ensembl mouse genome server and the University of Santa Cruz Genome Bioinformatics site). At the time of writing, whole-genome sequences for more than 800 organisms (bacteria, archaea and eukaryota, as well as many viruses and organelles) are either complete or being determined (see link to Entrez genome). Driven by this explosion of genome data, gene-finding programs have also proliferated, particularly those that are designed for specific organisms. However, the accuracy with which genes can be predicted is still far from satisfactory: although, at the nucleotide level, 80% of genes are accurately predicted, at the exon level only 45% are predicted, and at the whole-gene level only ~20%. This is why estimates of the number of genes in the human genome are still imprecise (ranging from 30,000 to 100,000 genes).

At present, the annotation of most human genes is based on cDNA sequence data. Systematic 'full-length' cDNA sequencing programs, such as those at the Mammalian Gene Collection (MGC) in the USA and at RIKEN (The Institute of Physical and Chemical Research) in Japan, are generating vitally important experimental data towards defining complete gene sets for the human and mouse genomes. Of the best-annotated genes in the REFSEQ database (~17,000), nearly half are from such large-scale cDNA sequencing projects. Given that expressed sequence tags (ESTs) are most often generated from highly expressed transcripts, *ab initio* gene-prediction approaches need to combine several sources of information, such as from comparisons of human and mouse sequences, to discover new genes or rare transcripts. It is clear that further improvements to gene prediction are much needed. Even if, one day, all human genes were determined experimentally, it would still be important to understand how the structures of genes are organized and defined, and how they can be recognized. The ability to predict a gene structure is both an intellectual and a practical challenge.

Because those interested in gene-prediction approaches come from both biological and computational backgrounds, this review has been written for a broad audience. It provides background information and a survey of the latest developments in gene-prediction programs. It also highlights the problems that face the gene-prediction field and discusses future research goals. I hope to stimulate the best minds in both camps, so that new and creative gene-prediction methods will be developed. Although the accuracy of gene prediction has been steadily improving, the basic algorithms that underlie the various approaches have changed little since 1997. Although there have been

REFSEQ
The NCBI Reference Sequence project (RefSeq) provides curated gene, mRNA and protein sequences that reflect current knowledge about a sequence and its function, and that are available in the GenBank and NCBI databases.

Watson School of Biological Sciences, Cold Spring Harbor Laboratory, 1 Bungtown Road, PO Box 100, Cold Spring Harbor, New York 11724, USA.
e-mail: mzhang@cshl.edu
doi:10.1038/nrg890

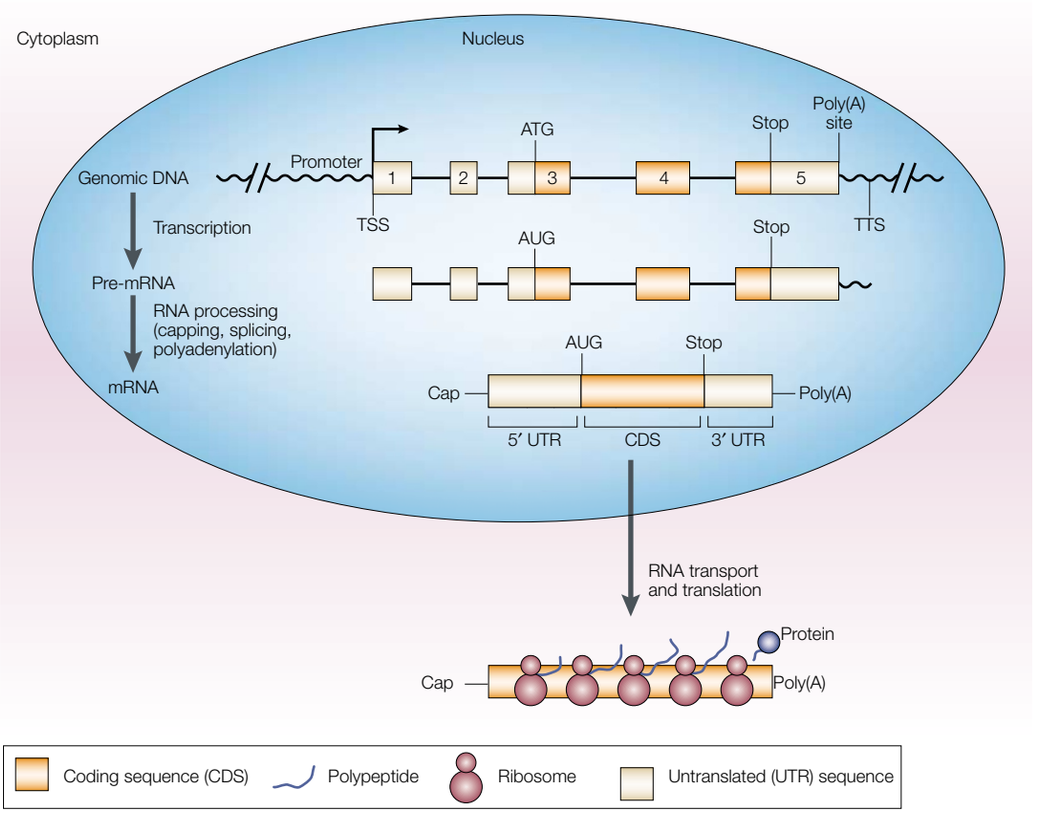

Figure 1 | **The central dogma of gene expression.** In the typical process of eukaryotic gene expression, a gene is transcribed from DNA to pre-mRNA. mRNA is then produced from pre-mRNA by RNA processing, which includes the capping, splicing and polyadenylation of the transcript. It is then transported from the nucleus to the cytoplasm for translation. TSS, transcription start site; TTS, transcription termination site.

many good reviews on this topic, and useful benchmarks in the research (for example, REFS 1–8), a truly fair comparison of the prediction programs is impossible as their performance depends crucially on the specific TRAINING DATA that are used to develop them.

TRAINING DATA SET
The known examples of an object (for example, an exon) that are used to train prediction algorithms, so that they learn the rules for predicting an object. They can be positive training sets (consisting of true objects, such as exons) or negative training sets (consisting of false objects, such as pseudoexons).

SPLICEOSOME
A ribonucleoprotein complex that is involved in splicing nuclear pre-mRNA. It is composed of five small nuclear ribonucleoproteins (snRNPs) and more than 50 non-snRNPs, which recognize and assemble on exon–intron boundaries to catalyse intron processing of the pre-mRNA.

Gene structure and exon classification

The main characteristic of a eukaryotic gene is the organization of its structure into exons and introns (FIG. 1). Generally, all exons can be separated into four classes: 5′ exons, internal exons, 3′ exons and intronless exons (or, simply, intronless genes) (FIG. 2). They can be further subdivided into 12 mutually exclusive subclasses, according to their coding content (FIG. 2a), and it has been shown that these subclasses have different statistical properties[9]. Because a vertebrate gene typically has many exons, internal coding exons (itexons, or internal translated exons) compose the main subclass that has been the focus of all gene-prediction programs. However, the definition of the term 'exon' has become confused, either unintentionally (due to lack of knowledge) or intentionally (for convenience). This confusion has led to the term 'exon' being used interchangeably with the term 'coding sequence' (CDS), which fails to take into account untranslated regions (UTRs). Almost all gene-prediction papers refer to four types of 'exon', as shown in FIG. 2b; however, these are just the coding regions of the exons. To avoid the misuse of these terms, I refer to subclasses of exons in this article as 5′ CDS, itexon, 3′ CDS and intronless CDS.

Finding internal coding exons

To determine exon–intron organization, an attempt can be made to detect either the introns or the exons. In early studies of pre-mRNA splicing, short splicing signals were identified in introns (FIG. 3): the donor site (5′ splice site or 5′ ss), which is characterized by the consensus AG|**GU**RAGU; the acceptor site (3′ ss), which is characterized by the consensus YYYYYYYYYNC**AG**|G; and the less-conserved branch site, which is characterized by CUR**A**Y[10]. These genetic elements direct the assembly of the SPLICEOSOME by base pairing with the RNA components of the splicing apparatus, which carries out the splicing reaction (FIG. 3). Where short introns, which are mostly found in lower eukaryotes (such as yeast), occur, the intron seems to be recognized molecularly by the interaction of the splicing factors, which bind to both ends of it. Such intron-based gene-structure prediction has also been used in some computer algorithms (for example, POMBE in REF. 11). Recently, however, Lim and

Burge[12], in a systematic analysis of short introns, have suggested that these standard splice sites might not be sufficient for defining introns in the genomes of plants and humans.

In vertebrates, the internal exons are small (~140 nucleotides on average), whereas introns are typically much larger (with some being more than 100 kb in length). In 1990, the 'exon-definition' model[13] was proposed to explain how the splicing machinery recognizes exons in a 'sea' of intronic DNA, where many cryptic splice sites exist. This model has since been validated by many experiments, and it proposes that an internal exon is initially recognized by the presence of a chain of interacting splicing factors that span it (FIG. 3). The binding of these *trans*-acting factors to the pre-mRNA is responsible for the non-random nucleotide patterns that form the molecular basis for all exon-recognition algorithms. These sequence features are often divided into two types: 'signals', which correspond to short *cis*-elements or boundary sites (such as splice sites and branch sites); and 'content', which corresponds to the extended functional regions (such as exons and introns). To evaluate each feature, one needs to define a scoring function of the feature (also called a feature variable). The best scoring function is the conditional probability $P(a|s)$ that the given sequence s contains the feature a. According to the Bayes equation $P(a|s) = P(s|a)P(a)/P(s)$ where $P(s|a)$ (that is, the likelihood P of s containing a). So, a training sample (sequence set) with the known feature a is built, and then the occurrence of a particular sequence s is counted. Different features can then be integrated into a single score for the whole object (an itexon in this case). Genes are predicted by finding the gene structure that has the highest score, given the sequence. Approaches differ in their choice of features, scoring functions and integration methods. Once the problem is phrased as a statistical-pattern recognition problem, many statistical or machine learning tools are available for recognizing these patterns. Indeed, almost all of them have been applied to the exon (or gene)-recognition problem. Here, I review just a few generic or popular approaches.

Most early programs used the simple positional weight matrix method (WMM, see BOX 1) to identify splice-site signals. In recent programs, the correlation among positions in a signal is also explored. The weight array method (WAM) or Markov models (BOX 1) are used to explore adjacent correlations; decision-tree or maximal-dependence decomposition (MDD) methods are used to explore non-adjacent correlations; and artificial neural network (ANN) methods are used to explore arbitrary, nonlinear dependencies. These more complex models typically yield significant, but not marked, improvements over the simple WMM. However, major improvements have come from designing programs that can combine many related sequence features. Such features can be combined at different levels. At the splice-site level, the simplest way of combining features (such as splice-site score with exon-content score on the one hand and with intron-content score on the other hand) is to use Fisher's linear discriminant analysis (LDA; BOX 1). In the LDA method, the total score is a linear sum of the scores of individual features, and the coefficients are determined by minimizing the prediction error using a positive and a negative training data set. This is equivalent to a perceptron method (for example, see REF. 14), which identifies an optimal plane surface to separate true positives from true negatives.

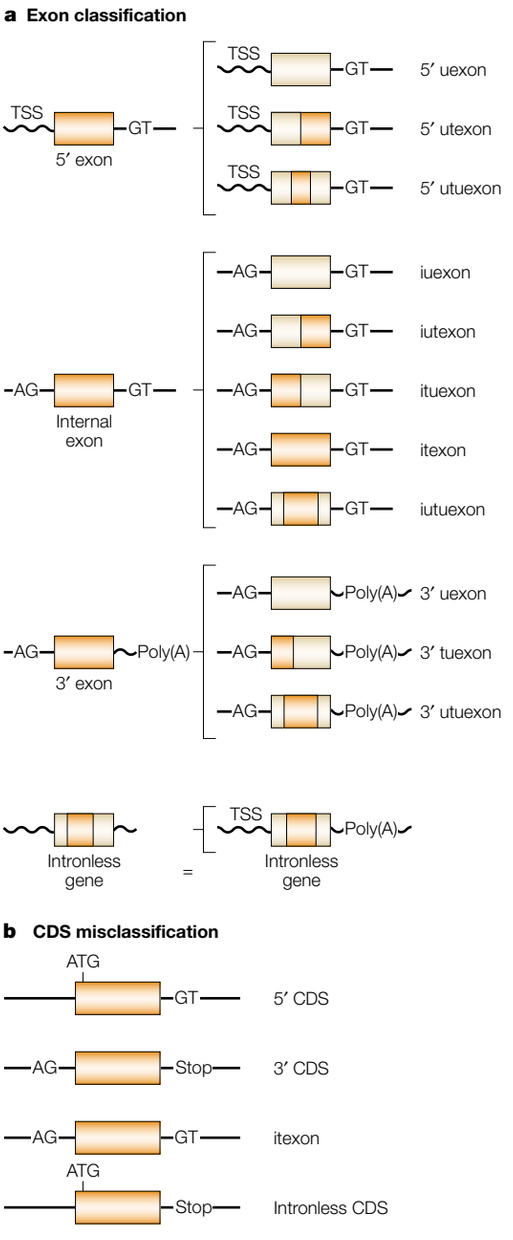

Figure 2 | **Exon classification. a** | Exons can be classified into four classes and 12 subclasses, as shown. **b** | Coding sequence (CDS) 'exons'. Four classes of exon-coding regions. These regions are not whole exons, except for the internal coding exons (itexons). i, internal; poly(A), polyadenylation; t, translated; TSS, transcription start site; u, untranslated .

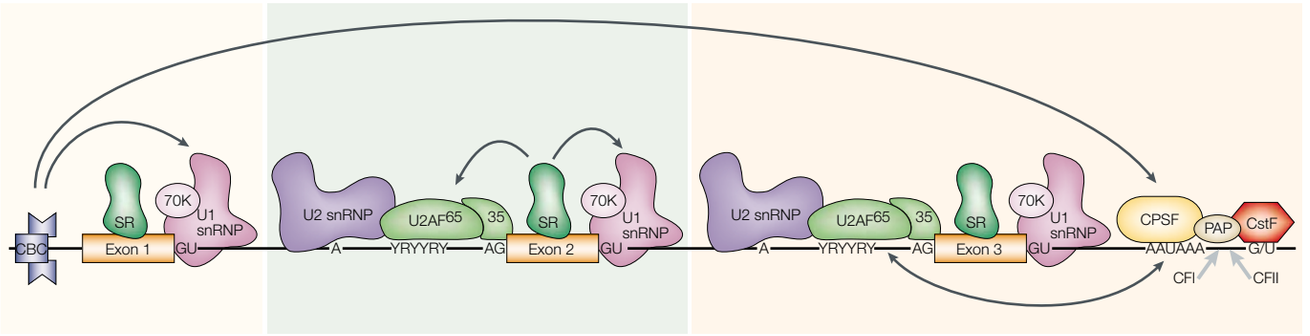

Figure 3 | **Exon-definition model.** Typically, in vertebrates, exons are much shorter than introns. According to the exon-definition model, before introns are recognized and spliced out, each exon is initially recognized by the protein factors that form a bridge across it. In this way, each exon, together with its flanking sequences, forms a molecular, as well as a computational, recognition module (arrows indicate molecular interactions). Modified with permission from REF. 26 © (2002) Macmillan Magazines Ltd. CBC, cap-binding complex; CFI/II, cleavage factor I/II; CPSF, cleavage and polyadenylation specificity factor; CstF, the cleavage stimulation factor; PAP, poly(A) polymerase; snRNP, small nuclear RNP; SR, SR protein; U2AF, U2 small nuclear ribonucleoprotein particle (snRNP) auxiliary factor.

LDA is implemented in SPL — a splice-site recognition module of the HEXON program[15]. A new splice-site detection program, GeneSplicer, has also been developed recently[16] and is reported to perform favourably when compared with many other programs (such as NetPlantGene, NetGene2, HSPL, NNSplice, GENIO and SpliceView; BOX 2).

To discriminate CDS from intervening sequence, the best content measures are the so-called frame-specific hexamer frequencies (BOX 1), because they capture codon-bias information and codon–codon correlations. They also capture splice-site preferences, which are the most characteristic exon–intron features[17]. For long open reading frames (ORFs), such as in bacterial or intronless genes, frame-specific hexamer frequencies alone can detect most of the CDS regions. An alternative approach[18] is to use an interpolated Markov model (IMM), in which the higher-order Markov probabilities are estimated from an average of the lower-order ones. Because the G+C content of mammalian genomes is biased by ISOCHORES (for example, see REF. 19), all content and signal measures need to be computed separately for different G+C regions. Exon size is another important feature variable because, for example, itexons have an approximately LOG-NORMAL DISTRIBUTION[9].

By combining splice-site features with exon–intron features (such as CDS measures, exon size and others), and by using a nonlinear quadratic discriminant analysis (QDA), the itexon-prediction program MZEF[20] has done better at the single-exon level than has HEXON (which is based on a LDA method) or GRAIL2 (which is based on an ANN method[21]). However, to further improve exon-prediction accuracy, exon–exon dependencies also have to be incorporated, as discussed below.

Finding poly(A) sites and 3′ exons

The correct identification of the boundaries of a gene is essential when searching for several genes in a large genomic region. Many gene-prediction programs fail to identify these boundaries, which results in predicted genes being either truncated or fused together. Determining the 3′ end of a gene is easier than determining its 5′ end. This is because most of the mRNA and EST sequences in GenBank are truncated at their 5′ ends. The exon-definition model can also be applied to 3′ exons by replacing the 5′ ss with the poly(A) site and by using the 3′-EXON LENGTH DISTRIBUTION — this is because long internal exons are rare in vertebrates, whereas 3′ exons frequently extend for many kilobases. The molecular bridge in this case is the interaction between the splicing factor U2AF65 and the carboxy-terminal domain of the poly(A) polymerase, which recognizes the poly(A) signal (FIG. 3).

By aligning 3′ ESTs against genomic sequence, many poly(A) sites have been identified. In this way, several statistical features (including the well-known poly(A) signal AAUAAA and the (G+U)-rich site) have been identified in six species (yeast, rice, *Arabidopsis*, fly, mouse and human) and used for poly(A)-site recognition[22]. More reliable 3′ ends have been obtained by aligning mRNAs with genomic sequences. By using such a training set, a QDA-based program called POLYADQ was developed[23], which can predict both AAUAAA- and AUUAAA-dependent poly(A) sites in the human genome.

Because almost all gene-prediction programs focus on coding regions, they can only identify the 3′ CDS instead of the real 3′ exon. However, any itexon-recognition methods can be modified for this task by replacing the donor-site signal with the STOP-codon signal (FIG. 2b), together with the correct exon length distribution.

A true 3′-exon-prediction program, JTEF[24] (BOX 2), was developed recently using a QDA-based method, which can predict the major subtype of 3′ exons — the 3′ tuexons (translated-then-untranslated 3′ exons, which are those that contain the true STOP codon, see FIG. 2a). Because it integrates several features across the 3′ exon, JTEF has substantially improved the accuracy of

ISOCHORE
A large region of mammalian genomic DNA sequence in which C+G compositions are relatively uniform.

LOG-NORMAL DISTRIBUTION
The distribution of a random variable, the logarithm of which follows a normal distribution. A normal log (length) implies a strong fixed-length selection pressure.

EXON LENGTH DISTRIBUTION
A statistical distribution of exon sizes.

NONSENSE-MEDIATED DECAY (NMD). A pathway ensuring that mRNAs that have premature stop codons are eliminated as templates for translation.

PSEUDOEXON
A pre-mRNA sequence that resembles an exon, both in its size and in the presence of flanking splice-site sequences, but that is never recognized as an exon by the splicing machinery (the spliceosome).

poly(A)-site prediction in comparison with that by either the poly(A)-site-specific program POLYADQ or the more sophisticated mutiple-gene prediction programs (such as Genscan and GeneMark). At present, no prediction program is available for the minor subtype of 3′ exons — the 3′ uexons (untranslated 3′ exons). Developing the 3′-uexon prediction program will make an important contribution to the gene-finding field. However, this will be difficult to achieve, as some of the annotated introns in 3′ UTRs might be annotation errors, especially in the light of recent results, which indicate that the presence of a stop codon before the last intron often leads to the degradation of a transcript by NONSENSE-MEDIATED DECAY (see recent reviews in REFS 25,26).

Finding promoters and 5′ exons

Identifying the 5′ end of a gene is one of the most difficult tasks in gene finding. This is mainly due to the difficulty of identifying the promoter and the transcriptional start site (TSS) sequences. At present, of the ~17,000 human RefSeq genes that are in GenBank, only ~3,000 of them are annotated for the TSS. Most of the

Box 1 | **Gene-prediction terms and concepts**

Linear discriminant analysis and quadratic discriminant analysis
Two classical, statistical pattern-recognition methods that are used to categorize samples into two classes. Once samples have been represented as points in space, linear discriminant analysis (LDA) finds an optimal plane surface that best separates points that belong to two classes. Quadratic discriminant analysis (QDA) finds an optimal curved (quadratic) surface instead. For example, if there are ten true exons and ten PSEUDOEXONS, and two feature variables — 5′ splice-site (ss) score and 3′-ss score — these samples could be represented by 20 points in a two-dimensional space (the 5′-ss score on the *x* axis and the 3′-ss score on the *y* axis). LDA (or QDA) would compute a straight (or curved) line through the space that can best separate the two classes of exons (with the minimal classification error).

Perceptron method
A machine learning algorithm for pattern recognition or classification. Unlike LDA-based approaches, which calculate theoretically the final best-discriminant plane, a perceptron method is based on a simple neural network that begins with an arbitrary initial plane and then iteratively moves the plane in a way that tries to reduce the classification error at each step.

Hidden Markov models
Probability models that were first developed in the speech-recognition field and later applied to protein- and DNA-sequence pattern recognition. Hidden Markov models (HMMs) represent a system as a set of discrete states and as transitions between those states. Each transition has an associated probability. Markov models are 'hidden' when one or more of the states cannot be observed directly. HMMs are valuable in bioinformatics because they allow a search or alignment algorithm to be built on firm probability bases, and it is straightforward to train the parameters (transition probabilities) with known data.

Hexamer-coding measures
Some methods interpret sequences as successions of 'words' — so-called because nucleotides are not independent of each other, but tend to occur together as if in a word — of length *k* (*k*-tuples); 6-tuples are called hexamers. In-frame hexamer frequencies in a region of DNA have traditionally been used as a powerful way of discriminating coding regions from non-coding regions, as some 'words' are more likely to be present in either type of DNA. A score *s* for a hexamer *w*, such as CAGCAG, can be defined as $s(w) = \log(\text{freq}(w))$. Because the frequency of CAGCAG is relatively high in exons, its score in exons will be higher than that of, for example, TAATAA.

Weight matrix method and weight array method
Used for scoring a signal motif site. In the weight matrix method (WMM), a score $s(x,b)$ is assigned to each position *x* for each base pair *b*, such that the total score of a motif site can be calculated as the sum of scores at all positions in the site. In the weight array method (WAM), a score $s(x,w)$ is assigned to each position *x* for each word *w* of length *k* (when $k = 1$, the two methods are the same).

Maximal-dependence decomposition (MDD) donor matrices
A set of donor splice-site weight matrices that are generated using the WMM, each of which is built for a different class of splicing donor sites in such a way that the dependence between nucleotide positions is minimized.

Decision tree
A classification scheme, which can be used, for example, to split a sample into two subsamples according to some rule (feature variable threshold). Each subsample can be further split, and so on.

Artificial neural networks
A collection of mathematical models that emulate some of the observed properties of biological nervous systems and draw on the analogies of adaptive biological learning. The key element of the artificial neural network (ANN) model is the novel structure of the information processing system. It is composed of many highly interconnected processing elements that are analogous to neurons and are tied together with weighted connections that are analogous to synapses. Once it is trained on known exon or intron sample sequences, it will be able to predict exons or introns in a query sequence automatically.

cDNA-derived mRNA sequences in GenBank are truncated at the 5′ end because of the falling-off of the reverse transcriptase during cDNA production. However, a recently reported new Database of Transcriptional Start Sites (DBTSS) contains the 5′ ends of ~8,000 human genes[27]; this resource will be extremely useful for promoter studies.

Promoter activation and transcription initiation is a complex process[28]. After chromatin around the promoter has been remodelled into the hyperacetylated and relaxed state that is associated with transcriptionally active chromatin, the next step in transcription is the binding of the pre-initiation complex to the core promoter (which lies ~100 bp either side of the TSS). The initiation of transcription is controlled mainly by transcription factors that bind to the proximal region of the promoter (which lies ~1 kb upstream of the TSS) and to the first intron region.

There are many promoter- and TSS-prediction programs. In general, their performance is far from satisfactory, especially with respect to the control of false-positive predictions (see, for example, REFS 29–32). For low-resolution (~2-kb) mapping of TSS sequences that are related to CpG islands in large genomic regions, CpG_Promoter[33] can be used. However, for the high-resolution (~100-bp) mapping of a TSS in a 2-kb region, Core_Promoter[34] might be a better choice. For general-purpose genome-wide promoter scans, PromoterInspector[35] is reported to have achieved the true-positive-to-false-positive ratio of 2.3, compared with the then best ratio of 0.6 for the TSSW program[36] (BOX 1). A new program, Eponine[37], performs with similar sensitivity and specificity to PromoterInspector, and is able to predict the location of the TSS better by exploiting significant discriminating features (such as the TATA box and nearby CpG islands). Further specificity can be achieved for specific co-regulated groups of genes by exploring specific correlations among several transcription-factor-binding sites in a functional module[38,39].

As in the case of 3′-exon prediction, almost all gene-prediction programs can only predict the 5′ CDS (FIG. 2b). This has been done by modifying the approach to predicting itexons, by replacing the 3′-ss signals with the translational initiation signal ATG, using KOZAK rules (for example, see REF. 40), together with the correct exon length distribution.

Recently, a real 5′-exon prediction algorithm, FirstEF (based on QDA), was published[41]. It separates the CpG-related 5′ exons from the non-CpG-related ones, and uses first-intron-specific MDD donor matrices. It can predict both 5′ utexons and 5′ uexons. By integrating many sequence features, it has also improved on the accuracy of promoter and TSS predictions.

KOZAK SEQUENCE
The consensus sequence for initiation of translation in vertebrates.

PSEUDOGENE
A DNA sequence that was derived originally from a functional protein-coding gene that has lost its function, owing to the presence of one or more inactivating mutations.

Finding intronless CDSs and pseudogenes

Predicting intronless CDSs might seem to be easy, but this would only be true if most genes were intronless and if few PSEUDOGENES existed (as in bacterial genomes or the genome of *Saccharomyces cerevisiae*). For example, many *S. cerevisiae* genes are defined as ORFs of 300 bp or more because an average protein is long (~1,000 amino acids), and such a long ORF is rare unless it has been selected for coding[42]. Although the usual hexamer-coding measures, or even simpler (species-independent) periodicity or entropy types of coding measure, do well at predicting a large coding region, they can still confuse an intronless gene for a long, internal-coding exon. Many pseudogenes are spliced copies of wild-type genes and, unless they have accumulated nonsense mutations, it can be very difficult to distinguish pseudogenes from intronless CDSs without knowing about the wild-type gene or without ruling out that the nonsense mutation-bearing region might actually be an intron. To make such a distinction requires experience and caution[43]. As current gene-prediction programs are biased towards intron-containing genes, many intronless genes might have been missed by such programs. Many false-positive exon predictions have also been caused by pseudogenes. Developing better and more specialized algorithms to recognize them is becoming increasingly important.

Exon assembly and single-transcript prediction

Just as integrating splice-site signals with coding measures at the single-exon level can increase the accuracy of predicting individual splice sites, integrating various exons into full transcripts can also increase the accuracy with which individual exons can be predicted. The non-random nature of DNA is such that molecular interactions and functional selection have together created and maintained subtle and complex interdependencies among different parts of the structure of a gene (FIG. 3). If these interdependencies are not incorporated into a prediction model, the model will perform less accurately. Because the first and last exons of a gene are the most difficult to identify, most current assembly programs only focus on coding fragments, such as the 5′ CDS, defined by ATG–GT; the itexon, defined by AG–GT; the 3′ CDS, defined by AG–STOP; and the intronless CDS, defined by ATG–STOP (FIG. 2b). A few programs (such as Genscan) add two untranslated states: the '5′ UTR', defined by TSS–ATG, and the '3′ UTR', defined by a STOP–poly(A) site. However, it should be noted that these 'untranslated' fragments are defined on a pre-mRNA that might contain introns; the real UTRs are defined on a mature (spliced) mRNA (FIG. 2a).

Given all possible gene fragments and their scores, dynamic programming (DP) was originally used by many programs to assemble a best (highest score) combination of compatible parts into a full pre-mRNA transcript (for example, see REFS 44,45). When scores for different parts are not probabilities, appropriate weighting has to be considered before the scores might be combined. In Stormo and Haussler[46], a general method is provided for optimizing such weights.

More recently, fully probabilistic state models (HMMs; BOX 1) have become preferable because, in these models, all scores are probabilities themselves. The weighting problem has become a matter of counting relative observed state frequencies. In a HMM, a DNA sequence is partitioned into disjointed fragments or states (because of the duality of the regions and

Box 2 | **Useful internet resources**

Gene-prediction programs: comparative genomics

Doublescan	http://www.sanger.ac.uk/Software/analysis/doublescan
SLAM	http://bio.math.berkeley.edu/slam
Twinscan	http://genes.cs.wustl.edu

Gene-prediction programs (many with homology searching capabilities)

GeneMachine	http://genome.nhgri.nih.gov/genemachine
Genscan	http://genes.mit.edu/GENSCAN.html
GenomeScan	http://genes.mit.edu/genomescan
Fgenesh, Fgenes-M, TSSW, TSSG, Polyah, SPL and RNASPL	http://genomic.sanger.ac.uk/gf/gf.shtml
Fgenesh, Fgenes-M, SPL and RNASPL	http://www.softberry.com/berry.phtml
HMMgene	http://www.cbs.dtu.dk/services/HMMgene
Genie	http://www.fruitfly.org/seq_tools/genie.html
GRAIL	http://compbio.ornl.gov/tools/index.shtml
GeneMark	http://www.ebi.ac.uk/genemark [OK?]
GeneID	http://www1.imim.es/software/geneid/geneid.html#top
GeneParser	http://beagle.colorado.edu/~eesnyder/GeneParser.html
MZEF and POMBE	http://argon.cshl.org/genefinder/ [OK?]
AAT, MZEF with homology	http://genome.cs.mtu.edu/aat.html
MZEF with SpliceProximalCheck	http://industry.ebi.ac.uk/~thanaraj/MZEF-SPC.html
Genesplicer, Glimmer and GlimmerM	http://www.tigr.org/~salzberg
WebGene	http://www.itba.mi.cnr.it/webgene
GenLang	http://www.cbil.upenn.edu/genlang/genlang_home.html
Xpound	ftp://igs-server.cnrs-mrs.fr/pub/Banbury/xpound

Gene-prediction programs: alignment based

Procrustes	http://www-hto.usc.edu/software/procrustes/index.html
GeneWise2	http://www.sanger.ac.uk/Software/Wise2
SplicePredictor	http://bioinformatics.iastate.edu/cgi-bin/sp.cgi
PredictGenes	http://cbrg.inf.ethz.ch/subsection3_1_8.html

Finding ORFs and splice sites

DioGenes	http://www.cbc.umn.edu/diogenes/index.html
OrfFinder	http://www.ncbi.nlm.nih.gov/gorf/gorf.html
YeastGene	http://tubic.tju.edu.cn/cgi-bin/Yeastgene.cgi
CDS: search coding regions	http://bioweb.pasteur.fr/seqanal/interfaces/cds-simple.html
Neural network splice site prediction	http://www.fruitfly.org/seq_tools/splice.html
NetGene2	http://www.cbs.dtu.dk/services/NetGene2

Last exon, promoter or TSS prediction

FirstEF, Core_Promoter, CpG_Promoter, Polyadq and JTEF	http://www.cshl.edu/mzhanglab
Eponine	http://www.sanger.ac.uk/Users/td2/eponine
Neural network promoter prediction	http://www.fruitfly.org/seq_tools/promoter.html
Transcription element search system	http://www.cbil.upenn.edu/tess
Signal Scan	http://bimas.dcrt.nih.gov/molbio/signal

AAT, analysis and annotation tool; ORF, open reading frame; TSS; transcription start site.

boundaries, we refer to a region as a state and to a boundary as a transition between states). If the conditional probability $P(s|q)$ of finding a base s in state q (which might depend on neighbouring bases as specified by the probability model) and the transition probability $T(q|q')$ of finding state q after state q', for any possible assignment (called a parse Φ) of states $\{q_i: i = 1,2,\ldots,N\}$ (i enumerates positions) are known, the joint probability is given by $P(\Phi, S) = P(s_1|q_1)T(q_1|q_2)P(s_2|q_2)\ldots T(q_{N-1}|q_N)P(s_N|q_N)P_0(q_N)$. The Viterbi algorithm (DP for a HMM) can be used to find the most probable parse Φ^* (REF. 47) that corresponds to the optimal transcript (exon or intron) prediction.

The advantage of HMMs is that more states (such as intergenic regions, promoters, UTRs, poly(A) and frame- or strand-dependent exons and introns) can be added, as well as flexible transitions between the states, to allow partial transcripts, intronless genes or even multiple genes to be incorporated into a model. Multiple transcript predictions (which might correspond to alternatively spliced transcripts) can also be obtained by using sub-optimal parses. Because many functional features that determine alternative splicing have not been incorporated into existing programs, sub-optimal parses (or assignments) are unlikely to represent alternative splicing events. Rather, they can serve as

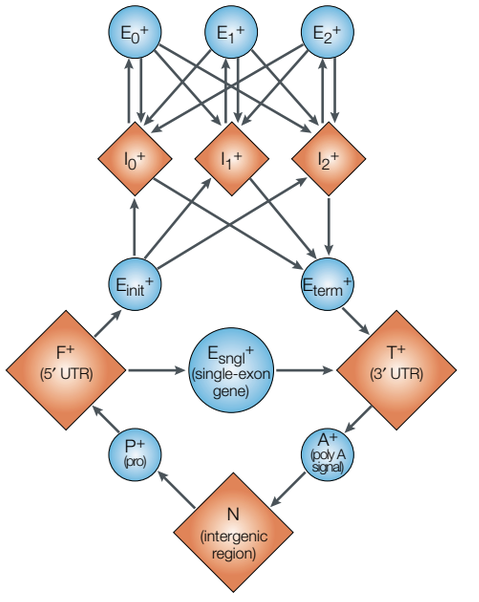

Figure 4 | **Different states and transitions in the Genscan hidden Markov model.** Genscan is a gene-prediction algorithm that, like other hidden Markov models (HMMs), models the transition probabilities from one part (state) of a gene to another. Here, each circle or square represents a functional unit (a state) of a gene on its forward strand (for example, E_{init} is the 5′ coding sequence (CDS) and E_{term} is the 3′ CDS, and the arrows represent the transition probability from one state to another. The Genscan algorithm is trained by pre-computing the transition probabilities from a set of known gene structures. Test sequence data can then be run one base position at a time, and the model will predict the optimal state for that position. The model for the reverse strand (beneath the dashed line) is in mirror symmetry to the model shown, with respect to the horizontal axis. Please note that these 'UTRs' (untranslated regions) might contain introns and so should not be confused with the standard UTR. E, exon; I, intron; pro, promoter. Modified with permission from REF. 2 © (1997) Elsevier Science.

a stability indicator: if many sub-optimal parses are very close (in terms of their probabilities) to the optimal one, the optimal prediction might not be very reliable.

Because HMMs are fully probabilistic, a score (conditional probability) can be obtained for any part of a gene. For example, the likelihood of finding an exon in a particular interval might be calculated by a 'forward' and a 'backward' algorithm[47]. Because of the interdependency of exons, the quality (probability score) of an exon also depends on other exons or even on the entire sequence. As a result, HMM-based exon-assembly methods explore exon–exon correlations and so predict exons more accurately than when predictions are based on single, isolated exons.

HMMgene[48] is based on HMM and can be optimized to predict exons to a high degree of accuracy. The Genie program was the first to introduce a generalized HMM[49] (GHMM; BOX 1) and used neural networks as individual sensors for splice signals, as well as for coding content. Like HMMgene and Genie, Genscan is also based on a GHMM[2]. It also allows exon-specific length distribution to be predicted (that is, the model generates blocks of base pairs — a whole exon and exons predicted to be of average length receive high probability scores). By contrast, the intrinsic length distribution for a standard HMM is geometric, which results in the exon score decaying exponentially with exon length. But, the splice-site sensors in Genscan are more advanced than those used in the other programs. Fgenesh[50], another GHMM-based algorithm, uses LDA (BOX 1) as the exon sensor. The coding-content sensors all use a fifth-order Markov chain (in this approach, the probability score for any base pair depends on the previous five base pairs; BOX 1). These GHMM programs also model promoters, poly(A) signals and the 5′ UTRs or 3′ UTRs (including possible introns) in a relatively simple way (FIG. 4).

Recently, another gene-prediction program, GRPL[51], has been developed. It is based on reference point logistic (RPL) regression, which is a generalization of logistic regression[52] that can be used in complex classification problems to model the conditional probability that an object belongs to a specified class given its observed features. In tests of this program, GRPL matches the performance of Genscan at the nucleotide level (with respect to the correct prediction of exons and introns), but does slightly worse than Genscan at the exon level. A more recent test of many programs (such as Fgenes, GeneMark, Genie, Genscan, HMMgene, Morgan and MZEF; BOX 2) on 195 newly sequenced DNAs showed that the accuracy of gene prediction (the average of sensitivity plus specificity) is ~70–90% at the nucleotide level and ~40–70% at the exon level[6]. In practice, combining the predictions of several programs can yield even greater accuracy[53].

Multiple genes, partial genes and both strands

It is easy to add more states or transitions between states to HMM-based models so that multiple genes, partial genes and genes on both strands can be predicted together. These features are essential when annotating genomes or large chunks of sequence data, such as large contigs, in an automated fashion. The technique of predicting multiple genes on both strands was initially implemented in Genscan[2], and was later adopted in other HMM-based algorithms, such as GeneMark[54] and Fgenesh[50]. The advantage of modelling both strands simultaneously is that it avoids the prediction of genes that overlap on the two strands as being two separate genes, which are presumed to be rare in mammalian genomes. More importantly, it makes the prediction of 'shadow exons' (exons that are predicted to be in the correct region but on the wrong DNA strand) much less likely. This can arise because coding-biased sequence composition can look distinct from intron or intergenic sequence to the predictor — the extent to which this effect occurs depends on the organism (see, for example, REF. 55). Most gene-prediction algorithms can achieve ~80% sensitivity and specificity at the exon level when tested on single-gene data sets[56], but these statistics drop

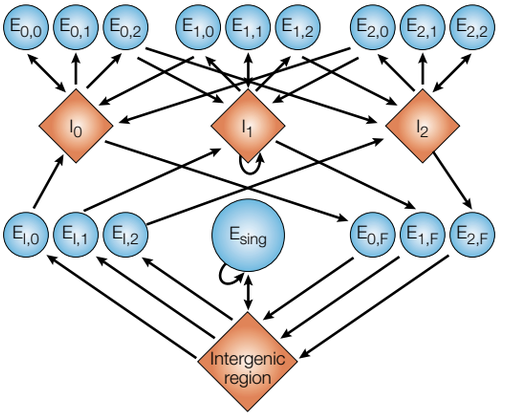

Figure 5 | **A generalized pair hidden Markov model.** A generalized pair hidden Markov model (GPHMM) for aligning and predicting exons using genomic DNA sequences from two related organisms. The main difference between this model and the usual HMM (such as Genscan) is that an exon pair (one from each organism) is generated according to some joint distribution. 'E' represents an exon-pair state. The internal exon pairs are shown at the top. E_{ij} indicates the state that can create an exon-pair connecting an upstream *i*-phase intron (I_i) and a downstream *j*-phase intron (I_j) (where *i*,*j* = 0, 1 or 2). E_{Ii} indicates the first exon state that can create an initial exon-pair that connects a downstream *i*-phase intron, and E_{iF} indicates the last exon-pair state that can create a final exon-pair that connect to an upstream *j*-phase intron. E_{sing} indicates a simple exon state (an intronless gene). Modified with permission from REF. 79 © (2002) Mary Ann Liebert, Inc.

to ~60% sensitivity and specificity when these programs are run on large-scale genomic DNA data sets[57].

By integrating features across several genes, a feature (such as an exon) in one gene becomes dependent on the features of other genes. And that is why, when a few starting or ending bases of the input sequence are deleted, it can change the overall prediction of gene structure. There is very little biological evidence for the existence of correlations among exons in different genes, except for genes at some tightly linked loci, such as the locus control region of the β-globin locus or where a pair of genes is controlled by a common promoter. A probability score only makes sense when: first, the underlying model is correct, and second, the training samples are not biased. As our knowledge about the dependencies between genes is very limited, multiple-gene models are unlikely to be accurate. As most algorithms cannot even predict the first and last exons, the splitting and fusing of genes occurs quite often, even with the best programs, when they are run on large genomic data sets. An accurate prediction of multiple genes will only be possible once we have a better understanding of the long-range features of chromosomes. These long-range features include insulator and boundary elements, and matrix- and scaffold-attachment regions[58], which all allow a chromosome to be broken up into its transcriptionally independent domains[59].

BLASTX
Basic local alignment tool (BLAST) is a computer program for comparing DNA and protein sequences. The BLASTX version compares a nucleotide query sequence that is translated in all reading frames with a protein sequence database.

Combining similarity scores

The use of database search-and-alignment programs, such as BLASTX[60] and Sim4 (REF. 61), in gene finding has been popular because matching a sequence to a known protein or cDNA/EST can greatly improve the accuracy of gene prediction. Traditionally, *ab initio* gene prediction and similarity searches are run independently, and a curator then combines the results manually for gene annotation. Many people have tried to integrate these methods automatically[62–67].

The 'splice alignment' program — Procrustes[62] — is based on the observation that the detection of exon boundaries in a gene can be improved if a close protein homologue for that gene exists. Similarly, the Ensembl automatic gene annotation engine — GeneWise[66] — combines a gene-prediction HMM with the protein-profile HMM (Pfam) to achieve simultaneous gene prediction and alignment. Although these methods can be highly accurate, they predict exactly one gene per genomic sequence, require close homologues to identify complete genes[68] and are computationally intensive, requiring a prescan with, for example, BLASTX to first identify candidate regions. To provide a first layer of annotation on the human draft, a new algorithm GenomeScan was developed recently[69], which combines exon or intron and splice-signal models with similarity to known protein sequences in an integrated model. Initial comparisons of GenomeScan with Procrustes and GeneWise seemed to favour GenomeScan[69], because Procrustes and GeneWise both predict partial genes, which results in the terminal exons being frequently truncated. However, if the prediction of internal exons (or splice sites) is considered, the performance of Procrustes and GeneWise is comparable with that of GenomeScan. Because the quality of an EST-derived sequence is generally very poor, it must be very carefully combined with any automatic gene-prediction algorithms[70]. In general, similarity searches can boost the accuracy of gene prediction by a few per cent. For example, GRPL+ is the similarity-enhanced version of GRPL[51], and has shown a 5% increase in prediction accuracy over GRPL.

Comparative genomics methods

The value of comparative genomics is illustrated by the sequencing of the mouse genome for the purpose of annotating the human genome. The availability of closely related genomes makes it possible to carry out genome-wise comparisons and analyses of synteny. When two genomes have only recently diverged, the order of many genes, gene numbers, gene positions and even gene structures (exon–intron organization, splice site usage, and so on) remain highly conserved. New genes can also be identified from direct genome comparisons. By comparing the genomes of several closely related species, conserved regulatory regions can also be easily identified[71]. For these reasons, making use of comparative genomic data will be a key challenge for the gene-prediction field.

Box 3 | **Future challenges for the gene-prediction field**

- To create better algorithms for identifying general, as well as tissue- or developmental-specific, classes of promoters.
- To achieve a greater understanding of CpG islands and methylation patterns.
- To have a better characterization of the splicing enhancers and silencers that mediate alternative splicing, to allow models to predict alternative exons or aberrant splicing events.
- To identify short exons, and to predict very long exons, more accurately.
- To identify non-translated exons.
- To predict polyadenylation sites and transcriptional termination sites.
- To identify mRNA features that are related to mRNA editing, nonsense-mediated decay, stability and transport.
- To predict genes that encode non-coding RNAs.
- To predict insulators and boundary elements, and matrix-attachment and scaffold-attachment regions.
- To predict replication origins and recombination hot spots.

To accommodate large genomic sequences, the traditional visualization tools, such as the simple dotplot, have been extended recently to more sophisticated programs, such as VISTA/AVID[72] and PipMaker[73], which both display the alignment of two or more genomes in the form of simple percentage-identify plots (for example, regions with 70% identity and above are shown). ROSETTA[74] is the first automated program that annotates human genes by using syntenic mouse genomic DNA. WABA (wobble aware bulk aligner[75]) has taken advantage of the third base wobble in coding exons to improve alignment, and has been successfully applied to aligning the genomes of two closely related worms, *Caenorhabditis briggsae* and *Caenorhabditis elegans*.

Computational tools for comparative genomics are being developed by several groups, and recently developed programs include CEM[76], TWINSCAN[57], SGP-1 (REF. 77) and SLAM (M. Alexandersson *et al.*, unpublished data) (BOX 2). By using comparisons between human and mouse, these groups have shown that gene-prediction accuracy can be further improved by using two closely related genomes. SLAM uses a generalized pair HMM (GPHMM or dual-HMM, REF. 78), which can simultaneously predict a pair of 'orthologous' base pairs according to a dual-HMM model (FIG. 5) in a syntenic region. This places the annotation and alignment problem on an equal footing. The mathematical beauty of the dual-HMM is quite appealing, but in its practical implementation, SLAM suffers from many restrictions (but perhaps also benefits by being faster to compute). For example, it is assumed that the same number of exons exist in each organism and in the same order in a region of conserved synteny, and certain key approximations of the genome-wide alignment (which are derived from a pre-processing step to reduce the computational complexity of the exact GPHMM) are used. As about one-half of the conserved regions between human and mouse are not in coding regions (M. Zhang, unpublished data), they have been the main source of false-positives in most comparative-genomics approaches. To reduce such errors, SLAM has introduced the conserved non-coding sequence (CNS) state. Although the CNS state allows SLAM to detect some homologous regulatory regions, the lack of a precise definition of CNS and a known CNS training set makes it the weakest point of this model. These programs and the programs that are now being developed, such as Doublescan (BOX 2), will have a great impact on finding new genes in vertebrate genomes, and, more importantly, will provide a testable list of genes for high-throughput experimental validation and refinement.

Future challenges

Gene-prediction algorithms have been steadily improving in the past decade, but there is still a long way to go. This is reflected by the fact that we still do not know how many genes are in the human genome[79], and fluctuations in the estimates of this number are now as big as the mean. In BOX 3, I list some of the key problems that remain to be solved in this field.

Bioinformatics is driven by genomic data, and the lack of high-throughput experimental approaches to identify genes and their functions has become the main bottleneck of this field[80]. However, computational biologists should not be deterred by not yet having experimental confirmation of their gene predictions, because many transcripts are hard to detect owing to their low abundance. We should work closely with our bench-partners, and together, many 'false-positives' (~30–50%) might be turned eventually into real positives (for example, see REF. 81). Although I have concentrated here on the computational side of this field, experimental approaches are equally, if not more, important. More functional-genomics methods for finding genes — such as using genomic microarrays to create transcription maps[82–84], sequencing full-length cDNAs and improving SAGE protocols (for example, see REF. 85) — are desperately needed. Furthermore, gene finding would not be complete without also identifying alternative transcripts and regulatory *cis*-elements. In this regard, functional-genomics approaches, such as ChIP (chromatin immunoprecipitation)-chip analyses (see the recent review by Horak and Snyder[86]) and large-scale analyses of alternative splicing (see, for example, REFS 87,88) have become the methods of choice. New algorithms will be needed to analyse such new data computationally. Together with the availability of the genomes of several species for comparative analyses, the gene-finding field is at its most exciting time. Despite large-scale genomic efforts, traditional single-gene dissections are still needed for understanding the details of gene-expression mechanisms. Only with sufficient mechanistic data can gene prediction be transformed from being statistical to being biological in nature[79]. Everyone in the field is working towards the ultimate dynamic model that can identify the consecutive exons of a gene, from its 5′- to its 3′-ends, as if they were being co-transcriptionally recognized and spliced[89,90].

1. Claverie, J.-M. Computational methods for the identification of genes in vertebrate genomic sequences. *Hum. Mol. Genet*. **6**, 1735–1744 (1997).
2. Burge, C. & Karlin, S. Prediction of complete gene structure in human genomic DNA. *J. Mol. Biol.* **268**, 78–94 (1997).
   In this paper, the popular Genscan gene-prediction algorithm was first reported.
3. Milanesi, L. & Rogozin, I. B. in *Guide to Human Genome Computing* 2nd edn (ed. Bishop, M. J.) 215–260 (Academic, New York, 1998).
4. Krogh, A. in *Guide to Human Genome Computing* 2nd edn (ed. Bishop, M. J.) 261–274 (Academic, New York, 1998).
5. Pavy, N. *et al.* Evaluation of gene prediction software using a genomic data set: application to *Arabidopsis thaliana* sequences. *Bioinformatics* **15**, 887–899 (1999).
6. Rogic, S., Mackworth, A. K. & Ouellette, F. B. F. Evaluation of gene-finding programs on mammalian sequences. *Genome Res*. **11**, 817–832 (2001).
7. Solovyev, V. V. in *Current Topics in Computational Molecular Biology* (eds Jiang, T., Xu, Y. & Zhang, M. Q.) 201–248 (MIT Press, Cambridge, Massachusetts, 2002).
   An up-to-date introduction and review on computational gene-prediction methods.
8. Brent, M. R. Predicting full-length transcripts. *Trends Biotechnol.* **20**, 273–275 (2002).
9. Zhang, M. Q. Statistical features of human exons and their flanking regions. *Hum. Mol. Genet.* **7**, 919–932 (1998).
10. Senapathy, P., Shapiro, M. B. & Harris, N. L. Splice junctions, branch point sites, and exons: sequence statistics, identification and application to genome project. *Methods Enzymol*. **183**, 252–278 (1990).
    A good introduction to the statistical features of splicing signals and exons.
11. Chen, T. & Zhang, M. Q. POMBE: a fission yeast gene-finding and exon–intron structure prediction system. *Yeast* **14**, 701–710 (1998).
12. Lim, L. P. & Burge, C. B. A computational analysis of sequence features involved in recognition of short introns. *Proc. Natl Acad. Sci. USA* **98**, 11193–11198 (2001).
    A systematic study of the sequence features that might define a short intron.
13. Robberson, B. L., Cote, G. J. & Berget, S. M. Exon definition may facilitate splice site selection in RNAs with multiple exons. *Mol. Cell. Biol*. **10**, 84–94 (1990).
14. Ripley, B. D. *Pattern Recognition and Neural Networks* (Cambridge Univ. Press, Cambridge, UK, 1996).
15. Solovyev, V. V., Salamov, A. A. & Lawrence, C. B. Predicting internal exons by oligonucleotide composition and discriminant analysis of spliceable open reading frames. *Nucleic Acids Res.* **22**, 248–250 (1994).
16. Pertea, M., Lin, X. & Salzberg, S. L. GeneSplicer: a new computational method for splice site prediction. *Nucleic Acids Res*. **29**, 1185–1190 (2001).
17. Fickett, J. W. & Tung, C.-S. Assessment of protein coding measures. *Nucleic Acids Res*. **20**, 6441–6450 (1992).
    This is a comprehensive assessment of protein-coding measures, which are used in many gene-prediction algorithms.
18. Salzberg, S. L., Delcher, A. L., Kasif, S. & White, O. Microbial gene identification using interpolated Markov models. *Nucleic Acids Res*. **26**, 544–548 (1998).
19. Bernardi, G. The human genome: organization and evolutionary history. *Annu. Rev. Genet.* **29**, 445–476 (1995).
20. Zhang, M. Q. Identification of protein coding regions in the human genome based on quadratic discriminant analysis. *Proc. Natl Acad. Sci. USA* **94**, 565–568 (1997).
21. Uberbacher, E. C. & Mural, R. J. Locating protein coding segments in human DNA sequences by a multiple sensor-neural network approach. *Proc. Natl Acad. Sci. USA* **88**, 11261–11265 (1991).
22. Graber, J. H., Cantor, C. R., Mohr, S. C. & Smith, T. F. *In silico* detection of control signals: mRNA 3′-end-processing sequences in diverse species. *Proc. Natl Acad. Sci. USA* **96**, 14055–14060 (1999).
23. Tabaska, J. E. & Zhang, M. Q. Detection of polyadenylation signals in human DNA sequences. *Gene* **231**, 77–86 (1999).
24. Tabaska, J. E., Davuluri, R. V. & Zhang, M. Q. Identifying the 3′-terminal exon in human DNA. *Bioinformatics* **17**, 602–607 (2001).
25. Schell, T., Kulozik, A. E. & Hentze, M. W. Integration of splicing, transport and translation to achieve mRNA quality control by the nonsense-mediated decay pathway. *Genome Biol.* **3**, ReviewS1006 (2002).
26. Cartegni, L., Chew, S. L. & Krainer, A. R. Listening to silence and understanding nonsense: exonic mutations that affect splicing. *Nature Rev. Genet*. **3**, 285–298 (2002).
27. Suzuki, Y. *et al.* DBTSS: database of human transcriptional start sites and full-length cDNAs. *Nucleic Acids Res*. **30**, 328–331 (2002).
28. Carey, M. & Smale, S. T. *Transcriptional Regulation in Eukaryotes: Concepts, Strategies, and Techniques* (Cold Spring Harbor Laboratory Press, New York, 2000).
29. Fickett, J. W. & Hatzigeorgiou, A. G. Eukaryotic promoter recognition. *Genome Res*. **7**, 861–878 (1997).
    The first comparison of promoter prediction programs.
30. Werner, T. Models for prediction and recognition of eukaryotic promoters. *Mamm. Genome* **23**, 168–175 (1999).
31. Ohler, U. & Niemann, H. Identification and analysis of eukaryotic promoters: recent computational approaches. *Trends Genet*. **17**, 56–60 (2001).
32. Zhang, M. Q. in *Current Topics in Computational Molecular Biology* (eds Jiang, T., Xu, Y. & Zhang, M. Q.) 249–268 (MIT Press, Cambridge, Massachusetts, 2002).
33. Ioshikhes, I. P. & Zhang, M. Q. Large-scale human promoter mapping using CpG islands. *Nature Genet.* **26**, 61–63 (2000).
34. Zhang, M. Q. Identification of human gene core promoters *in silico*. *Genome Res.* **8**, 319–326 (1998).
35. Scherf, M., Klingenhoff, A. & Werner, T. Highly specific localization of promoter regions in large genomic sequences by PromoterInspector: a novel context analysis approach. *J. Mol. Biol.* **297**, 599–606 (2000).
36. Solovyev, V. & Salamov, A. The Gene-Finder computer tools for analysis of human and model organisms genome sequences. *Proc. ISMB* **5**, 294–302 (1997).
37. Down, T. A. & Hubbard, T. J. P. Computational detection and location of transcription start sites in mammalian genomic DNA. *Genome Res*. **12**, 458–461 (2002).
38. Frech, K., Quandt, K. & Werner, T. Muscle actin genes: a first step towards computational classification of tissue specific promoters. *In Silico Biol*. **1**, 29–38 (1998).
39. Kel, A., Kel-Margoulis, O., Banemko, V. & Wingender, E. Recognition of NFATp/AP-1 composite elements within genes induced upon the activation of immune cells. *J. Mol. Biol*. **288**, 353–376 (1999).
40. Kozak, M. A progress report on translational control in eukaryotes. *SciSTKE* **2001**, PE1 (2001).
41. Davuluri, R. V., Grosse, I. & Zhang, M. Q. Computational identification of promoters and first exons in the human genome. *Nature Genet*. **29**, 412–417 (2001).
    The first report of a first-exon prediction algorithm.
42. Fickett, J. W. ORFs and genes: how strong a connection? *J. Comput. Biol*. **2**, 117–123 (1995).
43. Harrison, P. M. *et al.* Molecular fossils in the human genome: identification and analysis of the pseudogenes in chromosomes 21 and 22. *Genome Res*. **12**, 272–280 (2002).
44. Gelfand, M. S. & Roytberg, M. A. Prediction of the exon–intron structure by a dynamic programming approach. *Biosystems* **30**, 173–182 (1993).
45. Snyder, E. E. & Stormo, G. D. Identification of coding regions in genomic DNA sequences: an application of dynamic programming and neural networks. *Nucleic Acids Res.* **11**, 607–613 (1993).
46. Stormo, G. D. & Haussler, D. Optimally parsing a sequence into different classes based on multiple types of evidence. *Proc. Int. Conf. ISMB* **2**, 369–375 (1994).
47. Rabiner, L. R. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* **77**, 257–286 (1989).
48. Krogh, A. Two methods for improving performance of an HMM and their application for gene finding. *Proc. Int. Conf. Intell. Syst. Mol. Biol*. **5**, 179–186 (1997).
49. Kulp, D., Haussler, D., Reese, M. G. & Eeckman, F. H. A generalized hidden Markov model for the recognition of human genes in DNA. *Proc. Int. Conf. Intell. Syst. Mol. Biol*. **4**, 134–142 (1996).
50. Salamov, A. & Solovyev, V. *Ab initio* gene finding in *Drosophila* genome DNA. *Genome Res.* **10**, 516–522 (2000).
51. Hooper, P. M., Zhang, H. & Wishart, D. S. Prediction of genetic structure in eukaryotic DNA using reference point logistic regression and sequence alignment. *Bioinformatics* **16**, 425–438 (2000).
52. Cox, D. R. & Snell, E. J. *Analysis of Binary Data* 2nd edn (Chapman & Hall, London, 1989).
53. Rogic, S., Mackworth, A. K. & Ouellette, F. B. F. Improving gene recognition accuracy by combining predictions from two gene-finding programs. *Bioinformatics* (in the press).
54. Lukashin, A. V. & Borodovski, M. GeneMark.hmm: new solutions for gene finding. *Nucleic Acids Res*. **26**, 1107–1115 (1998).
55. Reese, M. G., Kulp, D., Tammana, H. & Haussler, D. Genie — gene finding in *Drosophila melanogaster*. *Genome Res*. **10**, 529–538 (2000).
56. Burset, M. & Guigo, R. Evaluation of gene structure prediction programs. *Genomics* **34**, 353–367 (1996).
    The first comprehensive evaluation of gene-prediction programs using a common standard training set.
57. Korf, I., Flicek, P., Duan, D. & Brent, M. R. Integrating genomic homology into gene structure prediction. *Bioinformatics* **17**(Suppl.), 140–148 (2001).
58. Frisch, M. *et al. In silico* prediction of scaffold/matrix attachment regions in large genome sequences. *Genome Res*. **12**, 349–354 (2002).
59. Zhan, H. C., Liu, D. P. & Liang, C. C. Insulator: from chromatin domain boundary to gene regulation. *Hum. Genet*. **109**, 471–478 (2001).
60. Gish, W. & States, D. J. Identification of protein coding regions by database similarity search. *Nature Genet*. **3**, 266–272 (1993).
61. Florea, L. *et al.* A computer program for aligning a cDNA sequence with a genomic DNA sequence. *Genome Res*. **8**, 967–974 (1998).
62. Gelfand, M. S., Mironov, A. & Pevner, P. Gene recognition via spliced sequence alignment. *Proc. Natl Acad. Sci. USA* **93**, 9061–9066 (1996).
63. Kulp, D., Haussler, D., Reese, M. G. & Eeckman, F. H. Integrating database homology in a probabilistic gene structure model. *Pacif. Symp. Biocomput*. 232–244 (1997).
64. Xu, Y. & Uberbacher, E. C. Gene prediction by pattern recognition and homology search. *Proc. Int. Conf. Intell. Syst. Mol. Biol.* **4**, 241–251 (1996).
65. Krogh, A. Using database matches with HMMgene for automated gene detection in *Drosophila*. *Genome Res*. **10**, 523–528 (2000).
66. Birney, E. & Durbin, R. Using GeneWise in the *Drosophila* annotation experiment. *Genome Res*. **10**, 547–548 (2000).
67. Gotoh, O. Homology-based gene structure prediction: simplified matching algorithm using a translated codon (tron) and improved accuracy by allowing for long gaps. *Bioinformatics* **16**, 190–202 (2000).
68. Guigo, R. *et al.* An assessment of gene prediction accuracy in large DNA sequences. *Genome Res*. **10**, 1631–1642 (2000).
    A comparison of *ab initio* and alignment-based gene-prediction programs.
69. Yeh, R. F., Lim, L. P. & Burge, C. B. Computational inference of homologous gene structures in the human genome. *Genome Res*. **11**, 803–816 (2001).
70. Reese, M. G. *et al.* Genome annotation assessment in *Drosophila melanogaster*. *Genome Res*. **10**, 483–501 (2000).
71. Pennacchio, L. A. & Rubin, E. M. Genomic strategies to identify mammalian regulatory sequences. *Nature Rev. Genet.* **2**, 100–119 (2001).
72. Mayor, C. *et al.* VISTA: visualizing global DNA sequence alignment of arbitrary length. *Bioinformatics* **16**,

1046–1047 (2000).
73. Schwartz, S. *et al.* PipMaker — a web server for aligning two genomic DNA sequences. *Genome Res*. **10**, 577–586 (2000).
74. Batzoglou, S. *et al.* Human and mouse gene structure: comparative analysis and application to exon prediction. *Genome Res*. **10**, 950–958 (2000).
75. Kent, W. J. & Zahler, A. M. Conservation, regulation, synteny, and introns in a large *C. briggsae–C. elegans* genomic alignment. *Genome Res*. **10**, 1115–1125 (2000).
76. Bafna, V. & Huson, D. H. The conserved exon method for gene finding. *Proc. Int. Conf. Intell. Syst. Mol. Biol.* **8**, 3–12 (2000).
77. Wiehe, T., Gebauer-Jung, S., Mitchell-Olds, T. & Guigo, R. SGP-1: prediction and validation of homologous genes based on sequence alignments. *Genome Res*. **11**, 1574–1583 (2001).
78. Pachter, L., Alexandersson, M. & Cawley, S. Applications of generalized pair hidden Markov models to alignment and gene finding problems. *J. Comput. Biol*. **9**, 389–399 (2002).
79. Claverie, J.-M. From bioinformatics to computational biology. *Genome Res*. **10**, 1277–1279 (2000).
80. Zhang, M. Q. Predicting full-length transcripts. *Nature Biotechnol.* **20**, 275 (2002).
81. Miyajima, N., Burge, C. B. & Saito, T. Computational and experimental analysis identifies many novel human genes. *Biochem. Biophys. Res. Commun*. **272**, 801–807 (2000).
82. Shoemaker, D. D. *et al.* Experimental annotation of the human genome using microarray technology. *Nature* **409**, 922–927 (2001).
83. Frazer, K. A. *et al.* Evolutionarily conserved sequences on human chromosome 21. *Genome Res*. **11**, 1651–1659 (2001).
84. Kapranov, P. *et al.* Large-scale transcriptional activity in chromosomes 21 and 22. *Science* **296**, 916–919 (2002).
85. Lee, S. *et al.* Correct identification of genes from serial analysis of gene expression tag sequences. *Genomics* **79**, 598–602 (2002).
86. Horak, C. E. & Snyder, M. ChIP-chip: a genomic approach for identifying transcription factor binding sites. *Methods Enzymol*. **350**, 469–483 (2002).
87. Clark, T. A., Sugnet, C. W. & Ares, M. Jr. Genomewide analysis of mRNA processing in yeast using splicing-specific microarrays. *Science* **296**, 907–910 (2002).
88. Yeakey, J. M. *et al.* Profiling alternative splicing on fiber-optic arrays. *Nature Biotechnol*. **20**, 353–358 (2002).
89. Goldstrohm, A. C., Greenleaf, A. L. & Garcia-Blanco, M. A. Co-transcriptional splicing of pre-messenger RNAs: considerations for the mechanism of alternative splicing. *Gene* **277**, 31–47 (2001).
90. Proudfoot, N. J., Furger, A. & Dye, M. J. Integrating mRNA processing with transcription. *Cell* **108**, 501–512 (2002).
A recent review on the interdependence of transcription and RNA processing.

Acknowledgements
My lab is supported by National Institutes of Health (NIH) grants. I thank L. Pachter and M. Alexandersson for providing their manuscript before publication; and R. Guigo and M. Brent for presenting their recent comparative analysis of human and mouse drafts at the 1% Workshop of NIH/NHGRI in July 2002. I also thank the anonymous reviewers for many helpful suggestions.

Online links

FURTHER INFORMATION
Ensembl mouse genome server:
http://www.ensembl.org/Mus_musculus
Entrez genome:
http://www.ncbi.nlm.nih.gov/Entrez/Genome/org.html
Mammalian Gene Collection:
http://mgc.nci.nih.gov/Info/ProjectSummary
RIKEN: http://www.gsc.riken.go.jp/e/FANTOM
University of Santa Cruz Genome Bioinformatics site:
http://genome.ucsc.edu
Access to this interactive links box is free online.

- With the recent explosion in the availability of genome data, gene-finding programs have proliferated. However, the accuracy with which genes can be predicted is still far from satisfactory. This review provides background information and surveys the latest developments in gene-prediction programs. It also highlights the problems that face the gene-prediction field and discusses future research goals.
- The main characteristic of a eukaryotic gene is its organization into exons and introns. The 'exon-definition' model explains how the splicing machinery recognizes exons in a sea of intronic DNA. It indicates that an internal exon is initially recognized by a chain of interacting splicing factors that span it. The binding of these factors to pre-mRNA is responsible for the non-random nucleotide patterns that form the molecular basis of all exon-recognition algorithms.
- Correctly identifying the boundaries of a gene is essential when searching for several genes in a large genomic region. It is relatively easy to find internal exons, but many gene-prediction programs fail to identify gene boundaries. Determining the 3′ end of a gene is easier than determining its 5′ end, mainly because of the difficulty of identifying the promoter and transcriptional start-site sequences, and because the 5′ ends of cDNA sequences are often truncated.
- As current gene-prediction programs are biased towards intron-containing genes, many intronless genes might have been missed by such programs. Many false-positive exon predictions have also been caused by pseudogenes. Developing better and more specialized algorithms to recognize them is becoming increasingly important.
- Hidden Markov model (HMM)-based programs can be used to predict multiple genes, partial genes and genes on both strands, all at the same time. These features are essential when annotating genomes or large chunks of sequence data, such as large contigs, in an automated fashion.
- By comparing the genomes of several closely related species, conserved regulatory regions can be identified easily. For these reasons, making use of comparative genomic data is an important future challenge for the gene-prediction field.
- More functional genomics methods for finding genes are desperately needed to improve gene prediction. Only with sufficient mechanistic data can gene prediction be transformed from being statistical to being biological in nature. The field is working towards the ultimate dynamic model that can identify the consecutive exons of a gene, from its 5′ to its 3′ ends, as if they were being co-transcriptionally recognized and spliced.

Michael Q. Zhang received his Ph.D. in Physics from Rutgers University in 1987, and did his postdoctoral research in applied mathematics in the Courant Institute of Mathematical Sciences at New York University, before he joined Cold Spring Harbor Laboratory in 1990 as a Genome Research Fellow. Since 1996, he has been on the faculty of the Cold Spring Harbor Laboratory and adjunct faculty of the State University of New York (SUNY) at Stony Brook. His research interest is in the computational biology of genome expression and regulation.

Links
Ensembl mouse genome server
http://www.ensembl.org/Mus_musculus
Entrez genome
http://www.ncbi.nlm.nih.gov/Entrez/Genome/org.html
Mammalian Gene Collection
http://mgc.nci.nih.gov/Info/ProjectSummary
RIKEN
http://www.gsc.riken.go.jp/e/FANTOM
University of Santa Cruz Genome Bioinformatics site
http://genome.ucsc.edu